Intel® Quark SoC X1000

Datasheet

October 2013

Document Number: 329676-001US

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/#/en_US_01

Code Names are only for use by Intel to identify products, platforms, programs, services, etc. (“products”) in development by Intel that have not been made commercially available to the public, i.e., announced, launched or shipped. They are never to be used as “commercial” names for products. Also, they are not intended to function as trademarks.

Intel and the Intel logo, are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2013, Intel Corporation. All rights reserved.

Contents

1.0 Introduction 37
1.1 About This Manual 37
1.2 Component Overview 37
1.2.1 SoC CPU Core Features 38
1.2.2 System Memory Controller Features 39
1.2.3 Embedded SRAM Features 39
1.2.4 Power Management Features 39
1.2.5 Security Features 39
1.2.6 PCI Express* Features 39
1.2.7 Ethernet Features 40
1.2.8 USB2 Host Controller Features 40
1.2.9 USB2 Device Controller Features 40
1.2.10 SD/SDIO/eMMC Controller Features 40
1.2.11 I^2C* Master Controller 40
1.2.12 GPIO Features 41
1.2.13 SPI Master Controller 41
1.2.14 High Speed UART Controller with DMA 41
1.2.15 Legacy Bridge 41
1.2.16 Package 41
1.3 Component Identification 41

2.0 Physical Interfaces 45
2.1 Pin States Through Reset 47
2.2 System Memory Signals 47
2.3 PCI Express* 2.0 Signals 48
2.4 Ethernet Interface Signals 49
2.5 USB 2.0 Interface Signals 49
2.6 Integrated Clock Interface Signals 50
2.7 SDIO/SD/MMC Signals 50
2.8 High Speed UART Interface Signals 51
2.9 I^2C* Interface Signals 51
2.10 Legacy Serial Peripheral Interface (SPI) Signals 52
2.11 Serial Peripheral Interface (SPI) 52
2.12 Real Time Clock (RTC) Interface Signals 53
2.13 Power Management Signals 53
2.14 JTAG and Debug Interface Signals 53
2.15 Legacy Interface Signals 54
2.16 General Purpose I/O Interface Signals 54
2.17 Power And Ground Pins 55
2.18 Hardware Straps 56

3.0 Ballout and Package Information 59

4.0 Electrical Characteristics 69
4.1 Absolute Maximum Ratings 69
4.2 Recommended Power Supply Ranges 70
4.3 Maximum Supply Current 71
4.4 Configurable IO Characteristics 72
4.5 RTC DC Characteristics 72
4.6 PCI Express* 2.0 DC/AC Characteristics 73
4.7 USB 2.0 DC/AC Characteristics 75

4.8 General Interface Timing 78
4.8.1 Legacy SPI Interface Timing 78
4.8.2 SPI0/1 Interface Timing 78
4.8.3 SDIO Interface Timing 79
4.9 Clock AC Timing 80
4.9.1 Reference Clock AC Characteristics 80

5.0 Register Access Methods 83
5.1 Fixed I/O Register Access 83
5.2 Fixed Memory Mapped Register Access 83
5.3 I/O Referenced Register Access 83
5.4 Memory Referenced Register Access 84
5.5 PCI Configuration Register Access 84
5.5.1 PCI Configuration Access - CAM: I/O Indexed Scheme 84
5.5.2 PCI Configuration Access - ECAM: Memory Mapped Scheme 85
5.6 Message Bus Register Access 86
5.7 Register Field Access Types 87

6.0 Mapping Address Spaces 89
6.1 Physical Address Space Mappings 89
6.1.1 Bridge Memory Map 89
6.1.1.1 MMIO 91
6.1.1.2 DOS DRAM 92
6.1.1.3 Additional Mappings 92
6.1.2 MMIO Map 93
6.2 I/O Address Space 93
6.2.1 Host Bridge I/O Map 94
6.2.2 I/O Fabric I/O Map 94
6.2.2.1 Legacy Bridge Fixed I/O Address Ranges 94
6.2.2.2 Variable I/O Address Ranges 94
6.3 PCI Configuration Space 95
6.4 Message Bus Space 97

7.0 Clocking 99
7.1 Clocking Features 99
7.2 Platform/System Clock Domains 100

8.0 Power Management 103
8.1 Power Management Features 103
8.2 ACPI Supported States 103
8.2.1 S-State Definition 103
8.2.1.1 S0 - Full On 103
8.2.1.2 S3 - Suspend to RAM (Standby) 103
8.2.1.3 S4 - Suspend to Disk (Hibernate) 103
8.2.1.4 S5 - Soft Off 104
8.2.2 System States 104
8.2.3 Processor Idle States 105
8.2.4 Integrated Memory Controller States 105
8.2.5 PCIe* States 105
8.2.6 Interface State Combinations 106
8.3 Processor Core Power Management 106
8.3.1 Low-Power Idle States 106
8.3.1.1 Clock Control and Low-Power States 106
8.3.2 Processor Core C-States Description 106
8.3.2.1 Core C0 State 106
8.3.2.2 Core C1 State 107
8.3.2.3 Core C2 State 107

8.4 Memory Controller Power Management 107
8.4.1 Disabling Unused System Memory Outputs 107
8.4.2 DRAM Power Management and Initialization 107
8.4.2.1 Initialization Role of CKE 107
8.4.2.2 Dynamic Self-Refresh 107
8.4.2.3 Dynamic Power Down Operation 108
8.4.2.4 Functional Clock Gating 108

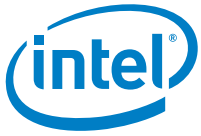
9.0 Power Up and Reset Sequence 109
9.1 Intel® Quark SoC X1000 System States 109
9.1.1 System Sleeping States Control (S-States) 109
9.2 Power Up and Down Sequences 109
9.2.1 Power Up, Wake and Reset Overview 109
9.2.2 RTC Power Well Transition: G5 to G3 State Transition 110
9.2.3 Power-Up Sequence without G2/G3: No Coin-Cell Battery 111
9.2.4 AC Power Applied: G3 to S4/S5 State Transition 112
9.2.5 Using PWR_BTN: Transition from S4/S5 to S0 112
9.2.6 Going to Sleep: Transitions from S0 to S3 or S4/S5 116
9.2.7 Wake Events: Transition from S3 to S0 116
9.2.8 System Reset Sequences 116
9.2.8.1 Cold Boot Sequence 117
9.2.8.2 Cold Reset Sequence 117
9.2.8.3 Warm Reset Sequence (Internal) 117
9.2.8.4 Externally Initiated Warm Reset Sequence 117
9.2.9 Handling Power Failures 117

10.0 Thermal Management 119
10.1 Overview 119
10.2 Thermal Sensor 119

11.0 Processor Core 121

12.0 Host Bridge 123
12.1 Embedded SRAM (eSRAM) 123
12.1.1 Initialization 123
12.1.2 Configuration 123
12.1.2.1 4KB Page Mode 123
12.1.2.2 512KB Block Page Mode 124
12.1.3 Configuration Locking 125
12.1.4 ECC Protection 126
12.1.5 Flush to DRAM 126
12.2 Isolated Memory Regions (IMR) 126
12.2.1 IMR Violation 127
12.2.2 IMR Locking 127
12.3 Remote Management Unit DMA 127
12.3.1 ECC Scrubbing 128
12.4 Register Map 128
12.5 PCI Configuration Registers 129
12.5.1 PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h 129
12.5.2 PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h 129
12.5.3 PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h 130
12.5.4 PCI Miscellaneous Fields (PCI_MISC)—Offset Ch 130
12.5.5 PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch 131
12.5.6 Message Bus Control Register (MCR) (SB_PACKET_REG)—Offset D0h 132
12.5.7 Message Data Register (MDR) (SB_DATA_REG)—Offset D4h 132

12.5.8 Message Control Register eXtension (MCRX) (SB_ADDR_EXTN_REG)—Offset D8h 132
12.5.9 Manufacturer ID (PCI_MANUFACTURER)—Offset F8h 133
12.6 IO Mapped Register 134
12.6.1 ACPI Processor Block 134
12.6.1.1 Processor Control (P_CNT)—Offset 0h 134
12.6.1.2 Level 2 Register (P_LVL2)—Offset 4h 135
12.6.1.3 C6 Control Register (P_C6C)—Offset Ch 135
12.6.2 SPI DMA Block 136
12.6.2.1 SPI DMA Count Register (SPI_DMA_CNT_IOSF)—Offset 0h 136
12.6.2.2 SPI DMA Destination Register (SPI_DMA_DST_IOSF)—Offset 4h 136
12.6.2.3 SPI DMA Source Register (SPI_DMA_SRC_IOSF)—Offset 8h 137
12.7 Message Bus Register 138
12.7.1 Host Bridge Arbiter (Port 0x00) 138
12.7.1.1 Enhanced Configuration Space (AEC_CTRL)—Offset 0h 138
12.7.1.2 STATUS—Offset 21h 138
12.7.1.3 Requester ID Match Control (ASUBCHAN_CTRL)—Offset 50h 139
12.7.1.4 Requester ID Match Sub-Channel 1 (ASUBCHAN1_MATCH)—Offset 51h 140
12.7.1.5 Requester ID Match Sub-Channel 2 (ASUBCHAN2_MATCH)—Offset 52h 141
12.7.1.6 Requester ID Match Sub-Channel 3 (ASUBCHAN3_MATCH)—Offset 53h 141
12.7.2 Host Bridge (Port 0x03) 142
12.7.2.1 Host Miscellaneous Controls 2 (HMISC2)—Offset 3h 143
12.7.2.2 Host System Management Mode Controls (HSMMCTL)—Offset 4h 144
12.7.2.3 Host Memory I/O Boundary (HMBOUND)—Offset 8h 145
12.7.2.4 Extended Configuration Space (HECREG)—Offset 9h 146
12.7.2.5 Host Bridge Write Flush Control (HWFLUSH)—Offset Ch 147
12.7.2.6 MTRR Capabilities (MTRR_CAP)—Offset 40h 147
12.7.2.7 MTRR Default Type (MTRR_DEF_TYPE)—Offset 41h 148
12.7.2.8 MTRR Fixed 64KB Range 0x00000 (MTRR_FIX64K_00000)—Offset 42h 148
12.7.2.9 MTRR Fixed 64KB Range 0x40000 (MTRR_FIX64K_40000)—Offset 43h 149
12.7.2.10 MTRR Fixed 16KB Range 0x80000 (MTRR_FIX16K_80000)—Offset 44h 149
12.7.2.11 MTRR Fixed 16KB Range 0x90000 (MTRR_FIX16K_90000)—Offset 45h 150
12.7.2.12 MTRR Fixed 16KB Range 0xA0000 (MTRR_FIX16K_A0000)—Offset 46h 151
12.7.2.13 MTRR Fixed 16KB Range 0xB0000 (MTRR_FIX16K_B0000)—Offset 47h 151
12.7.2.14 MTRR Fixed 4KB Range 0xC0000 (MTRR_FIX4K_C0000)—Offset 48h 152
12.7.2.15 MTRR Fixed 4KB Range 0xC4000 (MTRR_FIX4K_C4000)—Offset 49h 152
12.7.2.16 MTRR Fixed 4KB Range 0xC8000 (MTRR_FIX4K_C8000)—Offset 4Ah 153
12.7.2.17 MTRR Fixed 4KB Range 0xCC000 (MTRR_FIX4K_CC000)—Offset 4Bh 153
12.7.2.18 MTRR Fixed 4KB Range 0xD0000 (MTRR_FIX4K_D0000)—Offset 4Ch 154
12.7.2.19 MTRR Fixed 4KB Range 0xD40000 (MTRR_FIX4K_D4000)—Offset 4Dh 155
12.7.2.20 MTRR Fixed 4KB Range 0xD8000 (MTRR_FIX4K_D8000)—Offset 4Eh 155
12.7.2.21 MTRR Fixed 4KB Range 0xDC000 (MTRR_FIX4K_DC000)—Offset 4Fh 156



12.7.2.22 MTRR Fixed 4KB Range 0xE0000 (MTRR_FIX4K_E0000)—Offset 50h 156
12.7.2.23 MTRR Fixed 4KB Range 0xE4000 (MTRR_FIX4K_E4000)—Offset 51h 157
12.7.2.24 MTRR Fixed 4KB Range 0xE8000 (MTRR_FIX4K_E8000)—Offset 52h 157
12.7.2.25 MTRR Fixed 4KB Range 0xEC000 (MTRR_FIX4K_EC000)—Offset 53h 158
12.7.2.26 MTRR Fixed 4KB Range 0xF0000 (MTRR_FIX4K_F0000)—Offset 54h 158
12.7.2.27 MTRR Fixed 4KB Range 0xF4000 (MTRR_FIX4K_F4000)—Offset 55h 159
12.7.2.28 MTRR Fixed 4KB Range 0xF8000 (MTRR_FIX4K_F8000)—Offset 56h 160
12.7.2.29 MTRR Fixed 4KB Range 0xFC000 (MTRR_FIX4K_FC000)—Offset 57h 160
12.7.2.30 System Management Range Physical Base (MTRR_SMRR_PHYSBASE)—Offset 58h 161
12.7.2.31 System Management Range Physical Mask (MTRR_SMRR_PHYSMASK)—Offset 59h 161
12.7.2.32 MTRR Variable Range Physical Base 0 (MTRR_VAR_PHYSBASE0)—Offset 5Ah 162
12.7.2.33 MTRR Variable Range Physical Mask 0 (MTRR_VAR_PHYSMASK0)—Offset 5Bh 162
12.7.2.34 MTRR Variable Range Physical Base 1 (MTRR_VAR_PHYSBASE1)—Offset 5Ch 163
12.7.2.35 MTRR Variable Range Physical Mask 1 (MTRR_VAR_PHYSMASK1)—Offset 5Dh 164
12.7.2.36 MTRR Variable Range Physical Base 2 (MTRR_VAR_PHYSBASE2)—Offset 5Eh 164
12.7.2.37 MTRR Variable Range Physical Mask 2 (MTRR_VAR_PHYSMASK2)—Offset 5Fh 165
12.7.2.38 MTRR Variable Range Physical Base 3 (MTRR_VAR_PHYSBASE3)—Offset 60h 165
12.7.2.39 MTRR Variable Range Physical Mask 3 (MTRR_VAR_PHYSMASK3)—Offset 61h 166
12.7.2.40 MTRR Variable Range Physical Base 4 (MTRR_VAR_PHYSBASE4)—Offset 62h 166
12.7.2.41 MTRR Variable Range Physical Mask 4 (MTRR_VAR_PHYSMASK4)—Offset 63h 167
12.7.2.42 MTRR Variable Range Physical Base 5 (MTRR_VAR_PHYSBASE5)—Offset 64h 168
12.7.2.43 MTRR Variable Range Physical Mask 5 (MTRR_VAR_PHYSMASK5)—Offset 65h 168
12.7.2.44 MTRR Variable Range Physical Base 6 (MTRR_VAR_PHYSBASE6)—Offset 66h 169
12.7.2.45 MTRR Variable Range Physical Mask 6 (MTRR_VAR_PHYSMASK6)—Offset 67h 169
12.7.2.46 MTRR Variable Range Physical Base 7 (MTRR_VAR_PHYSBASE7)—Offset 68h 170
12.7.2.47 MTRR Variable Range Physical Mask 7 (MTRR_VAR_PHYSMASK7)—Offset 69h 170
12.7.3 Remote Management Unit (Port 0x04) 171
12.7.3.1 ECC Scrubber Configuration Register (P_CFG_50)—Offset 50h 171
12.7.3.2 SPI DMA Count Register (P_CFG_60)—Offset 60h 172
12.7.3.3 SPI DMA Destination Register (P_CFG_61)—Offset 61h 173
12.7.3.4 SPI DMA Source Register (P_CFG_62)—Offset 62h 173
12.7.3.5 Processor Register Block (P_BLK) Base Address (P_CFG_70)—Offset 70h 174
12.7.3.6 Control Register (P_CFG_71)—Offset 71h 174

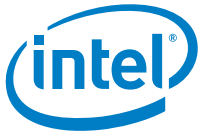
12.7.3.7 Watchdog Control Register (P_CFG_74)—Offset 74h 175
12.7.3.8 ECC Scrubber Start Address Register (P_CFG_76)—Offset 76h 176
12.7.3.9 ECC Scrubber End Address Register (P_CFG_77)—Offset 77h 176
12.7.3.10 ECC Scrubber Next Address Register (P_CFG_7C)—Offset 7Ch 177
12.7.3.11 Thermal Sensor Mode Register (P_CFG_B0)—Offset B0h 178
12.7.3.12 Thermal Sensor Temperature Register (P_CFG_B1)—Offset B1h 178
12.7.3.13 Thermal Sensor Programmable Trip Point Register (P_CFG_B2)—Offset B2h 179
12.7.4 Memory Manager (Port 0x05) 180
12.7.4.1 Control (BCTRL)—Offset 1h 181
12.7.4.2 Write Flush Policy (BWFLUSH)—Offset 2h 182
12.7.4.3 Isolated Memory Region Violation Control (BIMRVCTL)—Offset 19h 183
12.7.4.4 Debug 1 (DEBUG1)—Offset 31h 184
12.7.4.5 Isolated Memory Region 0 Low Address (IMR0L)—Offset 40h 185
12.7.4.6 Isolated Memory Region 0 High Address (IMR0H)—Offset 41h 186
12.7.4.7 Isolated Memory Region 0 Read Mask (IMR0RM)—Offset 42h 186
12.7.4.8 Isolated Memory Region 0 Write Mask (IMR0WM)—Offset 43h 188
12.7.4.9 Isolated Memory Region 1 Low Address (IMR1L)—Offset 44h 190
12.7.4.10 Isolated Memory Region 1 High Address (IMR1H)—Offset 45h 191
12.7.4.11 Isolated Memory Region 1 Read Mask (IMR1RM)—Offset 46h 191
12.7.4.12 Isolated Memory Region 1 Write Mask (IMR1WM)—Offset 47h 193
12.7.4.13 Isolated Memory Region 2 Low Address (IMR2L)—Offset 48h 195
12.7.4.14 Isolated Memory Region 2 High Address (IMR2H)—Offset 49h 195
12.7.4.15 Isolated Memory Region 2 Read Mask (IMR2RM)—Offset 4Ah 196
12.7.4.16 Isolated Memory Region 2 Write Mask (IMR2WM)—Offset 4Bh 198
12.7.4.17 Isolated Memory Region 3 Low Address (IMR3L)—Offset 4Ch 199
12.7.4.18 Isolated Memory Region 3 High Address (IMR3H)—Offset 4Dh 200
12.7.4.19 Isolated Memory Region 3 Read Mask (IMR3RM)—Offset 4Eh 200
12.7.4.20 Isolated Memory Region 3 Write Mask (IMR3WM)—Offset 4Fh 202
12.7.4.21 Isolated Memory Region 4 Low Address (IMR4L)—Offset 50h 204
12.7.4.22 Isolated Memory Region 4 High Address (IMR4H)—Offset 51h 205
12.7.4.23 Isolated Memory Region 4 Read Mask (IMR4RM)—Offset 52h 205
12.7.4.24 Isolated Memory Region 4 Write Mask (IMR4WM)—Offset 53h 207
12.7.4.25 Isolated Memory Region 5 Low Address (IMR5L)—Offset 54h 209
12.7.4.26 Isolated Memory Region 5 High Address (IMR5H)—Offset 55h 209
12.7.4.27 Isolated Memory Region 5 Read Mask (IMR5RM)—Offset 56h 210
12.7.4.28 Isolated Memory Region 5 Write Mask (IMR5WM)—Offset 57h 212
12.7.4.29 Isolated Memory Region 6 Low Address (IMR6L)—Offset 58h 213
12.7.4.30 Isolated Memory Region 6 High Address (IMR6H)—Offset 59h 214
12.7.4.31 Isolated Memory Region 6 Read Mask (IMR6RM)—Offset 5Ah 214
12.7.4.32 Isolated Memory Region 6 Write Mask (IMR6WM)—Offset 5Bh 216
12.7.4.33 Isolated Memory Region 7 Low Address (IMR7L)—Offset 5Ch 218
12.7.4.34 Isolated Memory Region 7 High Address (IMR7H)—Offset 5Dh 219
12.7.4.35 Isolated Memory Region 7 Read Mask (IMR7RM)—Offset 5Eh 219
12.7.4.36 Isolated Memory Region 7 Write Mask (IMR7WM)—Offset 5Fh 221
12.7.4.37 eSRAM Control (ESRAMCTRL)—Offset 81h 223
12.7.4.38 eSRAM Block Page Control (ESRAMPGCTRL_BLOCK)—Offset 82h 224
12.7.4.39 eSRAM Correctable Error (ESRAMCERR)—Offset 83h 225
12.7.4.40 eSRAM Uncorrectable Error (ESRAMUERR)—Offset 84h 226
12.7.4.41 eSRAM ECC Error Syndrome (ESRAMSDROME)—Offset 88h 227
12.7.5 Memory Manager eSRAM (Port 0x05) 228
12.7.5.1 eSRAM Page Control Register[0-127] (ESRAMPGCTRL[0-127])—Offset 0h, Count 128, Stride 4h 228
12.7.6 SoC Unit (Port 0x31) 229
12.7.6.1 Thermal Sensor Configuration 4 (SCU_TSCFG4_Config)—Offset 34h 229
12.7.6.2 Sticky Write Once (CFGSTICKY_W1)—Offset 50h 230
12.7.6.3 Sticky Read/Write (CFGSTICKY_RW)—Offset 51h 230
12.7.6.4 Non-Sticky Read/Write Once (CFGNONSTICKY_W1)—Offset 52h 231

13.0 System Memory Controller 233
13.1 Signal Descriptions 233
13.2 Features 234
13.2.1 System Memory Technology Supported 234
13.2.2 Rules for Populating Memory Down Ranks 235
13.2.3 DRAM Error Detection & Correction (EDC) 235
13.2.4 DRAM Data Scrambling 235
13.2.5 Power Management 235
13.3 Register Map 235
13.4 Message Bus Registers 236
13.4.1 DRAM Rank Population (DRP)—Offset 0h 237
13.4.2 DRAM Timing Register 0 (DTR0)—Offset 1h 238
13.4.3 DRAM Timing Register 1 (DTR1)—Offset 2h 240
13.4.4 DRAM Timing Register 2 (DTR2)—Offset 3h 242
13.4.5 DRAM Timing Register 3 (DTR3)—Offset 4h 243
13.4.6 DRAM Timing Register 4 (DTR4)—Offset 5h 244
13.4.7 DRAM Power Management Control 0 (DPMC0)—Offset 6h 245
13.4.8 DRAM Refresh Control (DRFC)—Offset 8h 247
13.4.9 DRAM Scheduler Control (DSCH)—Offset 9h 248
13.4.10 DRAM Calibration Control (DCAL)—Offset Ah 249
13.4.11 DRAM Reset Management Control (DRMC)—Offset Bh 250
13.4.12 Power Management Status (PMSTS)—Offset Ch 251
13.4.13 DRAM Control Operation (DCO)—Offset Fh 252
13.4.14 Sticky Scratchpad 0 (SSKPD0)—Offset 4Ah 252
13.4.15 Sticky Scratchpad 1 (SSKPD1)—Offset 4Bh 253
13.4.16 DRAM ECC Control Register (DECCCTRL)—Offset 60h 253
13.4.17 DRAM ECC Status (DECCSTAT)—Offset 61h 254
13.4.18 DRAM ECC Single Bit Error Count (DECCSBECNT)—Offset 62h 254
13.4.19 DRAM Single Bit ECC Error Captured Address (DECCSBECA)—Offset 68h 255
13.4.20 DRAM Single Bit ECC Error Captured Syndrome (DECCSBECS)—Offset 69h 256
13.4.21 DRAM Double Bit ECC Error Captured Address (DECCDBECA)—Offset 6Ah 256
13.4.22 DRAM Double Bit ECC Error Captured Syndrome (DECCDBECS)—Offset 6Bh 257
13.4.23 Memory Controller Fuse Status (DFUSESTAT)—Offset 70h 257
13.4.24 Scrambler Seed (DSCRMSEED)—Offset 80h 258

14.0 PCI Express* 2.0 259
14.1 Signal Descriptions 259
14.2 Features 259
14.2.1 Interrupts and Events 260
14.2.1.1 Express Card Hot Plug Events 260
14.2.1.2 System Error (SERR) 261
14.2.2 Power Management 261
14.3 References 261
14.4 Register Map 261
14.5 PCI Configuration Registers 262
14.5.1 Identifiers (ID)—Offset 0h 264
14.5.2 Primary Status (CMD_PSTS)—Offset 4h 264
14.5.3 Class Code (RID_CC)—Offset 8h 266
14.5.4 Header Type (CLS_PLT_HTYPE)—Offset Ch 266
14.5.5 Secondary Latency Timer (BNUM_SLT)—Offset 18h 267
14.5.6 Secondary Status (IOBL_SSTS)—Offset 1Ch 267
14.5.7 Memory Base and Limit (MBL)—Offset 20h 268
14.5.8 Prefetchable Memory Base and Limit (PMBL)—Offset 24h 269
14.5.9 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h 269
14.5.10 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch 270

14.5.11 Capabilities List Pointer (CAPP)—Offset 34h 270
14.5.12 Bridge Control (INTR_BCTRL)—Offset 3Ch 271
14.5.13 PCI Express Capabilities (CLIST_XCAP)—Offset 40h 272
14.5.14 Device Capabilities (DCAP)—Offset 44h 273
14.5.15 Device Status (DCTL_DSTS)—Offset 48h 274
14.5.16 Link Capabilities (LCAP)—Offset 4Ch 275
14.5.17 Link Status (LCTL_LSTS)—Offset 50h 277
14.5.18 Slot Capabilities (SLCAP)—Offset 54h 278
14.5.19 Slot Status (SLCTL_SLSTS)—Offset 58h 279
14.5.20 Root Control (RCTL)—Offset 5Ch 281
14.5.21 Root Status (RSTS)—Offset 60h 281
14.5.22 Device Capabilities 2 (DCAP2)—Offset 64h 282
14.5.23 Device Status 2 (DCTL2_DSTS2)—Offset 68h 283
14.5.24 Link Capability 2 (LCAP2)—Offset 6Ch 284
14.5.25 Link Status 2 (LCTL2_LSTS2)—Offset 70h 284
14.5.26 Slot Capabilities 2 (SLCAP2)—Offset 74h 286
14.5.27 Slot Status 2 (SLCTL2_SLSTS2)—Offset 78h 286
14.5.28 Message Signaled Interrupt Message Control (MID_MC)—Offset 80h 287
14.5.29 Message Signaled Interrupt Message Address (MA)—Offset 84h 287
14.5.30 Message Signaled Interrupt Message Data (MD)—Offset 88h 288
14.5.31 Subsystem Vendor Capability (SVCAP)—Offset 90h 288
14.5.32 Subsystem Vendor IDs (SVID)—Offset 94h 289
14.5.33 PCI Power Management Capabilities (PMCAP_PMC)—Offset A0h 289
14.5.34 PCI Power Management Control And Status (PMCS)—Offset A4h 290
14.5.35 Channel Configuration (CCFG)—Offset D0h 291
14.5.36 Miscellaneous Port Configuration 2 (MPC2)—Offset D4h 292
14.5.37 Miscellaneous Port Configuration (MPC)—Offset D8h 293
14.5.38 SMI / SCI Status (SMSCS)—Offset DCh 294
14.5.39 Message Bus Control (PHYCTL_PHYCTL2_IOSFSBCTL)—Offset F4h 295
14.5.40 Advanced Error Reporting Capability Header (AECH)—Offset 100h 296
14.5.41 Uncorrectable Error Status (UES)—Offset 104h 297
14.5.42 Uncorrectable Error Mask (UEM)—Offset 108h 298
14.5.43 Uncorrectable Error Severity (UEV)—Offset 10Ch 299
14.5.44 Correctable Error Status (CES)—Offset 110h 300
14.5.45 Correctable Error Mask (CEM)—Offset 114h 301
14.5.46 Advanced Error Capabilities and Control (AECC)—Offset 118h 302
14.5.47 Header Log (HL_DW1)—Offset 11Ch 302
14.5.48 Header Log (HL_DW2)—Offset 120h 303
14.5.49 Header Log (HL_DW3)—Offset 124h 303
14.5.50 Header Log (HL_DW4)—Offset 128h 303
14.5.51 Root Error Command (REC)—Offset 12Ch 304
14.5.52 Root Error Status (RES)—Offset 130h 304
14.5.53 Error Source Identification (ESID)—Offset 134h 305

15.0 10/100 Mbps Ethernet 307
15.1 Signal Descriptions 307
15.2 Features: 307
15.3 References 308
15.4 Register Map 309
15.5 PCI Configuration Registers 309
15.5.1 Vendor ID (VENDOR_ID)—Offset 0h 310
15.5.2 Device ID (DEVICE_ID)—Offset 2h 311
15.5.3 Command Register (COMMAND_REGISTER)—Offset 4h 311
15.5.4 Status Register (STATUS)—Offset 6h 312
15.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 313



15.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 313
15.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh 313
15.5.8 Header Type (HEADER_TYPE)—Offset Eh 314
15.5.9 BIST (BIST)—Offset Fh 314
15.5.10 Base Address Register (BAR0)—Offset 10h 315
15.5.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 315
15.5.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 316
15.5.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh 316
15.5.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 316
15.5.15 Capabilities Pointer (CAP_POINTER)—Offset 34h 317
15.5.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch 317
15.5.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh 318
15.5.18 MIN_GNT (MIN_GNT)—Offset 3Eh 318
15.5.19 MAX_LAT (MAX_LAT)—Offset 3Fh 318
15.5.20 Capability ID (PM_CAP_ID)—Offset 80h 319
15.5.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 319
15.5.22 Power Management Capabilities (PMC)—Offset 82h 319
15.5.23 Power Management Control/Status Register (PMCSR)—Offset 84h 320
15.5.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 321
15.5.25 Power Management Data Register (DATA_REGISTER)—Offset 87h 321
15.5.26 Capability ID (MSI_CAP_ID)—Offset A0h 322
15.5.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 322
15.5.28 Message Control (MESSAGE_CTRL)—Offset A2h 322
15.5.29 Message Address (MESSAGE_ADDR)—Offset A4h 323
15.5.30 Message Data (MESSAGE_DATA)—Offset A8h 323
15.5.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 324
15.5.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 324
15.6 Memory Mapped Registers 325
15.6.1 MAC Configuration Register (Register 0) (GMAC_REG_0)—Offset 0h 329
15.6.2 MAC Frame Filter (Register 1) (GMAC_REG_1)—Offset 4h 332
15.6.3 Hash Table High Register (Register 2) (GMAC_REG_2)—Offset 8h 334
15.6.4 Hash Table Low Register (Register 3) (GMAC_REG_3)—Offset Ch 334
15.6.5 GMII Address Register (Register 4) (GMAC_REG_4)—Offset 10h 335
15.6.6 GMII Data Register (Register 5) (GMAC_REG_5)—Offset 14h 336
15.6.7 Flow Control Register (Register 6) (GMAC_REG_6)—Offset 18h 337
15.6.8 VLAN Tag Register (Register 7) (GMAC_REG_7)—Offset 1Ch 338
15.6.9 Version Register (Register 8) (GMAC_REG_8)—Offset 20h 339
15.6.10 Debug Register (Register 9) (GMAC_REG_9)—Offset 24h 340
15.6.11 Interrupt Register (Register 14) (GMAC_REG_14)—Offset 38h 341
15.6.12 Interrupt Mask Register (Register 15) (GMAC_REG_15)—Offset 3Ch 342
15.6.13 MAC Address0 High Register (Register 16) (GMAC_REG_16)—Offset 40h 343
15.6.14 MAC Address0 Low Register (Register 17) (GMAC_REG_17)—Offset 44h 343
15.6.15 MMC Control Register (Register 64) (GMAC_REG_64)—Offset 100h 344
15.6.16 MMC Receive Interrupt Register (MMC_INTR_RX)—Offset 104h 345
15.6.17 MMC Transmit Interrupt Register (MMC_INTR_TX)—Offset 108h 347
15.6.18 MMC Receive Interrupt Mask Register (MMC_INTR_MASK_RX)—Offset 10Ch 349
15.6.19 MMC Transmit Interrupt Mask Register (MMC_INTR_MASK_TX)—Offset 110h 351
15.6.20 MMC Transmit Good Bad Octet Counter Register (TXOCTETCOUNT_GB)—Offset 114h 353
15.6.21 MMC Transmit Good Bad Frame Counter Register (TXFRAMECOUNT_GB)—Offset 118h 353
15.6.22 MMC Transmit Broadcast Good Frame Counter Register (TXBROADCASTFRAMES_G)—Offset 11Ch 354
15.6.23 MMC Transmit Multicast Good Frame Counter Register (TXMULTICASTFRAMES_G)—Offset 120h 354

15.6.24 MMC Transmit 64 Octet Good Bad Frame Counter Register (TX64OCTETS_GB)—Offset 124h 355
15.6.25 MMC Transmit 65 to 127 Octet Good Bad Frame Counter Register (TX65TO127OCTETS_GB)—Offset 128h 355
15.6.26 MMC Transmit 128 to 255 Octet Good Bad Frame Counter Register (TX128TO255OCTETS_GB)—Offset 12Ch 356
15.6.27 MMC Transmit 256 to 511 Octet Good Bad Frame Counter Register (TX256TO511OCTETS_GB)—Offset 130h 356
15.6.28 MMC Transmit 512 to 1023 Octet Good Bad Frame Counter Register (TX512TO1023OCTETS_GB)—Offset 134h 356
15.6.29 MMC Transmit 1024 to Maximum Octet Good Bad Frame Counter Register (TX1024TOMAXOCTETS_GB)—Offset 138h 357
15.6.30 MMC Transmit Unicast Good Bad Frame Counter Register (TXUNICASTFRAMES_GB)—Offset 13Ch 357
15.6.31 MMC Transmit Multicast Good Bad Frame Counter Register (TXMULTICASTFRAMES_GB)—Offset 140h 358
15.6.32 MMC Transmit Broadcast Good Bad Frame Counter Register (TXBROADCASTFRAMES_GB)—Offset 144h 358
15.6.33 MMC Transmit Underflow Error Frame Counter Register (TXUNDERFLOWERROR)—Offset 148h 359
15.6.34 MMC Transmit Single Collision Good Frame Counter Register (TXSINGLECOL_G)—Offset 14Ch 359
15.6.35 MMC Transmit Multiple Collision Good Frame Counter Register (TXMULTICOL_G)—Offset 150h 360
15.6.36 MMC Transmit Deferred Frame Counter Register (TXDEFERRED)—Offset 154h 360
15.6.37 MMC Transmit Late Collision Frame Counter Register (TXLATECOL)—Offset 158h 360
15.6.38 MMC Transmit Excessive Collision Frame Counter Register (TXEXESSCOL)—Offset 15Ch 361
15.6.39 MMC Transmit Carrier Error Frame Counter Register (TXCARRIERERROR)—Offset 160h 361
15.6.40 MMC Transmit Good Octet Counter Register (TXOCTETCOUNT_G)—Offset 164h 362
15.6.41 MMC Transmit Good Frame Counter Register (TXFRAMECOUNT_G)—Offset 168h 362
15.6.42 MMC Transmit Excessive Deferral Frame Counter Register (TXEXCESSDEF)—Offset 16Ch 363
15.6.43 MMC Transmit Pause Frame Counter Register (TXPAUSEFRAMES)—Offset 170h 363
15.6.44 MMC Transmit VLAN Good Frame Counter Register (TXVLANFRAMES_G)—Offset 174h 364
15.6.45 MMC Transmit Oversize Good Frame Counter Register (TXOVERSIZE_G)—Offset 178h 364
15.6.46 MMC Receive Good Bad Frame Counter Register (RXFRAMECOUNT_GB)—Offset 180h 364
15.6.47 MMC Receive Good Bad Octet Counter Register (RXOCTETCOUNT_GB)—Offset 184h 365
15.6.48 MMC Receive Good Octet Counter Register (RXOCTETCOUNT_G)—Offset 188h 365
15.6.49 MMC Receive Broadcast Good Frame Counter Register (RXBROADCASTFRAMES_G)—Offset 18Ch 366
15.6.50 MMC Receive Multicast Good Frame Counter Register (RXMULTICASTFRAMES_G)—Offset 190h 366
15.6.51 MMC Receive CRC Error Frame Counter Register (RXCRCERROR)—Offset 194h 367

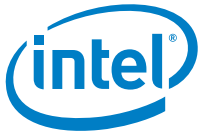
15.6.52 MMC Receive Alignment Error Frame Counter Register (RXALIGNMENTERROR)—Offset 198h 367
15.6.53 MMC Receive Runt Frame Counter Register (RXRUNTERROR)—Offset 19Ch .. 368
15.6.54 MMC Receive Jabber Error Frame Counter Register (RXJABBERERROR)—Offset 1A0h 368
15.6.55 MMC Receive Undersize Good Frame Counter Register (RXUNDERSIZE_G)—Offset 1A4h 368
15.6.56 MMC Receive Oversize Good Frame Counter Register (RXOVERSIZE_G)—Offset 1A8h 369
15.6.57 MMC Receive 64 Octet Good Bad Frame Counter Register (RX64OCTETS_GB)—Offset 1ACh 369
15.6.58 MMC Receive 65 to 127 Octet Good Bad Frame Counter Register (RX65TO127OCTETS_GB)—Offset 1B0h 370
15.6.59 MMC Receive 128 to 255 Octet Good Bad Frame Counter Register (RX128TO255OCTETS_GB)—Offset 1B4h 370
15.6.60 MMC Receive 256 to 511 Octet Good Bad Frame Counter Register (RX256TO511OCTETS_GB)—Offset 1B8h 371
15.6.61 MMC Receive 512 to 1023 Octet Good Bad Frame Counter Register (RX512TO1023OCTETS_GB)—Offset 1BCh 371
15.6.62 MMC Receive 1024 to Maximum Octet Good Bad Frame Counter Register (RX1024TOMAXOCTETS_GB)—Offset 1C0h 372
15.6.63 MMC Receive Unicast Good Frame Counter Register (RXUNICASTFRAMES_G)—Offset 1C4h 372
15.6.64 MMC Receive Length Error Frame Counter Register (RXLENGTHERROR)—Offset 1C8h 372
15.6.65 MMC Receive Out Of Range Error Frame Counter Register (RXOUTOFRANGETYPE)—Offset 1CCh 373
15.6.66 MMC Receive Pause Frame Counter Register (RXPAUSEFRAMES)—Offset 1D0h 373
15.6.67 MMC Receive FIFO Overflow Frame Counter Register (RXFIFOOVERFLOW)—Offset 1D4h 374
15.6.68 MMC Receive VLAN Good Bad Frame Counter Register (RXVLANFRAMES_GB)—Offset 1D8h 374
15.6.69 MMC Receive Watchdog Error Frame Counter Register (RXWATCHDOGERROR)—Offset 1DCh 375
15.6.70 MMC Receive Error Frame Counter Register (RXRCVERROR)—Offset 1E0h 375
15.6.71 MMC Receive Control Frame Counter Register (RXCTRLFRAMES_G)—Offset 1E4h 376
15.6.72 MMC IPC Receive Checksum Offload Interrupt Mask Register (MMC_IPC_INTR_MASK_RX)—Offset 200h 376
15.6.73 MMC Receive Checksum Offload Interrupt Register (MMC_IPC_INTR_RX)—Offset 208h 378
15.6.74 MMC Receive IPV4 Good Frame Counter Register (RXIPV4_GD_FRMS)—Offset 210h 380
15.6.75 MMC Receive IPV4 Header Error Frame Counter Register (RXIPV4_HDRERR_FRMS)—Offset 214h 381
15.6.76 MMC Receive IPV4 No Payload Frame Counter Register (RXIPV4_NOPAY_FRMS)—Offset 218h 381
15.6.77 MMC Receive IPV4 Fragmented Frame Counter Register (RXIPV4_FRAG_FRMS)—Offset 21Ch 382
15.6.78 MMC Receive IPV4 UDP Checksum Disabled Frame Counter Register (RXIPV4_UDSBL_FRMS)—Offset 220h 382
15.6.79 MMC Receive IPV6 Good Frame Counter Register (RXIPV6_GD_FRMS)—Offset 224h 382
15.6.80 MMC Receive IPV6 Header Error Frame Counter Register (RXIPV6_HDRERR_FRMS)—Offset 228h 383

15.6.81 MMC Receive IPV6 No Payload Frame Counter Register (RXIPV6_NOPAY_FRMS)—Offset 22Ch 383
15.6.82 MMC Receive UDP Good Frame Counter Register (RXUDP_GD_FRMS)—Offset 230h 384
15.6.83 MMC Receive UDP Error Frame Counter Register (RXUDP_ERR_FRMS)—Offset 234h 384
15.6.84 MMC Receive TCP Good Frame Counter Register (RXTCP_GD_FRMS)—Offset 238h 385
15.6.85 MMC Receive TCP Error Frame Counter Register (RXTCP_ERR_FRMS)—Offset 23Ch 385
15.6.86 MMC Receive ICMP Good Frame Counter Register (RXICMP_GD_FRMS)—Offset 240h 386
15.6.87 MMC Receive ICMP Error Frame Counter Register (RXICMP_ERR_FRMS)—Offset 244h 386
15.6.88 MMC Receive IPV4 Good Octet Counter Register (RXIPV4_GD_OCTETS)—Offset 250h 386
15.6.89 MMC Receive IPV4 Header Error Octet Counter Register (RXIPV4_HDRERR_OCTETS)—Offset 254h 387
15.6.90 MMC Receive IPV4 No Payload Octet Counter Register (RXIPV4_NOPAY_OCTETS)—Offset 258h 387
15.6.91 MMC Receive IPV4 Fragmented Octet Counter Register (RXIPV4_FRAG_OCTETS)—Offset 25Ch 388
15.6.92 MMC Receive IPV4 UDP Checksum Disabled Octet Counter Register (RXIPV4_UDSBL_OCTETS)—Offset 260h 388
15.6.93 MMC Receive IPV6 Good Octet Counter Register (RXIPV6_GD_OCTETS)—Offset 264h 389
15.6.94 MMC Receive IPV6 Good Octet Counter Register (RXIPV6_HDRERR_OCTETS)—Offset 268h 389
15.6.95 MMC Receive IPV6 Header Error Octet Counter Register (RXIPV6_NOPAY_OCTETS)—Offset 26Ch 390
15.6.96 MMC Receive IPV6 No Payload Octet Counter Register (RXUDP_GD_OCTETS)—Offset 270h 390
15.6.97 MMC Receive UDP Good Octet Counter Register (RXUDP_ERR_OCTETS)—Offset 274h 390
15.6.98 MMC Receive TCP Good Octet Counter Register (RXTCP_GD_OCTETS)—Offset 278h 391
15.6.99 MMC Receive TCP Error Octet Counter Register (RXTCP_ERR_OCTETS)—Offset 27Ch 391
15.6.100 MMC Receive ICMP Good Octet Counter Register (RXICMP_GD_OCTETS)—Offset 280h 392
15.6.101 MMC Receive ICMP Error Octet Counter Register (RXICMP_ERR_OCTETS)—Offset 284h 392
15.6.102 VLAN Tag Inclusion or Replacement Register (Register 353) (GMAC_REG_353)—Offset 584h 393
15.6.103 VLAN Hash Table Register (Register 354) (GMAC_REG_354)—Offset 588h 394
15.6.104 Timestamp Control Register (Register 448) (GMAC_REG_448)—Offset 700h 394
15.6.105 Sub-Second Increment Register (Register 449) (GMAC_REG_449)—Offset 704h 396
15.6.106 System Time - Seconds Register (Register 450) (GMAC_REG_450)—Offset 708h 396
15.6.107 System Time - Nanoseconds Register (Register 451) (GMAC_REG_451)—Offset 70Ch 397
15.6.108 System Time - Seconds Update Register (Register 452) (GMAC_REG_452)—Offset 710h 397
15.6.109 System Time - Nanoseconds Update Register (Register 453) (GMAC_REG_453)—Offset 714h 398
15.6.110 Timestamp Addend Register (Register 454) (GMAC_REG_454)—Offset 718h 398

15.6.111Target Time Seconds Register (Register 455) (GMAC_REG_455)—Offset 71Ch 399
15.6.112Target Time Nanoseconds Register (Register 456) (GMAC_REG_456)—Offset 720h 399
15.6.113System Time - Higher Word Seconds Register (Register 457) (GMAC_REG_457)—Offset 724h 400
15.6.114Timestamp Status Register (Register 458) (GMAC_REG_458)—Offset 728h 401
15.6.115Bus Mode Register (Register 0) (DMA_REG_0)—Offset 1000h 402
15.6.116Transmit Poll Demand Register (Register 1) (DMA_REG_1)—Offset 1004h .. 404
15.6.117Receive Poll Demand Register (Register 2) (DMA_REG_2)—Offset 1008h.... 404
15.6.118Receive Descriptor List Address Register (Register 3) (DMA_REG_3)—Offset 100Ch 405
15.6.119Transmit Descriptor List Address Register (Register 4) (DMA_REG_4)—Offset 1010h 405
15.6.120Status Register (Register 5) (DMA_REG_5)—Offset 1014h 406
15.6.121Operation Mode Register (Register 6) (DMA_REG_6)—Offset 1018h 409
15.6.122Interrupt Enable Register (Register 7) (DMA_REG_7)—Offset 101Ch 412
15.6.123Missed Frame and Buffer Overflow Counter Register (Register 8) (DMA_REG_8)—Offset 1020h 413
15.6.124Receive Interrupt Watchdog Timer Register (Register 9) (DMA_REG_9)—Offset 1024h 414
15.6.125AHB Status Register (Register 11) (DMA_REG_11)—Offset 102Ch 414
15.6.126Current Host Transmit Descriptor Register (Register 18) (DMA_REG_18)—Offset 1048h 415
15.6.127Current Host Receive Descriptor Register (Register 19) (DMA_REG_19)—Offset 104Ch 415
15.6.128Current Host Transmit Buffer Address Register (Register 20) (DMA_REG_20)—Offset 1050h 416
15.6.129Current Host Receive Buffer Address Register (Register 21) (DMA_REG_21)—Offset 1054h 416
15.6.130HW Feature Register (Register 22) (DMA_REG_22)—Offset 1058h 417

16.0 USB 2.0 421
16.1 Signal Descriptions 421
16.2 Features 421
16.2.1 USB2.0 Host Controller Features 421
16.2.2 USB2.0 Device Features 422
16.3 References 422
16.4 Register Map 423
16.5 PCI Configuration Registers 423
16.5.1 USB Device 423
16.5.1.1 Vendor ID (VENDOR_ID)—Offset 0h 424
16.5.1.2 Device ID (DEVICE_ID)—Offset 2h 425
16.5.1.3 Command Register (COMMAND_REGISTER)—Offset 4h 425
16.5.1.4 Status Register (STATUS)—Offset 6h 426
16.5.1.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 426
16.5.1.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 427
16.5.1.7 Latency Timer (LATENCY_TIMER)—Offset Dh 427
16.5.1.8 Header Type (HEADER_TYPE)—Offset Eh 428
16.5.1.9 BIST (BIST)—Offset Fh 428
16.5.1.10Base Address Register (BAR0)—Offset 10h 429
16.5.1.11Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 429
16.5.1.12Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 430
16.5.1.13Subsystem ID (SUB_SYS_ID)—Offset 2Eh 430
16.5.1.14Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 430
16.5.1.15Capabilities Pointer (CAP_POINTER)—Offset 34h 431
16.5.1.16Interrupt Line Register (INTR_LINE)—Offset 3Ch 431
16.5.1.17Interrupt Pin Register (INTR_PIN)—Offset 3Dh 432

16.5.1.18MIN_GNT (MIN_GNT)—Offset 3Eh 432
16.5.1.19MAX_LAT (MAX_LAT)—Offset 3Fh 432
16.5.1.20Capability ID (PM_CAP_ID)—Offset 80h 433
16.5.1.21Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 433
16.5.1.22Power Management Capabilities (PMC)—Offset 82h 433
16.5.1.23Power Management Control/Status Register (PMCSR)—Offset 84h . 434
16.5.1.24PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 435
16.5.1.25Power Management Data Register (DATA_REGISTER)—Offset 87h . 435
16.5.1.26Capability ID (MSI_CAP_ID)—Offset A0h 436
16.5.1.27Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 436
16.5.1.28Message Control (MESSAGE_CTRL)—Offset A2h 436
16.5.1.29Message Address (MESSAGE_ADDR)—Offset A4h 437
16.5.1.30Message Data (MESSAGE_DATA)—Offset A8h 437
16.5.1.31Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 438
16.5.1.32Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 438
16.5.2 USB EHCI 439
16.5.2.1 Vendor ID (VENDOR_ID)—Offset 0h 440
16.5.2.2 Device ID (DEVICE_ID)—Offset 2h 440
16.5.2.3 Command Register (COMMAND_REGISTER)—Offset 4h 440
16.5.2.4 Status Register (STATUS)—Offset 6h 441
16.5.2.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 442
16.5.2.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 442
16.5.2.7 Latency Timer (LATENCY_TIMER)—Offset Dh 443
16.5.2.8 Header Type (HEADER_TYPE)—Offset Eh 443
16.5.2.9 BIST (BIST)—Offset Fh 443
16.5.2.10Base Address Register (BAR0)—Offset 10h 444
16.5.2.11Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 445
16.5.2.12Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 445
16.5.2.13Subsystem ID (SUB_SYS_ID)—Offset 2Eh 445
16.5.2.14Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h. 446
16.5.2.15Capabilities Pointer (CAP_POINTER)—Offset 34h 446
16.5.2.16Interrupt Line Register (INTR_LINE)—Offset 3Ch 446
16.5.2.17Interrupt Pin Register (INTR_PIN)—Offset 3Dh 447
16.5.2.18MIN_GNT (MIN_GNT)—Offset 3Eh 447
16.5.2.19MAX_LAT (MAX_LAT)—Offset 3Fh 448
16.5.2.20Serial Bus Release Number Register (SBRN)—Offset 60h 448
16.5.2.21Frame Length Adjustment Register (FLADJ)—Offset 61h 448
16.5.2.22Capability ID (PM_CAP_ID)—Offset 80h 449
16.5.2.23Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 449
16.5.2.24Power Management Capabilities (PMC)—Offset 82h 449
16.5.2.25Power Management Control/Status Register (PMCSR)—Offset 84h . 450
16.5.2.26PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 451
16.5.2.27Power Management Data Register (DATA_REGISTER)—Offset 87h . 451
16.5.2.28Capability ID (MSI_CAP_ID)—Offset A0h 451
16.5.2.29Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 452
16.5.2.30Message Control (MESSAGE_CTRL)—Offset A2h 452
16.5.2.31Message Address (MESSAGE_ADDR)—Offset A4h 453
16.5.2.32Message Data (MESSAGE_DATA)—Offset A8h 453
16.5.2.33Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 454
16.5.2.34Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 454
16.5.2.35USB Legacy Support Extended Capability (USBLEGSUP)—Offset C0h 454
16.5.2.36USB Legacy Support Control/Status (USBLEGCTLSTS)—Offset C4h 455
16.5.3 USB OHCI 457
16.5.3.1 Vendor ID (VENDOR_ID)—Offset 0h 458
16.5.3.2 Device ID (DEVICE_ID)—Offset 2h 458
16.5.3.3 Command Register (COMMAND_REGISTER)—Offset 4h 459
16.5.3.4 Status Register (STATUS)—Offset 6h 459



16.5.3.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 460
16.5.3.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 461
16.5.3.7 Latency Timer (LATENCY_TIMER)—Offset Dh 461
16.5.3.8 Header Type (HEADER_TYPE)—Offset Eh 461
16.5.3.9 BIST (BIST)—Offset Fh 462
16.5.3.10 Base Address Register (BAR0)—Offset 10h 462
16.5.3.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 463
16.5.3.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 463
16.5.3.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh 464
16.5.3.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 464
16.5.3.15 Capabilities Pointer (CAP_POINTER)—Offset 34h 465
16.5.3.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch 465
16.5.3.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh 465
16.5.3.18 MIN_GNT (MIN_GNT)—Offset 3Eh 466
16.5.3.19 MAX_LAT (MAX_LAT)—Offset 3Fh 466
16.5.3.20 Capability ID (PM_CAP_ID)—Offset 80h 466
16.5.3.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 467
16.5.3.22 Power Management Capabilities (PMC)—Offset 82h 467
16.5.3.23 Power Management Control/Status Register (PMCSR)—Offset 84h . 468
16.5.3.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 469
16.5.3.25 Power Management Data Register (DATA_REGISTER)—Offset 87h . 469
16.5.3.26 Capability ID (MSI_CAP_ID)—Offset A0h 469
16.5.3.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 470
16.5.3.28 Message Control (MESSAGE_CTRL)—Offset A2h 470
16.5.3.29 Message Address (MESSAGE_ADDR)—Offset A4h 470
16.5.3.30 Message Data (MESSAGE_DATA)—Offset A8h 471
16.5.3.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 471
16.5.3.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 472
16.6 Memory Mapped Registers 472
16.6.1 USB Device 472
16.6.1.1 IN Endpoint 0 Control Register (ep0_in_ctrl_udc_reg)—Offset 0h .. 475
16.6.1.2 IN Endpoint 0 Status Register (ep0_in_sts_udc_reg)—Offset 4h 476
16.6.1.3 IN Endpoint 0 Buffer Size Register (ep0_in_bufsize_udc_reg)—Offset 8h 478
16.6.1.4 IN Endpoint 0 Maximum Packet Size Register (ep0_in_mpkt_sz_reg)—Offset Ch 479
16.6.1.5 IN Endpoint 0 Data Descriptor Pointer Register (ep0_in_desptr_udc_reg)—Offset 14h 479
16.6.1.6 IN Endpoint 0 Write Confirmation register (for Slave-Only mode) (ep0_wr_cfrm_udc_reg)—Offset 1Ch 480
16.6.1.7 IN Endpoint 1 Control Register (ep1_in_ctrl_udc_reg)—Offset 20h 480
16.6.1.8 IN Endpoint 1 Status Register (ep1_in_sts_udc_reg)—Offset 24h .. 481
16.6.1.9 IN Endpoint 1 Buffer Size Register (ep1_in_bufsize_udc_reg)—Offset 28h 483
16.6.1.10 IN Endpoint 1 Maximum Packet Size Register (ep1_in_mpkt_sz_reg)—Offset 2Ch 484
16.6.1.11 IN Endpoint 1 Data Descriptor Pointer Register (ep1_in_desptr_udc_reg)—Offset 34h 484
16.6.1.12 IN Endpoint 1 Write Confirmation register (for Slave-Only mode) (ep1_wr_cfrm_udc_reg)—Offset 3Ch 485
16.6.1.13 IN Endpoint 2 Control Register (ep2_in_ctrl_udc_reg)—Offset 40h 485
16.6.1.14 IN Endpoint 2 Status Register (ep2_in_sts_udc_reg)—Offset 44h .. 486
16.6.1.15 IN Endpoint 2 Buffer Size Register (ep2_in_bufsize_udc_reg)—Offset 48h 488
16.6.1.16 IN Endpoint 2 Maximum Packet Size Register (ep2_in_mpkt_sz_reg)—Offset 4Ch 489
16.6.1.17 IN Endpoint 2 Data Descriptor Pointer Register (ep2_in_desptr_udc_reg)—Offset 54h 489

16.6.1.18IN Endpoint 2 Write Confirmation register (for Slave-Only mode) (ep2_wr_cfrm_udc_reg)—Offset 5Ch 490
16.6.1.19IN Endpoint 3 Control Register (ep3_in_ctrl_udc_reg)—Offset 60h . 490
16.6.1.20IN Endpoint 3 Status Register (ep3_in_sts_udc_reg)—Offset 64h... 491
16.6.1.21IN Endpoint 3 Buffer Size Register (ep3_in_bufsize_udc_reg)—Offset 68h 493
16.6.1.22IN Endpoint 3 Maximum Packet Size Register (ep3_in_mpkt_sz_reg)—Offset 6Ch 494
16.6.1.23IN Endpoint 3 Data Descriptor Pointer Register (ep3_in_desptr_udc_reg)—Offset 74h 494
16.6.1.24IN Endpoint 3 Write Confirmation register (for Slave-Only mode) (ep3_wr_cfrm_udc_reg)—Offset 7Ch 495
16.6.1.25OUT Endpoint 0 Control Register (ep0_out_ctrl_udc_reg)—Offset 200h 495
16.6.1.26OUT Endpoint 0 Status Register (ep0_out_sts_udc_reg)—Offset 204h 496
16.6.1.27OUT Endpoint 0 Receive Packet Frame Number Register (ep0_out_rpf_udc_reg)—Offset 208h 498
16.6.1.28OUT Endpoint 0 Buffer Size Register (ep0_out_bufsize_udc_reg)—Offset 20Ch 499
16.6.1.29OUT Endpoint 0 SETUP Buffer Pointer Register (ep0_subptr_udc_reg)—Offset 210h 499
16.6.1.30OUT Endpoint 0 Data Descriptor Pointer Register (ep0_out_desptr_udc_reg)—Offset 214h 500
16.6.1.31OUT Endpoint 0 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep0_rd_cfrm_udc_reg)—Offset 21Ch . 500
16.6.1.32OUT Endpoint 1 Control Register (ep1_out_ctrl_udc_reg)—Offset 220h 501
16.6.1.33OUT Endpoint 1 Status Register (ep1_out_sts_udc_reg)—Offset 224h 502
16.6.1.34OUT Endpoint 1 Receive Packet Frame Number Register (ep1_out_rpf_udc_reg)—Offset 228h 504
16.6.1.35OUT Endpoint 1 Buffer Size Register (ep1_out_bufsize_udc_reg)—Offset 22Ch 505
16.6.1.36OUT Endpoint 1 SETUP Buffer Pointer Register (ep1_subptr_udc_reg)—Offset 230h 505
16.6.1.37OUT Endpoint 1 Data Descriptor Pointer Register (ep1_out_desptr_udc_reg)—Offset 234h 506
16.6.1.38OUT Endpoint 1 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep1_rd_cfrm_udc_reg)—Offset 23Ch . 506
16.6.1.39OUT Endpoint 2 Control Register (ep2_out_ctrl_udc_reg)—Offset 240h 507
16.6.1.40OUT Endpoint 2 Status Register (ep2_out_sts_udc_reg)—Offset 244h 508
16.6.1.41OUT Endpoint 2 Receive Packet Frame Number Register (ep2_out_rpf_udc_reg)—Offset 248h 510
16.6.1.42OUT Endpoint 2 Buffer Size Register (ep2_out_bufsize_udc_reg)—Offset 24Ch 511
16.6.1.43OUT Endpoint 2 SETUP Buffer Pointer Register (ep2_subptr_udc_reg)—Offset 250h 511
16.6.1.44OUT Endpoint 2 Data Descriptor Pointer Register (ep2_out_desptr_udc_reg)—Offset 254h 512
16.6.1.45OUT Endpoint 2 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep2_rd_cfrm_udc_reg)—Offset 25Ch . 512
16.6.1.46OUT Endpoint 3 Control Register (ep3_out_ctrl_udc_reg)—Offset 260h 513
16.6.1.47OUT Endpoint 3 Status Register (ep3_out_sts_udc_reg)—Offset 264h 514
16.6.1.48OUT Endpoint 3 Receive Packet Frame Number Register (ep3_out_rpf_udc_reg)—Offset 268h 516

16.6.1.49 OUT Endpoint 3 Buffer Size Register (ep3_out_bufsize_udc_reg)—Offset 26Ch 517
16.6.1.50 OUT Endpoint 3 SETUP Buffer Pointer Register (ep3_subptr_udc_reg)—Offset 270h 517
16.6.1.51 OUT Endpoint 3 Data Descriptor Pointer Register (ep3_out_desptr_udc_reg)—Offset 274h 518
16.6.1.52 OUT Endpoint 3 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep3_rd_cfrm_udc_reg)—Offset 27Ch. 518
16.6.1.53 Device Configuration Register (d_cfg_udc_reg)—Offset 400h 519
16.6.1.54 Device Control Register (d_ctrl_udc_reg)—Offset 404h 520
16.6.1.55 Device Status Register (d_sts_udc_reg)—Offset 408h 522
16.6.1.56 Device Interrupt Register (d_intr_udc_reg)—Offset 40Ch 523
16.6.1.57 Device Interrupt Mask Register (d_intr_msk_udc_reg)—Offset 410h 524
16.6.1.58 Endpoints Interrupt Register (ep_intr_udc_reg)—Offset 414h 525
16.6.1.59 Endpoints Interrupt Mask Register (ep_intr_msk_udc_reg)—Offset 418h 525
16.6.1.60 Test Mode Register (test_mode_udc_reg)—Offset 41Ch 526
16.6.1.61 Product Release Number Register (revision_udc_reg)—Offset 420h 527
16.6.1.62 SETUP command address pointer register (udc_desc_addr_udc_reg)—Offset 500h 527
16.6.1.63 Physical Endpoint 0 Register (udc_ep_ne_udc_reg_0)—Offset 504h 528
16.6.1.64 Physical Endpoint 1 Register (udc_ep_ne_udc_reg_1)—Offset 508h 528
16.6.1.65 Physical Endpoint 2 Register (udc_ep_ne_udc_reg_2)—Offset 50Ch 529
16.6.1.66 Physical Endpoint 3 Register (udc_ep_ne_udc_reg_3)—Offset 510h 530
16.6.1.67 Physical Endpoint 4 Register (udc_ep_ne_udc_reg_4)—Offset 514h 531
16.6.1.68 Physical Endpoint 5 Register (udc_ep_ne_udc_reg_5)—Offset 518h 532
16.6.1.69 Physical Endpoint 6 Register (udc_ep_ne_udc_reg_6)—Offset 51Ch 532
16.6.1.70 RxFIFO Array[0-511] (udc_rx_fifo_reg_array[0-511])—Offset 800h, Count 512, Stride 4h 533
16.6.1.71 TxFIFO 0 Array[0-255] (udc_tx_fifo_reg_0_array[0-255])—Offset 1000h, Count 256, Stride 4h 534
16.6.1.72 TxFIFO 1 Array[0-255] (udc_tx_fifo_reg_1_array[0-255])—Offset 1400h, Count 256, Stride 4h 534
16.6.1.73 TxFIFO 2 Array[0-255] (udc_tx_fifo_reg_2_array[0-255])—Offset 1800h, Count 256, Stride 4h 534
16.6.1.74 TxFIFO 3 Array[0-255] (udc_tx_fifo_reg_3_array[0-255])—Offset 1C00h, Count 256, Stride 4h 535
16.6.2 USB EHCI 535
16.6.2.1 Host Controller Interface Version Number and Capability Registers Length (HCCAPBASE)—Offset 0h 536
16.6.2.2 Host Controller Structural Parameters (HCSPARAMS)—Offset 4h ... 536
16.6.2.3 Host Controller Capability Parameters (HCCPARAMS)—Offset 8h ... 538
16.6.2.4 USB Command (USBCMD)—Offset 10h 539
16.6.2.5 USB Status (USBSTS)—Offset 14h 541
16.6.2.6 USB Interrupt Enable (USBINTR)—Offset 18h 543
16.6.2.7 USB Frame Index (FRINDEX)—Offset 1Ch 544
16.6.2.8 4 Gigabyte Memory Segment Selector (CTRLDSSEGMENT)—Offset 20h 545
16.6.2.9 Periodic Frame List Base Address (PERIODICLISTBASE)—Offset 24h 545
16.6.2.10 Asynchronous List Address (ASYNCLISTADDR)—Offset 28h 546
16.6.2.11 Configure Flag (CONFIGFLAG)—Offset 50h 546
16.6.2.12 Port Status/Control[0-1] (PORTSC[0-1])—Offset 54h, Count 2, Stride 4h 547
16.6.2.13 Programmable Microframe Base Value (INSNREG00)—Offset 90h .. 550
16.6.2.14 Programmable Packet Buffer OUT/IN Thresholds (INSNREG01)—Offset 94h 551

16.6.2.15 Programmable Packet Buffer Depth (INSNREG02)—Offset 98h 551
16.6.2.16 Programmable Controller Settings (INSNREG03)—Offset 9Ch 552
16.6.2.17 Programmable Controller Settings (INSNREG04)—Offset A0h 553
16.6.2.18 UTMI Configuration (INSNREG05)—Offset A4h 554
16.6.3 USB OHCI 555
16.6.3.1 OHCI Revision (HCREVISION)—Offset 0h 556
16.6.3.2 Host Controller Control (HCCONTROL)—Offset 4h 556
16.6.3.3 Host Controller Command Status (HCCMDSTATUS)—Offset 8h 557
16.6.3.4 Host Controller Interrupt Status (HCINTRSTATUS)—Offset Ch 559
16.6.3.5 Host Controller Interrupt Enable (HCINTRENABLE)—Offset 10h 560
16.6.3.6 Host Controller Interrupt Disable (HCINTRDISABLE)—Offset 14h ... 561
16.6.3.7 Host Controller Communication Area (HCHCCA)—Offset 18h 562
16.6.3.8 Host Controller Current Isochronous or Interrupt Endpoint (HCPRDCURED)—Offset 1Ch 563
16.6.3.9 Host Controller Current First Control Endpoint (HCCTRLHEADED)—Offset 20h 563
16.6.3.10 Host Controller Current Control Endpoint (HCCTRLCURED)—Offset 24h 564
16.6.3.11 Host Controller First Bulk Endpoint (HCBULKHEADED)—Offset 28h . 564
16.6.3.12 Host Controller Current Bulk Endpoint (HCBULKCURED)—Offset 2Ch 565
16.6.3.13 Host Controller Last Completed Descriptor (HCDONEHEAD)—Offset 30h 566
16.6.3.14 Host Controller Frame Interval (HCFMINTERVAL)—Offset 34h 566
16.6.3.15 Host Controller Remaining Frame (HCFMREMAINING)—Offset 38h . 567
16.6.3.16 Host Controller Frame Number (HCFMNUMBER)—Offset 3Ch 568
16.6.3.17 Host Controller Periodic List Start (HCPERIODICSTART)—Offset 40h 569
16.6.3.18 Host Controller LS Threshold (HCLSTHRESHOLD)—Offset 44h 569
16.6.3.19 Host Controller Root Hub Descriptor A (HCRHDESPA)—Offset 48h .. 570
16.6.3.20 Host Controller Root Hub Descriptor B (HCRHDESPB)—Offset 4Ch .. 571
16.6.3.21 Host Controller Root Hub Status (HCRHSTATUS)—Offset 50h 572
16.6.3.22 Host Controller Root Hub Port Status (HCRHPORTSTS)—Offset 54h 573

17.0 SDIO/SD/eMMC 577
17.1 Signal Descriptions 577
17.2 Features 578
17.2.1 SDIO/SD/eMMC Features 578
17.2.2 SD 3.0/ SDIO 3.0 / eMMC 4.41 Interfaces 578
17.2.2.1 SD 3.0 Bus Topology 578
17.2.2.2 SDIO 3.0 Interface 579
17.2.2.3 eMMC Interface 580
17.2.3 SDIO/SD/eMMC Host Controller 580
17.2.3.1 SD DMA 581
17.3 References 581
17.4 Register Map 581
17.5 PCI Configuration Registers 582
17.5.1 Vendor ID (VENDOR_ID)—Offset 0h 583
17.5.2 Device ID (DEVICE_ID)—Offset 2h 584
17.5.3 Command Register (COMMAND_REGISTER)—Offset 4h 584
17.5.4 Status Register (STATUS)—Offset 6h 585
17.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 585
17.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 586
17.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh 586
17.5.8 Header Type (HEADER_TYPE)—Offset Eh 587
17.5.9 BIST (BIST)—Offset Fh 587
17.5.10 Base Address Register (BAR0)—Offset 10h 588
17.5.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 588

17.5.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 589
17.5.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh 589
17.5.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 589
17.5.15 Capabilities Pointer (CAP_POINTER)—Offset 34h 590
17.5.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch 590
17.5.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh 591
17.5.18 MIN_GNT (MIN_GNT)—Offset 3Eh 591
17.5.19 MAX_LAT (MAX_LAT)—Offset 3Fh 591
17.5.20 Capability ID (PM_CAP_ID)—Offset 80h 592
17.5.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 592
17.5.22 Power Management Capabilities (PMC)—Offset 82h 592
17.5.23 Power Management Control/Status Register (PMCSR)—Offset 84h 593
17.5.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 594
17.5.25 Power Management Data Register (DATA_REGISTER)—Offset 87h 594
17.5.26 Capability ID (MSI_CAP_ID)—Offset A0h 595
17.5.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 595
17.5.28 Message Control (MESSAGE_CTRL)—Offset A2h 595
17.5.29 Message Address (MESSAGE_ADDR)—Offset A4h 596
17.5.30 Message Data (MESSAGE_DATA)—Offset A8h 596
17.5.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 597
17.5.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 597
17.6 Memory Mapped Registers 598
17.6.1 SDMA System Address Register (SYS_ADR)—Offset 0h 599
17.6.2 Block Size Register (BLK_SIZE)—Offset 4h 600
17.6.3 Block Count Register (BLK_COUNT)—Offset 6h 601
17.6.4 Argument Register (ARGUMENT)—Offset 8h 602
17.6.5 Transfer Mode Register (TX_MODE)—Offset Ch 602
17.6.6 Command Register (CMD)—Offset Eh 604
17.6.7 Response Register 0 (RESPONSE0)—Offset 10h 605
17.6.8 Response Register 2 (RESPONSE2)—Offset 14h 606
17.6.9 Response Register 4 (RESPONSE4)—Offset 18h 606
17.6.10 Response Register 6 (RESPONSE6)—Offset 1Ch 607
17.6.11 Buffer Data Port Register (BUF_DATA_PORT)—Offset 20h 607
17.6.12 Present State Register (PRE_STATE)—Offset 24h 608
17.6.13 Host Control Register (HOST_CTL)—Offset 28h 612
17.6.14 Power Control Register (PWR_CTL)—Offset 29h 613
17.6.15 Block Gap Control Register (BLK_GAP_CTL)—Offset 2Ah 613
17.6.16 Wakeup Control Register (WAKEUP_CTL)—Offset 2Bh 615
17.6.17 Clock Control Register (CLK_CTL)—Offset 2Ch 616
17.6.18 Timeout Control Register (TIMEOUT_CTL)—Offset 2Eh 618
17.6.19 Software Reset Register (SW_RST)—Offset 2Fh 619
17.6.20 Normal Interrupt Status Register (NML_INT_STATUS)—Offset 30h 620
17.6.21 Error Interrupt Status Register (ERR_INT_STATUS)—Offset 32h 622
17.6.22 Normal Interrupt Status Enable (NRM_INT_STATUS_EN)—Offset 34h 624
17.6.23 Error Interrupt Status Enable Register (ERR_INT_STAT_EN)—Offset 36h 625
17.6.24 Normal Interrupt Signal Enable Register (NRM_INT_SIG_EN)—Offset 38h 626
17.6.25 Error Interrupt Signal Enable Register (ERR_INT_SIG_EN)—Offset 3Ah 628
17.6.26 Auto CMD12 Error Status Register (CMD12_ERR_STAT)—Offset 3Ch 629
17.6.27 Host Control 2 Register (HOST_CTRL_2)—Offset 3Eh 630
17.6.28 Capabilities Register (CAPABILITIES)—Offset 40h 631
17.6.29 Capabilities Register 2 (CAPABILITIES_2)—Offset 44h 633
17.6.30 Maximum Current Capabilities Register (MAX_CUR_CAP)—Offset 48h 634
17.6.31 Force Event Register for Auto CMD12 Error Status (FORCE_EVENT_CMD12_ERR_STAT)—Offset 50h 635

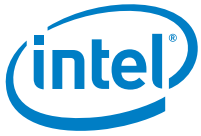
17.6.32 Force Event Register for Error Interrupt Status (FORCE_EVENT_ERR_INT_STAT)—Offset 52h 636
17.6.33 ADMA Error Status Register (ADMA_ERR_STAT)—Offset 54h 637
17.6.34 ADMA System Address Register (ADMA_SYS_ADDR)—Offset 58h 638
17.6.35 initialization Preset Values Register (3.3v or 1.8v) (PRESET_VALUE_0)—Offset 60h 639
17.6.36 Default Speed Preset Values Register (PRESET_VALUE_1)—Offset 62h 639
17.6.37 High Speed Preset Values Register (PRESET_VALUE_2)—Offset 64h 640
17.6.38 SDR12 Preset Values Register (PRESET_VALUE_3)—Offset 66h 641
17.6.39 SDR25 Preset Values Register (PRESET_VALUE_4)—Offset 68h 641
17.6.40 SDR50 Preset Values Register (PRESET_VALUE_5)—Offset 6Ah 642
17.6.41 SDR104 Preset Values Register (PRESET_VALUE_6)—Offset 6Ch 643
17.6.42 DDR50 Preset Values Register (PRESET_VALUE_7)—Offset 6Eh 643
17.6.43 Boot Time-out control register (BOOT_TIMEOUT_CTRL)—Offset 70h 644
17.6.44 Debug Selection Register (DEBUG_SEL)—Offset 74h 644
17.6.45 Shared Bus Control Register (SHARED_BUS)—Offset E0h 645
17.6.46 SPI Interrupt Support Register (SPI_INT_SUP)—Offset F0h 647
17.6.47 Slot Interrupt Status Register (SLOT_INT_STAT)—Offset FCh 647
17.6.48 Host Controller Version Register (HOST_CTRL_VER)—Offset FEh 648

18.0 High Speed UART 649
18.1 Signal Descriptions 649
18.2 Features 650
18.2.1 UART Function 650
18.2.2 Baud Rate Generator 650
18.3 Use 651
18.3.1 DMA Mode Operation 651
18.3.1.1 Receiver DMA 651
18.3.1.2 Transmitter DMA 652
18.3.2 FIFO Interrupt-Mode Operation 652
18.3.2.1 Receiver Interrupt 652
18.3.2.2 Transmitter Interrupt 652
18.3.3 FIFO Polled-Mode Operation 653
18.3.3.1 Receive Data Service 653
18.3.3.2 Transmit Data Service 653
18.3.4 Autoflow Control 653
18.3.4.1 RTS (UART Output) 653
18.3.4.2 CTS (UART Input) 654
18.4 Register Map 654
18.5 PCI Configuration Registers 655
18.5.1 Vendor ID (VENDOR_ID)—Offset 0h 655
18.5.2 Device ID (DEVICE_ID)—Offset 2h 656
18.5.3 Command Register (COMMAND_REGISTER)—Offset 4h 656
18.5.4 Status Register (STATUS)—Offset 6h 657
18.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 658
18.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 658
18.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh 659
18.5.8 Header Type (HEADER_TYPE)—Offset Eh 659
18.5.9 BIST (BIST)—Offset Fh 659
18.5.10 Base Address Register (BAR0)—Offset 10h 660
18.5.11 Base Address Register (BAR1)—Offset 14h 660
18.5.12 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 661
18.5.13 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 661
18.5.14 Subsystem ID (SUB_SYS_ID)—Offset 2Eh 662
18.5.15 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 662
18.5.16 Capabilities Pointer (CAP_POINTER)—Offset 34h 662

18.5.17 Interrupt Line Register (INTR_LINE)—Offset 3Ch 663
18.5.18 Interrupt Pin Register (INTR_PIN)—Offset 3Dh 663
18.5.19 MIN_GNT (MIN_GNT)—Offset 3Eh 664
18.5.20 MAX_LAT (MAX_LAT)—Offset 3Fh 664
18.5.21 Capability ID (PM_CAP_ID)—Offset 80h 664
18.5.22 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 665
18.5.23 Power Management Capabilities (PMC)—Offset 82h 665
18.5.24 Power Management Control/Status Register (PMCSR)—Offset 84h 666
18.5.25 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 667
18.5.26 Power Management Data Register (DATA_REGISTER)—Offset 87h 667
18.5.27 Capability ID (MSI_CAP_ID)—Offset A0h 667
18.5.28 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 668
18.5.29 Message Control (MESSAGE_CTRL)—Offset A2h 668
18.5.30 Message Address (MESSAGE_ADDR)—Offset A4h 668
18.5.31 Message Data (MESSAGE_DATA)—Offset A8h 669
18.5.32 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 669
18.5.33 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 670
18.6 Memory Mapped Registers 670
18.6.1 UART Registers 670
18.6.1.1 Receive Buffer / Transmit Holding / Divisor Latch Low (RBR_THR_DLL)—Offset 0h 671
18.6.1.2 Interrupt Enable / Divisor Latch High (IER_DLH)—Offset 4h 671
18.6.1.3 Interrupt Identification / FIFO Control (IIR_FCR)—Offset 8h 672
18.6.1.4 Line Control (LCR)—Offset Ch 674
18.6.1.5 MODEM Control (MCR)—Offset 10h 674
18.6.1.6 Line Status (LSR)—Offset 14h 675
18.6.1.7 MODEM Status (MSR)—Offset 18h 677
18.6.1.8 Scratchpad (SCR)—Offset 1Ch 678
18.6.1.9 UART Status (USR)—Offset 7Ch 679
18.6.1.10 Halt Transmission (HTX)—Offset A4h 679
18.6.1.11 DMA Software Acknowledge (DMASA)—Offset A8h 680
18.6.2 DMA Controller Registers 680
18.6.2.1 Channel 0 Source Address (SAR0)—Offset 0h 682
18.6.2.2 Channel 0 Destination Address (DAR0)—Offset 8h 682
18.6.2.3 Channel 0 Linked List Pointer (LLP0)—Offset 10h 683
18.6.2.4 Channel 0 Control LOWER (CTL0_L)—Offset 18h 683
18.6.2.5 Channel 0 Control UPPER (CTL0_U)—Offset 1Ch 685
18.6.2.6 Channel 0 Source Status (SSTAT0)—Offset 20h 686
18.6.2.7 Channel 0 Destination Status (DSTAT0)—Offset 28h 686
18.6.2.8 Channel 0 Source Status Address (SSTATAR0)—Offset 30h 687
18.6.2.9 Channel 0 Destination Status Address (DSTATAR0)—Offset 38h 687
18.6.2.10 Channel 0 Configuration LOWER (CFG0_L)—Offset 40h 688
18.6.2.11 Channel 0 configuration UPPER (CFG0_U)—Offset 44h 689
18.6.2.12 Channel 0 Source Gather (SGR0)—Offset 48h 690
18.6.2.13 Channel 0 Destination Scatter (DSR0)—Offset 50h 691
18.6.2.14 Channel 1 Source Address (SAR1)—Offset 58h 691
18.6.2.15 Channel 1 Destination Address (DAR1)—Offset 60h 692
18.6.2.16 Channel 1 Linked List Pointer (LLP1)—Offset 68h 692
18.6.2.17 Channel 1 Control LOWER (CTL1_L)—Offset 70h 693
18.6.2.18 Channel 1 Control UPPER (CTL1_U)—Offset 74h 695
18.6.2.19 Channel 1 Source Status (SSTAT1)—Offset 78h 696
18.6.2.20 Channel 1 Destination Status (DSTAT1)—Offset 80h 696
18.6.2.21 Channel 1 Source Status Address (SSTATAR1)—Offset 88h 697
18.6.2.22 Channel 1 Destination Status Address (DSTATAR1)—Offset 90h 697
18.6.2.23 Channel 1 Configuration LOWER (CFG1_L)—Offset 98h 698
18.6.2.24 Channel 1 configuration UPPER (CFG1_U)—Offset 9Ch 699
18.6.2.25 Channel 1 Source Gather (SGR1)—Offset A0h 700
18.6.2.26 Channel 1 Destination Scatter (DSR1)—Offset A8h 700

18.6.2.27 Raw Status for IntTfr Interrupt (RAW_TFR)—Offset 2C0h 701
18.6.2.28 Raw Status for IntBlock Interrupt (RAW_BLOCK)—Offset 2C8h 701
18.6.2.29 Raw Status for IntSrcTran Interrupt (RAW_SRC_TRAN)—Offset 2D0h 702
18.6.2.30 Raw Status for IntDstTran Interrupt (RAW_DST_TRAN)—Offset 2D8h 702
18.6.2.31 Raw Status for IntErr Interrupt (RAW_ERR)—Offset 2E0h 703
18.6.2.32 Status for IntTfr Interrupt (STATUS_TFR)—Offset 2E8h 703
18.6.2.33 Status for IntBlock Interrupt (STATUS_BLOCK)—Offset 2F0h 704
18.6.2.34 Status for IntSrcTran Interrupt (STATUS_SRC_TRAN)—Offset 2F8h 704
18.6.2.35 Status for IntDstTran Interrupt (STATUS_DST_TRAN)—Offset 300h 705
18.6.2.36 Status for IntErr Interrupt (STATUS_ERR)—Offset 308h 705
18.6.2.37 Mask for IntTfr Interrupt (MASK_TFR)—Offset 310h 706
18.6.2.38 Mask for IntBlock Interrupt (MASK_BLOCK)—Offset 318h 706
18.6.2.39 Mask for IntSrcTran Interrupt (MASK_SRC_TRAN)—Offset 320h 707
18.6.2.40 Mask for IntDstTran Interrupt (MASK_DST_TRAN)—Offset 328h 708
18.6.2.41 Mask for IntErr Interrupt (MASK_ERR)—Offset 330h 708
18.6.2.42 Clear for IntTfr Interrupt (CLEAR_TFR)—Offset 338h 709
18.6.2.43 Clear for IntBlock Interrupt (CLEAR_BLOCK)—Offset 340h 709
18.6.2.44 Clear for IntSrcTran Interrupt (CLEAR_SRC_TRAN)—Offset 348h ... 710
18.6.2.45 Clear for IntDstTran Interrupt (CLEAR_DST_TRAN)—Offset 350h ... 710
18.6.2.46 Clear for IntErr Interrupt (CLEAR_ERR)—Offset 358h 711
18.6.2.47 Combined Interrupt Status (STATUS_INT)—Offset 360h 711
18.6.2.48 Source Software Transaction Request (REQ_SRC_REG)—Offset 368h 712
18.6.2.49 Destination Software Transaction Request register (REQ_DST_REG)—Offset 370h 712
18.6.2.50 Source Single Transaction Request (SGL_REQ_SRC_REG)—Offset 378h 713
18.6.2.51 Destination Single Software Transaction Request (SGL_REQ_DST_REG)—Offset 380h 714
18.6.2.52 Source Last Transaction Request (LST_SRC_REG)—Offset 388h 714
18.6.2.53 Destination Single Transaction Request (LST_DST_REG)—Offset 390h 715
18.6.2.54 DMA Configuration (DMA_CFG_REG)—Offset 398h 715
18.6.2.55 Channel Enable (CH_EN_REG)—Offset 3A0h 716

19.0 I²C* Controller/GPIO Controller 717
19.1 I²C Controller 717
19.1.1 Signal Descriptions 717
19.1.2 Features 717
19.1.2.1 I²C* Protocol 717
19.1.2.2 I²C* Modes of Operation 718
19.1.2.3 Functional Description 718
19.1.3 Use 723
19.1.3.1 Master Mode Operation 723
19.1.3.2 Disabling I²C* Controller 723
19.1.4 References 724
19.2 GPIO Controller 724
19.2.1 Signal Descriptions 724
19.2.2 Features 724
19.3 Register Map 724
19.4 PCI Configuration Registers 725
19.4.1 Vendor ID (VENDOR_ID)—Offset 0h 726
19.4.2 Device ID (DEVICE_ID)—Offset 2h 727
19.4.3 Command Register (COMMAND_REGISTER)—Offset 4h 727
19.4.4 Status Register (STATUS)—Offset 6h 728
19.4.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 728
19.4.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 729



19.4.7 Latency Timer (LATENCY_TIMER)—Offset Dh 729
19.4.8 Header Type (HEADER_TYPE)—Offset Eh 730
19.4.9 BIST (BIST)—Offset Fh 730
19.4.10 Base Address Register (BAR0)—Offset 10h 731
19.4.11 Base Address Register (BAR1)—Offset 14h 731
19.4.12 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h 732
19.4.13 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 732
19.4.14 Subsystem ID (SUB_SYS_ID)—Offset 2Eh 733
19.4.15 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 733
19.4.16 Capabilities Pointer (CAP_POINTER)—Offset 34h 733
19.4.17 Interrupt Line Register (INTR_LINE)—Offset 3Ch 734
19.4.18 Interrupt Pin Register (INTR_PIN)—Offset 3Dh 734
19.4.19 MIN_GNT (MIN_GNT)—Offset 3Eh 735
19.4.20 MAX_LAT (MAX_LAT)—Offset 3Fh 735
19.4.21 Capability ID (PM_CAP_ID)—Offset 80h 735
19.4.22 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 736
19.4.23 Power Management Capabilities (PMC)—Offset 82h 736
19.4.24 Power Management Control/Status Register (PMCSR)—Offset 84h 737
19.4.25 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h 738
19.4.26 Power Management Data Register (DATA_REGISTER)—Offset 87h 738
19.4.27 Capability ID (MSI_CAP_ID)—Offset A0h 738
19.4.28 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 739
19.4.29 Message Control (MESSAGE_CTRL)—Offset A2h 739
19.4.30 Message Address (MESSAGE_ADDR)—Offset A4h 740
19.4.31 Message Data (MESSAGE_DATA)—Offset A8h 740
19.4.32 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 740
19.4.33 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 741
19.5 Memory Mapped Registers 741
19.5.1 I2C* Controller Memory Mapped Registers 741
19.5.1.1 Control Register (IC_CON)—Offset 0h 742
19.5.1.2 Master Target Address (IC_TAR)—Offset 4h 743
19.5.1.3 Data Buffer and Command (IC_DATA_CMD)—Offset 10h 744
19.5.1.4 Standard Speed Clock SCL High Count (IC_SS_SCL_HCNT)—Offset 14h 745
19.5.1.5 Standard Speed Clock SCL Low Count (IC_SS_SCL_LCNT)—Offset 18h 746
19.5.1.6 Fast Speed Clock SCL High Count (IC_FS_SCL_HCNT)—Offset 1Ch 746
19.5.1.7 Fast Speed Clock SCL Low Count (IC_FS_SCL_LCNT)—Offset 20h . 747
19.5.1.8 Interrupt Status (IC_INTR_STAT)—Offset 2Ch 747
19.5.1.9 Interrupt Mask (IC_INTR_MASK)—Offset 30h 749
19.5.1.10 Raw Interrupt Status (IC_RAW_INTR_STAT)—Offset 34h 750
19.5.1.11 Receive FIFO Threshold Level (IC_RX_TL)—Offset 38h 752
19.5.1.12 Transmit FIFO Threshold Level (IC_TX_TL)—Offset 3Ch 752
19.5.1.13 Clear Combined and Individual Interrupt (IC_CLR_INTR)—Offset 40h 753
19.5.1.14 Clear RX_UNDER Interrupt (IC_CLR_RX_UNDER)—Offset 44h 753
19.5.1.15 Clear RX_OVER Interrupt (IC_CLR_RX_OVER)—Offset 48h 754
19.5.1.16 Clear TX_OVER Interrupt (IC_CLR_TX_OVER)—Offset 4Ch 754
19.5.1.17 Clear RD_REQ Interrupt (IC_CLR_RD_REQ)—Offset 50h 755
19.5.1.18 Clear TX_ABRT Interrupt (IC_CLR_TX_ABRT)—Offset 54h 755
19.5.1.19 Clear ACTIVITY Interrupt (IC_CLR_ACTIVITY)—Offset 5Ch 756
19.5.1.20 Clear STOP_DET Interrupt (IC_CLR_STOP_DET)—Offset 60h 756
19.5.1.21 Clear START_DET Interrupt (IC_CLR_START_DET)—Offset 64h 757
19.5.1.22 Enable (IC_ENABLE)—Offset 6Ch 757
19.5.1.23 Status (IC_STATUS)—Offset 70h 758
19.5.1.24 Transmit FIFO Level (IC_TXFLR)—Offset 74h 759
19.5.1.25 Receive FIFO Level (IC_RXFLR)—Offset 78h 760

19.5.1.26SDA Hold (IC_SDA_HOLD)—Offset 7Ch 760
19.5.1.27Transmit Abort Source (IC_TX_ABRT_SOURCE)—Offset 80h..... 761
19.5.1.28Enable Status (IC_ENABLE_STATUS)—Offset 9Ch 762
19.5.1.29SS and FS Spike Suppression Limit (IC_FS_SPKLEN)—Offset A0h .. 763
19.5.2 GPIO Controller Memory Mapped Registers..... 763
19.5.2.1 Port A Data (GPIO_SWPORTA_DR)—Offset 0h 764
19.5.2.2 Port A Data Direction (GPIO_SWPORTA_DDR)—Offset 4h..... 764
19.5.2.3 Interrupt Enable (GPIO_INTEN)—Offset 30h 765
19.5.2.4 Interrupt Mask (GPIO_INTMASK)—Offset 34h 765
19.5.2.5 Interrupt Type (GPIO_INTTYPE_LEVEL)—Offset 38h..... 766
19.5.2.6 Interrupt Polarity (GPIO_INT_POLARITY)—Offset 3Ch 767
19.5.2.7 Interrupt Status (GPIO_INTSTATUS)—Offset 40h..... 767
19.5.2.8 Raw Interrupt Status (GPIO_RAW_INTSTATUS)—Offset 44h..... 768
19.5.2.9 Debounce Enable (GPIO_DEBOUNCE)—Offset 48h 768
19.5.2.10Clear Interrupt (GPIO_PORTA_EOI)—Offset 4Ch 769
19.5.2.11Port A External Port (GPIO_EXT_PORTA)—Offset 50h 770
19.5.2.12Synchronization Level (GPIO_LS_SYNC)—Offset 60h..... 770

20.0 SPI Interface 773
20.1 Signal Descriptions 773
20.2 Features..... 773
20.2.1 SPI Controller..... 773
20.2.1.1 Processor-Initiated Data Transfer 774
20.2.1.2 Data Format 774
20.2.1.3 FIFO Operation 775
20.2.1.4 Baud Rate Generation 775
20.3 Register Map..... 777
20.4 PCI Configuration Registers..... 778
20.4.1 Vendor ID (VENDOR_ID)—Offset 0h 778
20.4.2 Device ID (DEVICE_ID)—Offset 2h..... 779
20.4.3 Command Register (COMMAND_REGISTER)—Offset 4h..... 779
20.4.4 Status Register (STATUS)—Offset 6h..... 780
20.4.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h 781
20.4.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch 781
20.4.7 Latency Timer (LATENCY_TIMER)—Offset Dh..... 782
20.4.8 Header Type (HEADER_TYPE)—Offset Eh 782
20.4.9 BIST (BIST)—Offset Fh..... 782
20.4.10Base Address Register (BAR0)—Offset 10h..... 783
20.4.11Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h..... 784
20.4.12Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch 784
20.4.13Subsystem ID (SUB_SYS_ID)—Offset 2Eh..... 784
20.4.14Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h 785
20.4.15Capabilities Pointer (CAP_POINTER)—Offset 34h 785
20.4.16Interrupt Line Register (INTR_LINE)—Offset 3Ch 785
20.4.17Interrupt Pin Register (INTR_PIN)—Offset 3Dh 786
20.4.18MIN_GNT (MIN_GNT)—Offset 3Eh..... 786
20.4.19MAX_LAT (MAX_LAT)—Offset 3Fh 787
20.4.20Capability ID (PM_CAP_ID)—Offset 80h 787
20.4.21Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h 787
20.4.22Power Management Capabilities (PMC)—Offset 82h 788
20.4.23Power Management Control/Status Register (PMCSR)—Offset 84h..... 788
20.4.24PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h..... 789
20.4.25Power Management Data Register (DATA_REGISTER)—Offset 87h..... 789
20.4.26Capability ID (MSI_CAP_ID)—Offset A0h..... 790
20.4.27Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h 790
20.4.28Message Control (MESSAGE_CTRL)—Offset A2h..... 790
20.4.29Message Address (MESSAGE_ADDR)—Offset A4h..... 791

20.4.30 Message Data (MESSAGE_DATA)—Offset A8h 791
20.4.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh 792
20.4.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h 792
20.5 Memory Mapped Registers 793
20.5.1 SPI Control Register 0 (SSCR0)—Offset 0h 793
20.5.2 SPI Control Register 1 (SSCR1)—Offset 4h 794
20.5.3 SPI Status Register (SSSR)—Offset 8h 796
20.5.4 SPI Data Register (SSDR)—Offset 10h 797
20.5.5 DDS Clock Rate Register (DDS_RATE)—Offset 28h 798

21.0 Legacy Bridge 801
21.1 Features 801
21.2 Register Map 802
21.3 PCI Configuration Registers 803
21.3.1 PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h 804
21.3.2 PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h 804
21.3.3 PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h 805
21.3.4 PCI Miscellaneous Fields (PCI_MISC)—Offset Ch 805
21.3.5 PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch 806
21.3.6 GPIO Base Address (GBA)—Offset 44h 807
21.3.7 PM1_BLK Base Address (PM1BLK)—Offset 48h 807
21.3.8 GPE0_BLK Base Address (GPE0BLK)—Offset 4Ch 807
21.3.9 ACPI Control (ACTL)—Offset 58h 808
21.3.10 PIRQA, PIRQB, PIRQC and PIRQD Routing Control (PABCDRC)—Offset 60h 808
21.3.11 PIRQE, PIRQF, PIRQG and PIRQH Routing Control (PEFGHRC)—Offset 64h 810
21.3.12 Watch Dog Timer Base Address (WDTBA)—Offset 84h 810
21.3.13 BIOS Decode Enable (BCE)—Offset D4h 811
21.3.14 BIOS Control (BC)—Offset D8h 812
21.3.15 Root Complex Base Address (RCBA)—Offset F0h 813
21.4 Memory Mapped Registers 813
21.4.1 Root Complex Register Block 813
21.4.1.1 Root Complex Topology Capabilities List (RCTCL)—Offset 0h 814
21.4.1.2 Element Self Description (ESD)—Offset 4h 814
21.4.1.3 Interrupt Queue Agent 0 (IRQAGENT0)—Offset 3140h 815
21.4.1.4 Interrupt Queue Agent 1 (IRQAGENT1)—Offset 3142h 815
21.4.1.5 Interrupt Queue Agent 2 (IRQAGENT2)—Offset 3144h 816
21.4.1.6 Interrupt Queue Agent 3 (IRQAGENT3)—Offset 3146h 816
21.4.1.7 RTC Configuration (RC)—Offset 3400h 817
21.5 IO Registers 818
21.5.1 Fixed IO Registers 818
21.5.1.1 NMI Status and Control Register (NSC)—Offset 61h 818
21.5.1.2 NMI Enable and RTC Index Register (NMIE)—Offset 70h 819
21.5.1.3 Software SMI Control Port (SWSMICTL)—Offset B2h 819
21.5.1.4 Software SMI Status Port (SWSMISTS)—Offset B3h 820
21.5.1.5 Reset Control Register (RSTC)—Offset CF9h 820
21.5.2 ACPI GPE0 Block 821
21.5.2.1 GPE0 Status Register (GPE0STS)—Offset 0h 821
21.5.2.2 GPE0 Enable Register (GPE0EN)—Offset 4h 822
21.5.2.3 SMI Enable Register (SMIEN)—Offset 10h 822
21.5.2.4 SMI Status Register (SMISTS)—Offset 14h 824
21.5.2.5 General Purpose Event Control Register (GPEC)—Offset 18h 825
21.5.2.6 Power Management Configuration Core Well Register (PMCW)—Offset 28h 825
21.5.2.7 Power Management Configuration Suspend Well Register (PMSW)—Offset 2Ch 826

21.5.2.8 Power Management Configuration RTC Well Register (PMRW)—Offset 30h 826
21.5.3 ACPI PM1 Block 827
21.5.3.1 PM1 Status Register (PM1S)—Offset 0h 827
21.5.3.2 PM1 Enable Register (PM1E)—Offset 2h 828
21.5.3.3 PM1 Control Register (PM1C)—Offset 4h 829
21.5.3.4 Power Management 1 Timer Register (PM1T)—Offset 8h 830
21.6 Legacy GPIO 831
21.6.1 Signal Descriptions 831
21.6.2 Features 831
21.6.3 Use 831
21.6.4 Register Map 832
21.6.5 IO Mapped Registers 832
21.6.5.1 Core Well GPIO Enable (CGEN)—Offset 0h 833
21.6.5.2 Core Well GPIO Input/Output Select (CGIO)—Offset 4h 833
21.6.5.3 Core Well GPIO Level for Input or Output (CGLVL)—Offset 8h 834
21.6.5.4 Core Well GPIO Trigger Positive Edge Enable (CGTPE)—Offset Ch .. 834
21.6.5.5 Core Well GPIO Trigger Negative Edge Enable (CGTNE)—Offset 10h 834
21.6.5.6 Core Well GPIO GPE Enable (CGGPE)—Offset 14h 835
21.6.5.7 Core Well GPIO SMI Enable (CGSMI)—Offset 18h 835
21.6.5.8 Core Well GPIO Trigger Status (CGTS)—Offset 1Ch 836
21.6.5.9 Resume Well GPIO Enable (RGEN)—Offset 20h 836
21.6.5.10 Resume Well GPIO Input/Output Select (RGIO)—Offset 24h 837
21.6.5.11 Resume Well GPIO Level for Input or Output (RGLVL)—Offset 28h . 837
21.6.5.12 Resume Well GPIO Trigger Positive Edge Enable (RGTPE)—Offset 2Ch 837
21.6.5.13 Resume Well GPIO Trigger Negative Edge Enable (RGTNE)—Offset 30h 838
21.6.5.14 Resume Well GPIO GPE Enable (RGGPE)—Offset 34h 838
21.6.5.15 Resume Well GPIO SMI Enable (RGSMI)—Offset 38h 839
21.6.5.16 Resume Well GPIO Trigger Status (RGTS)—Offset 3Ch 839
21.6.5.17 Core Well GPIO NMI Enable (CGNMIEN)—Offset 40h 840
21.6.5.18 Resume Well GPIO NMI Enable (RGNMIEN)—Offset 44h 840
21.7 Legacy SPI Controller 841
21.7.1 Signal Descriptions 841
21.7.2 Features 841
21.7.3 Register Map 841
21.7.4 Legacy SPI Host Interface Registers 842
21.7.4.1 SPI Status (SPISTS)—Offset 3020h 843
21.7.4.2 SPI Control (SPICTL)—Offset 3022h 844
21.7.4.3 SPI Address (SPIADDR)—Offset 3024h 845
21.7.4.4 SPI Data 0 - Lower 32 Bits (SPID0_1)—Offset 3028h 846
21.7.4.5 SPI Data 0 - Upper 32 Bits (SPID0_2)—Offset 302Ch 846
21.7.4.6 SPI Data 1 - Lower 32 Bits (SPID1_1)—Offset 3030h 846
21.7.4.7 SPI Data 1 - Upper 32 Bits (SPID1_2)—Offset 3034h 847
21.7.4.8 SPI Data 2 - Lower 32 Bits (SPID2_1)—Offset 3038h 847
21.7.4.9 SPI Data 2 - Upper 32 Bits (SPID2_2)—Offset 303Ch 847
21.7.4.10 SPI Data 3 - Lower 32 Bits (SPID3_1)—Offset 3040h 848
21.7.4.11 SPI Data 3 - Upper 32 Bits (SPID3_2)—Offset 3044h 848
21.7.4.12 SPI Data 4 - Lower 32 Bits (SPID4_1)—Offset 3048h 848
21.7.4.13 SPI Data 4 - Upper 32 Bits (SPID4_2)—Offset 304Ch 849
21.7.4.14 SPI Data 5 - Lower 32 Bits (SPID5_1)—Offset 3050h 849
21.7.4.15 SPI Data 5 - Upper 32 Bits (SPID5_2)—Offset 3054h 850
21.7.4.16 SPI Data 6 - Lower 32 Bits (SPID6_1)—Offset 3058h 850
21.7.4.17 SPI Data 6 - Upper 32 Bits (SPID6_2)—Offset 305Ch 850
21.7.4.18 SPI Data 7 - Lower 32 Bits (SPID7_1)—Offset 3060h 851
21.7.4.19 SPI Data 7 - Upper 32 Bits (SPID7_2)—Offset 3064h 851
21.7.4.20 BIOS Base Address (BBAR)—Offset 3070h 851

21.7.4.21 Prefix Opcode Configuration (PREOP)—Offset 3074h 852
21.7.4.22 Opcode Type Configuration (OPTYPE)—Offset 3076h 853
21.7.4.23 Opcode Menu Configuration - Lower 32 Bits (OPMENU_1)—Offset 3078h 853
21.7.4.24 Opcode Menu Configuration - Upper 32 Bits (OPMENU_2)—Offset 307Ch 854
21.7.4.25 Protected BIOS Range 0 (PBR0)—Offset 3080h 855
21.7.4.26 Protected BIOS Range 1 (PBR1)—Offset 3084h 855
21.7.4.27 Protected BIOS Range 2 (PBR2)—Offset 3088h 856
21.8 8254 Programmable Interval Timer 858
21.8.1 Features 858
21.8.1.1 Counter 0, System Timer 858
21.8.1.2 Counter 1, Refresh Request Signal 858
21.8.1.3 Counter 2, Speaker Tone 858
21.8.2 Use 858
21.8.2.1 Timer Programming 858
21.8.2.2 Reading from the Interval Timer 859
21.8.3 Register Map 860
21.8.4 Timer I/O Registers 861
21.8.4.1 Counter 0 Interval Time Status Byte Format (C0TS)—Offset 40h ... 862
21.8.4.2 Counter 1 Interval Time Status Byte Format (C1TS)—Offset 41h ... 862
21.8.4.3 Counter 2 Interval Time Status Byte Format (C2TS)—Offset 42h ... 863
21.8.4.4 Timer Control Word Register (TCW)—Offset 43h 864
21.8.4.5 Counter 0 Counter Access Port Register (C0AP)—Offset 50h 864
21.8.4.6 Counter 1 Counter Access Port Register (C1AP)—Offset 51h 865
21.8.4.7 Counter 2 Counter Access Port Register (C2AP)—Offset 52h 865
21.9 High Precision Event Timer (HPET) 865
21.9.1 Features 866
21.9.1.1 Non-Periodic Mode - All Timers 866
21.9.1.2 Periodic Mode - Timer 0 Only 866
21.9.1.3 Interrupts 867
21.9.2 Register Map 867
21.9.3 Memory Mapped Registers 868
21.9.3.1 General Capabilities and ID Register - Lower 32 Bits (GCID_1)—Offset 0h 869
21.9.3.2 General Capabilities and ID Register - Upper 32 Bits (GCID_2)—Offset 4h 870
21.9.3.3 General Configuration (GC)—Offset 10h 870
21.9.3.4 General Interrupt Status Register (GIS)—Offset 20h 870
21.9.3.5 Main Counter Value Register - Lower 32 Bits (MCV_1)—Offset F0h 871
21.9.3.6 Main Counter Value Register - Upper 32 Bits (MCV_2)—Offset F4h 871
21.9.3.7 Timer 0 Config and Capabilities Register - Lower 32 Bits (T0C_1)—Offset 100h 872
21.9.3.8 Timer 0 Config and Capabilities Register - Upper 32 Bits (T0C_2)—Offset 104h 873
21.9.3.9 Timer 0 Comparator Value Register - Lower 32 Bits (T0CV_1)—Offset 108h 873
21.9.3.10 Timer 0 Comparator Value Register - Upper 32 Bits (T0CV_2)—Offset 10Ch 873
21.9.3.11 Timer 1 Config and Capabilities Register - Lower 32 Bits (T1C_1)—Offset 120h 874
21.9.3.12 Timer 1 Config and Capabilities Register - Upper 32 Bits (T1C_2)—Offset 124h 875
21.9.3.13 Timer 1 Comparator Value Register (T1CV_1)—Offset 128h 875
21.9.3.14 Timer 2 Config and Capabilities Register - Lower 32 Bits (T2C_1)—Offset 140h 875
21.9.3.15 Timer 2 Config and Capabilities Register - Upper 32 Bits (T2C_2)—Offset 144h 876
21.9.3.16 Timer 2 Comparator Value Register (T2CV_1)—Offset 148h 877

21.9.4 References....877
21.10 Real Time Clock (RTC)....877
21.10.1 Signal Descriptions....877
21.10.2 Features....878
21.10.2.1 Update Cycles....878
21.10.2.2 Interrupts....878
21.10.2.3 Lockable RAM Ranges....878
21.10.3 Register Map....879
21.10.4 I/O Registers....879
21.10.5 Indexed Registers....880
21.10.5.1 Offset 0Ah: Register A....881
21.10.5.2 Offset 0Bh: Register B - General Configuration....881
21.10.5.3 Offset 0Ch: Register C - Flag Register....882
21.10.5.4 Offset 0Dh: Register D - Flag Register....883
21.10.6 References....883
21.11 Interrupt Decoding & Routing....883
21.11.1 Features....883
21.11.1.1 Interrupt Decoder....883
21.11.1.2 Interrupt Router....884
21.12 8259 Programmable Interrupt Controllers (PIC)....884
21.12.1 Features....885
21.12.1.1 Interrupt Handling....885
21.12.1.2 Initialization Command Words (ICWx)....887
21.12.1.3 Operation Command Words (OCW)....888
21.12.1.4 Modes of Operation....888
21.12.1.5 Masking Interrupts....890
21.12.1.6 Steering of PCI Interrupts....890
21.12.2 Register Map....890
21.12.3 I/O Registers....891
21.12.3.1 Master Initialization Command Word 1 (MICW1)—Offset 20h....893
21.12.3.2 Master Initialization Command Word 2 (MICW2)—Offset 21h....894
21.12.3.3 Master Operational Control Word 2 (MOCW2)—Offset 24h....894
21.12.3.4 Master Initialization Command Word 3 (MICW3)—Offset 25h....895
21.12.3.5 Master Operational Control Word 3 (MOCW3)—Offset 28h....895
21.12.3.6 Master Initialization Command Word 4 (MICW4)—Offset 29h....896
21.12.3.7 Master Operational Control Word 1 (MOCW1)—Offset 2Dh....897
21.12.3.8 Slave Initialization Command Word 1 (SICW1)—Offset A0h....897
21.12.3.9 Slave Initialization Command Word 2 (SICW2)—Offset A1h....898
21.12.3.10 Slave Operational Control Word 2 (SoCW2)—Offset A4h....898
21.12.3.11 Slave Initialization Command Word 3 (SICW3)—Offset A5h....899
21.12.3.12 Slave Operational Control Word 3 (SoCW3)—Offset A8h....899
21.12.3.13 Slave Initialization Command Word 4 (SICW4)—Offset A9h....900
21.12.3.14 Slave Operational Control Word 1 (SoCW1)—Offset ADh....901
21.12.3.15 Master Edge/Level Control (ELCR1)—Offset 4D0h....901
21.12.3.16 Slave Edge/Level Control (ELCR2)—Offset 4D1h....901
21.13 I/O APIC....903
21.13.1 Features....903
21.13.2 Use....904
21.13.3 Unsupported Modes....904
21.13.4 Register Map....905
21.13.5 Memory Mapped Registers....905
21.13.5.1 Index Register (IDX)—Offset FEC00000h....906
21.13.5.2 Window Register (WDW)—Offset FEC00010h....906
21.13.5.3 End of Interrupt Register (EOI)—Offset FEC00040h....906
21.13.6 Index Registers....907
21.13.6.1 Identification Register (ID)—Offset 0h....907
21.13.6.2 Version Register (VS)—Offset 1h....908

21.13.6.3 Redirection Table Entry Lower (RTE[0-23]L)—Offset 10h - 3Eh 908
21.13.6.4 Redirection Table Entry Upper (RTE[0-23]U)—Offset 11h - 3Fh 909
21.14 Watchdog Timer 911
21.14.1 Features 911
21.14.2 Use 911
21.14.3 Register Map 912
21.14.4 I/O Mapped Registers 912
21.14.4.1 Preload Value 1 Register 0 (PV1R0)—Offset 0h 913
21.14.4.2 Preload Value 1 Register 1 (PV1R1)—Offset 1h 913
21.14.4.3 Preload Value 1 Register 2 (PV1R2)—Offset 2h 914
21.14.4.4 Preload Value 2 Register 0 (PV2R0)—Offset 4h 914
21.14.4.5 Preload Value 2 Register 1 (PV2R1)—Offset 5h 915
21.14.4.6 Preload Value 2 Register 2 (PV2R2)—Offset 6h 915
21.14.4.7 Reload Register 0 (RR0)—Offset Ch 915
21.14.4.8 Reload Register 1 (RR1)—Offset Dh 916
21.14.4.9 WDT Configuration Register (WDTCR)—Offset 10h 916
21.14.4.10 WDT Lock Register (WDTLR)—Offset 18h 917

22.0 Debug Port and JTAG/TAP 919
22.1 Signal Descriptions 919
22.2 Features 920
22.2.1 OpenOCD 920

Figures

1 Block Diagram 38
2 Intel® Quark SoC X1000 PCI View 42
3 Signals In Default System Pin List 46
4 PCI Express Transmitter Eye 74
5 PCI Express Receiver Eye 75
6 USB Rise and Fall Time 77
7 USB Jitter 77
8 USB EOP Width 78
9 SPI Interface Timing 79
10 SDIO Interface Timing 80
11 Measurement Points for Differential Clocks 82
12 Physical Address Space - Low DRAM & MMIO 90
13 Physical Address Space - MMIO 91
14 Physical Address Space - DOS DRAM 92
15 Physical Address Space - SMM Range 93
16 Bus 0 PCI Devices and Functions 96
17 Message Bus with PCI Space 97
18 SoC Platform Clocking 100
19 RTC Power Well Timing Diagrams 111
20 Power-Up Sequence without G2/G3 112
21 Power Up Sequence 114
22 eSRAM 4KB Page Mapping 124
23 eSRAM 512KB Page Mapping 125
24 Intel® Quark SoC X1000 Host Bridge Register Map 128
25 Register Map 236
26 PCI Express Register Map 262
27 Ethernet Register Map 309
28 USB Register Map 423
29 SD Memory Card Bus Topology 579
30 SDIO Card Bus Topology 579

31 eMMC Interface 580
32 SDIO/SD/eMMC Register Map 582
33 UART Data Transfer Flow 650
34 HSUART Register Map 654
35 Data Transfer on the I2C* Bus 719
36 START and STOP Conditions 719
37 7-Bit Address Format 720
38 10-bit Address Format 720
39 Master Transmitter Protocol 721
40 Master Receiver Protocol 722
41 START Byte Transfer 722
42 I2C*/GPIO Register Map 725
43 Generic SPI Waveform 774
44 SPI Register Map 777
45 Legacy Bridge Register Map 803
46 Legacy GPIO Register Map 832
47 Legacy SPI Register Map 842
48 8254 Timers Register Map 861
49 HPET Register Map 868
50 RTC Register Map 879
51 8259 Register Map 891
52 Detailed I/O APIC Block Diagram 903
53 MSI Address and Data 904
54 I/O APIC Register Map 905
55 Watchdog Timer Register Map 912

Tables

1 Industry Specifications 37
2 Component Identification 42
3 Intel® Quark SoC X1000 Device ID 43
4 I/O Power Well Definitions 47
5 Buffer Type Definitions 47
6 Default Buffer State Definitions 47
7 System Memory Signals 48
8 PCI Express* 2.0 Signals 48
9 Ethernet Interface Signals 49
10 USB 2.0 Interface Signals 49
11 Integrated Clock Interface Signals 50
12 SD/SDIO/MMC Signals 50
13 High Speed UART Signals 51
14 I2C* Signals 51
15 Legacy SPI Signals 52
16 SPI Signals 52
17 Real Time Clock (RTC) Interface Signals 53
18 Power Management Interface Signals 53
19 JTAG and Debug Interface Signals 54
20 Legacy Interface Signals 54
21 General Purpose I/O Signals 54
22 Power and Ground Pins 55
23 Hardware Straps 57
24 Alphabetical Ball Listing 59
25 Alphabetical Signal Listing 63
26 67

27 Intel® Quark SoC X1000 Absolute Maximum Voltage Ratings 69
28 Power Supply Rail Ranges 70
29 Maximum Supply Current: ICC Max 71
30 CFIO AC Characteristics 72
31 RTC DC Characteristics 72
32 PCI Express* 2.0 Differential Signal DC Characteristics 73
33 PCI Express* 2.0 Interface Timings 73
34 USB 2.0 Differential Signal DC Characteristics 75
35 USB 2.0 Interface Timings 76
36 Legacy SPI Interface Timings (20 MHz) 78
37 SPI0/1 Interface Timings (25 MHz) 78
38 SDIO Timing 79
39 Reference Clocks AC Characteristics 80
40 Fixed I/O Register Access Method Example (NSC Register) 83
41 Fixed Memory Mapped Register Access Method Example (IDX Register) 83
42 Referenced I/O Register Access Method Example (PM1S Register) 84
43 Memory Mapped Register Access Method Example (ESD Register) 84
44 PCI Register Access Method Example (PCI_DEVICE_VENDOR Register) 84
45 PCI CONFIG_ADDRESS Register (I/O PORT CF8h) Mapping 85
46 PCI Configuration Memory Bar Mapping 85
47 MCR Description 86
48 MCRX Description 86
49 Register Access Types and Definitions 87
50 Fixed Memory Ranges in the Legacy Bridge 93
51 Fixed I/O Ranges in the Legacy Bridge 94
52 Movable I/O Ranges Decoded by PCI Devices on the I/O Fabric 94
53 PCI Devices and Functions 95
54 Intel® Quark SoC X1000 Clock Inputs 101
55 Intel® Quark SoC X1000 Clock Outputs 101
56 General Power States for System 104
57 ACPI PM State Transition Rules 104
58 Processor Core/ States Support 105
59 Main Memory States 105
60 PCIe* States 105
61 G, S and C State Combinations 106
62 RTC Power Well Timing Parameters 111
63 S4/S5 to S0 Timing Parameters 115
64 Intel® Quark SoC X1000 S3 Wake Events 116
65 SoC Reset Events 117
66 Summary of PCI Configuration Registers—0/0/0 129
67 Summary of I/O Registers—PMBA 134
68 Summary of I/O Registers—SPI_DMA_BAR 136
69 Summary of Message Bus Registers—0x00 138
70 Summary of Message Bus Registers—0x03 142
71 Summary of Message Bus Registers—0x04 171
72 Summary of Message Bus Registers—0x05 180
73 Summary of Message Bus Registers—0x05 228
74 Summary of Message Bus Registers—0x31 229
75 Memory Signals 233
76 Supported DDR3 DRAM Devices 235
77 Supported DDR3 Memory Configurations 235
78 Summary of Message Bus Registers—0x01 236
79 PCI Express* 2.0 Signals 259
80 Possible Interrupts Generated From Events/Packets 260
81 Summary of PCI Configuration Registers—0/23/0 262

82 10/100 Ethernet Interface Signals 307
83 Summary of PCI Configuration Registers—0/20/6 309
84 Summary of Memory Mapped I/O Registers—BAR0 325
85 Signals 421
86 Summary of PCI Configuration Registers—0/20/2 423
87 Summary of PCI Configuration Registers—0/20/3 439
88 Summary of PCI Configuration Registers—0/20/4 457
89 Summary of Memory Mapped I/O Registers—BAR0 472
90 Summary of Memory Mapped I/O Registers—BAR0 535
91 Summary of Memory Mapped I/O Registers—BAR0 555
92 SDIO/SD/eMMC Interface Signals 577
93 SDIO/SD/eMMC Features 578
94 Summary of PCI Configuration Registers—0/20/0 582
95 Summary of Memory Mapped I/O Registers—BAR0 598
96 UART 0 Interface Signals 649
97 UART 1 Interface Signals 649
98 Baud Rates Achievable with Different DLAB Settings 650
99 Summary of PCI Configuration Registers—0/20/1 655
100 Summary of Memory Mapped I/O Registers—BAR0 670
101 Summary of Memory Mapped I/O Registers—BAR1 680
102 I^2C* Signals 717
103 I^2C* Definition of Bits in First Byte 720
104 GPIO Signals 724
105 Summary of PCI Configuration Registers—0/21/2 725
106 Summary of Memory Mapped I/O Registers—BAR0 741
107 Summary of Memory Mapped I/O Registers—BAR1 763
108 SPI Interface Signals 773
109 SPI Clock Frequency Settings 776
110 Summary of PCI Configuration Registers—0/21/0 778
111 Summary of Memory Mapped I/O Registers—BAR0 793
112 Summary of PCI Configuration Registers—0/31/0 803
113 Summary of Memory Mapped I/O Registers—RCBA 813
114 Summary of I/O Registers 818
115 Summary of I/O Registers—GPE0BLK 821
116 Summary of I/O Registers—PM1BLK 827
117 Legacy GPIO Signals 831
118 Summary of I/O Registers—GBA 832
119 Legacy SPI Signals 841
120 Summary of Memory Mapped I/O Registers—RCBA 843
121 Counter Operating Modes 859
122 Register Aliases 861
123 8254 Interrupt Mapping 867
124 Summary of Memory Mapped I/O Registers—0xFED00000 868
125 RTC Signals 878
126 I/O Registers Alias Locations 880
127 Indexed Registers 880
128 IRQAGENT Description 884
129 Interrupt Controller Connections 885
130 Interrupt Status Registers 886
131 Content of Interrupt Vector Byte 886
132 8259 I/O Registers Alias Locations 892
133 Summary of I/O Registers 892
134 I/O APIC Memory Mapped Registers 906
135 Index Registers 907
136 Summary of I/O Registers—WDTBA 912

137 Debug Port and JTAG/TAP Signals 919

Revision History

Date	Revision	Description
October 2013	001	Initial Public Release

1.0 Introduction

1.1 About This Manual

This document is intended for Original Equipment Manufacturers and BIOS vendors creating products based on the Intel® Quark SoC X1000 application processor.

Note: Throughout this document, SoC is used as a general term and refers to all Intel® Quark SoC X1000 SKUs, unless specifically noted otherwise.

This manual assumes a working knowledge of the vocabulary and principles of interfaces and architectures such as PCI Express*, USB, SDIO/MMC, and ACPI. Although some details of these features are described within this manual, refer to the individual industry specifications listed in Table 1 for the complete details.

All PCI buses, devices and functions in this manual are abbreviated using the following nomenclature; Bus:Device:Function. This manual abbreviates buses as B*n*, devices as D*n* and functions as F*n*. For example, Device 31 Function 0 is abbreviated as D31:F0, Bus 1 Device 8 Function 0 is abbreviated as B1:D8:F0. Generally, the bus number is not used, and can be considered to be Bus 0.

Table 1. Industry Specifications

Specification	Location
PCI Express Base Specification, Revision 2.0*	http://www.pcisig.com/specifications
PCI Local Bus Specification, Revision 2.3 (PCI)	http://www.pcisig.com/specifications
PCI Power Management Specification, Revision 1.2	http://www.pcisig.com/specifications
Universal Serial Bus Specification (USB), Revision 3.0	http://www.usb.org/developers/docs
Advanced Configuration and Power Interface, Version 4.0a (ACPI)	http://www.acpi.info/spec.htm
Enhanced Host Controller Interface Specification for Universal Serial Bus, Revision 1.0 (EHCI)	http://developer.intel.com/technology/usb/ehcispec.htm
IEEE 802.3 Fast Ethernet	http://standards.ieee.org/getieee802/
AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6)	http://T13.org (T13 1410D)
IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a	http://www.intel.com/content/www/us/en/software-developers/software-developers-hpet-spec-1-0a.html

1.2 Component Overview

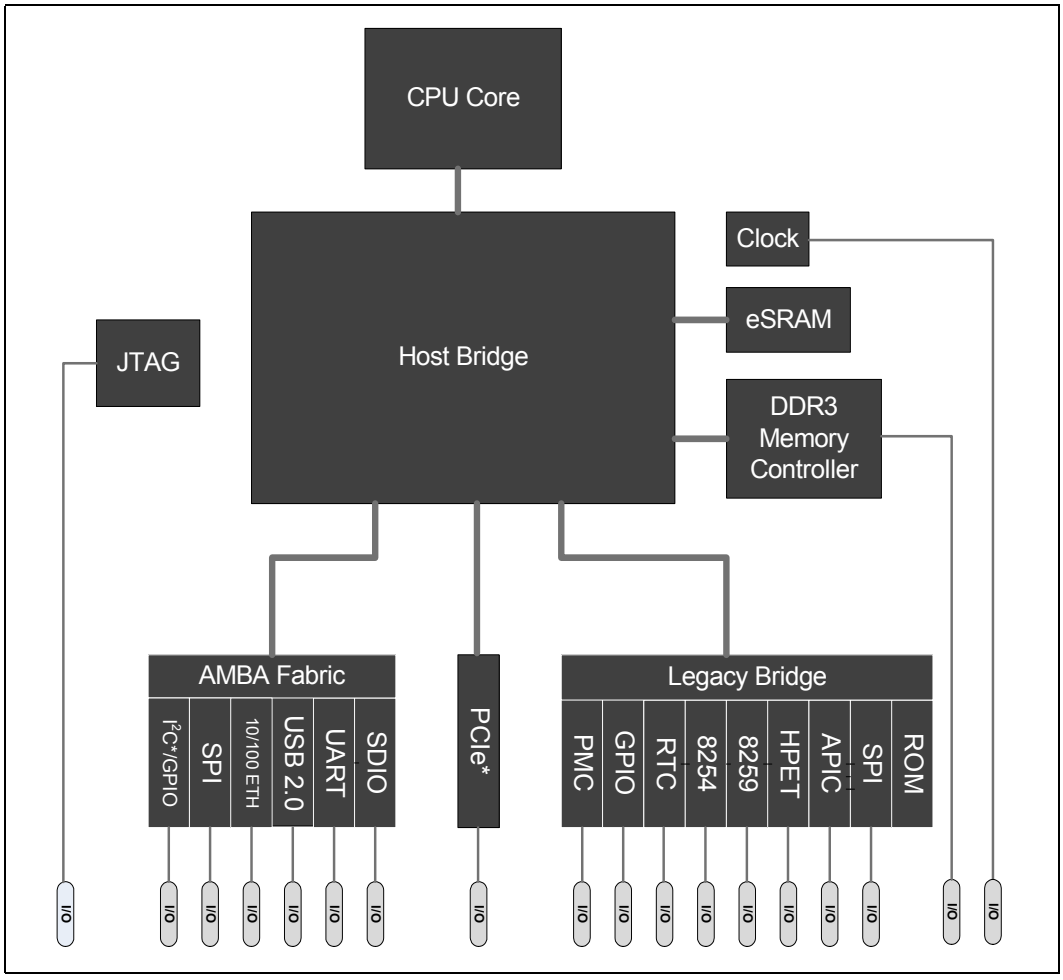
The Intel® Quark SoC X1000 processor is the next generation secure, low-power Intel® Architecture (IA) SoC for deeply embedded applications. The SoC integrates the Inte® Quark SoC X1000 Core plus all the required hardware components to run off-the-shelf operating systems and to leverage the vast x86 software ecosystem.

To enable secure applications, the SoC features an on-die Boot ROM that is used to establish a hardware Root of Trust (RoT). The immutable code located within the Boot ROM is used to initiate an iterative firmware authentication process ensuring only trusted code is executed when taking the platform out of reset.

To facilitate low-cost platforms with sensitive Bill of Material (BOM) requirements, all SoC clocks can be generated from a single crystal oscillator while all the required SoC voltage levels can be derived from a single commercial off-the-shelf (COTS) voltage regulator. In addition, the SoC provides an ECC-protected DRAM solution using only standard x8 DDR3 devices.

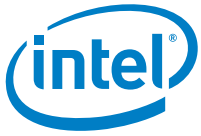
The SoC also features a 512 Kbyte on-die embedded SRAM (eSRAM) that can be configured to overlay regions of DRAM to provide low latency access to critical portions of system memory. For robustness, the contents of this on-die eSRAM are also ECC protected.

Figure 1. Block Diagram



1.2.1 SoC CPU Core Features

- 400 MHz maximum operating frequency
- Low power options to run at half or at quarter of maximum CPU frequency



- 32-bit address bus, 32-bit data bus
- 16 Kbyte shared instruction and data L1 cache.

1.2.2 System Memory Controller Features

- Single channel DDR3 memory controller with ECC support
- 16-bit data bus
- Supports up to two ranks total
- Supports DDR3 with 800 MT/s data rates
- x8 DRAM device data width
- 1 Gbit, 2 Gbit, and 4 Gbit DRAM device densities
- Total memory size from 128 Mbyte to 2 Gbyte
- Supports different physical mappings of bank addresses to optimize performance
- Out-of-order request processing to increase performance
- Aggressive power management to reduce power consumption
- Proactive page closing policies to close unused pages
- Supports soldered down DRAM devices

1.2.3 Embedded SRAM Features

- Low Latency 512 Kbyte on-die embedded SRAM
- Configurable to either overlay a 512 Kbyte block or overlay individual 4 Kbyte pages of system memory
- ECC protected

1.2.4 Power Management Features

- Supports ACPI 3.0 specification
- Supports C0, C1, and C2 processor power states
- Supports S0, S3, and S4/S5 system power states

1.2.5 Security Features

- On-die Boot ROM provides Hardware Root of Trust (RoT) for firmware authentication

1.2.6 PCI Express* Features

The SoC has two PCI Express* root ports, each supporting the PCI Express Base specification Rev 2.0 at a maximum of 2.5 GT/s data transfer rates. Each root port is configured as a x1 link.

- 128 Byte max payload size with the capability of splitting the request at 64 Byte granularity
- Software-Initiated Link Power Management (D1, D2, D3Hot, and L1 States)
- PME event generation

1.2.7 Ethernet Features

- 10 and 100 Mbps data transfer rates with RMII interface to communicate with an external Fast Ethernet PHY
- Full-duplex operation:
  - IEEE 802.3x flow control support
  - Optional forwarding of received pause control frames to the user application
- Half-duplex operation:
  - CSMA/CD Protocol support
- Flexible address filtering modes:
  - 64-bit Hash filter for multicast and unicast (DA) addresses
  - Option to pass all multicast addressed frames
  - Promiscuous mode to pass all frames without any filtering for network monitoring
  - Pass all incoming packets (as per filter) with a status report
- Programmable frame length to support Standard Ethernet frames with size up to 1522 bytes
- Enhanced Receive module for checking IPv4 header checksum and TCP, UDP, or ICMP checksum encapsulated in IPv4 or IPv6 datagrams (Type 2)
- Support Ethernet frame time stamping as described in IEEE 1588-2002 and IEEE 1588-2008. The 64-bit timestamps are given in the transmit or receive status of each frame.

1.2.8 USB2 Host Controller Features

- 2 host ports that support high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) operation
- EHCI and OHCI host controllers

1.2.9 USB2 Device Controller Features

- Single device port that supports high-speed (480 Mbps) and full-speed (12 Mbps) operation

1.2.10 SD/SDIO/eMMC Controller Features

- Host Controller provides a single port configurable as an SD, SDIO, or eMMC interface
- SD Clock Frequency up to 50 MHz
- Supports SD Host Controller Standard Specification 3.0
- Supports SDIO card specification 3.0
- Supports SD Memory Card Specification 3.0
- Supports SD Memory Card Security Specification 1.01
- Supports eMMC Specification 4.41

1.2.11 I^2C* Master Controller

- Two-wire I^2C serial bus interface

- Two I²C speeds supported: Standard (100 Kbit/s) and Fast (400 Kbit/s) data rates
- Fully asynchronous I²C clock signal
- Master I²C operation

1.2.12 GPIO Features

- 16 GPIO pins provided
- 6 GPIO pins remain active during S3 and can be used to wake the system from the Suspend state.
- Remaining 10 GPIO pins are powered during S0 state only and are not available in S3

1.2.13 SPI Master Controller

- Two SPI Master controllers
- One Chip Select per master controller
- Configurable SCLK frequency from 1 kHz up to 25 MHz

1.2.14 High Speed UART Controller with DMA

- Two 16550 compliant UART controllers
- Supported Baud rates from 300 to 2764800
- Integrated DMA capability with hardware flow control

1.2.15 Legacy Bridge

The Legacy Bridge is a collection of hardware blocks critical to implement an Intel Architecture compatible platform. Some of its key features are:

- A 20 MHz Serial Peripheral Interface (SPI) for Flash only - stores boot FW and system configuration data
- A Power Management Controller (PMC) that controls many of the power management features present in the SoC
- Legacy Bridge Components - Provides hardware blocks required to support legacy PC platform features. The legacy bridge components include the RTC, Interrupt Controllers, Timers and General Purpose I/Os (GPIO).

1.2.16 Package

The SoC is packaged in a Flip-Chip Ball Grid Array (FCBGA) package with 393 solder balls with 0.593 mm ball pitch. The package dimensions are 15mm x 15mm.

1.3 Component Identification

The Intel® Quark SoC X1000 stepping is identified by both:

- Processor Family/Model/Stepping returned by the CPUID instruction. This always returns 0x590 for SoC.
- Revision ID register of the Host Bridge, located at D0:F0. Reads of the register reflect the stepping.

Table 2. Component Identification

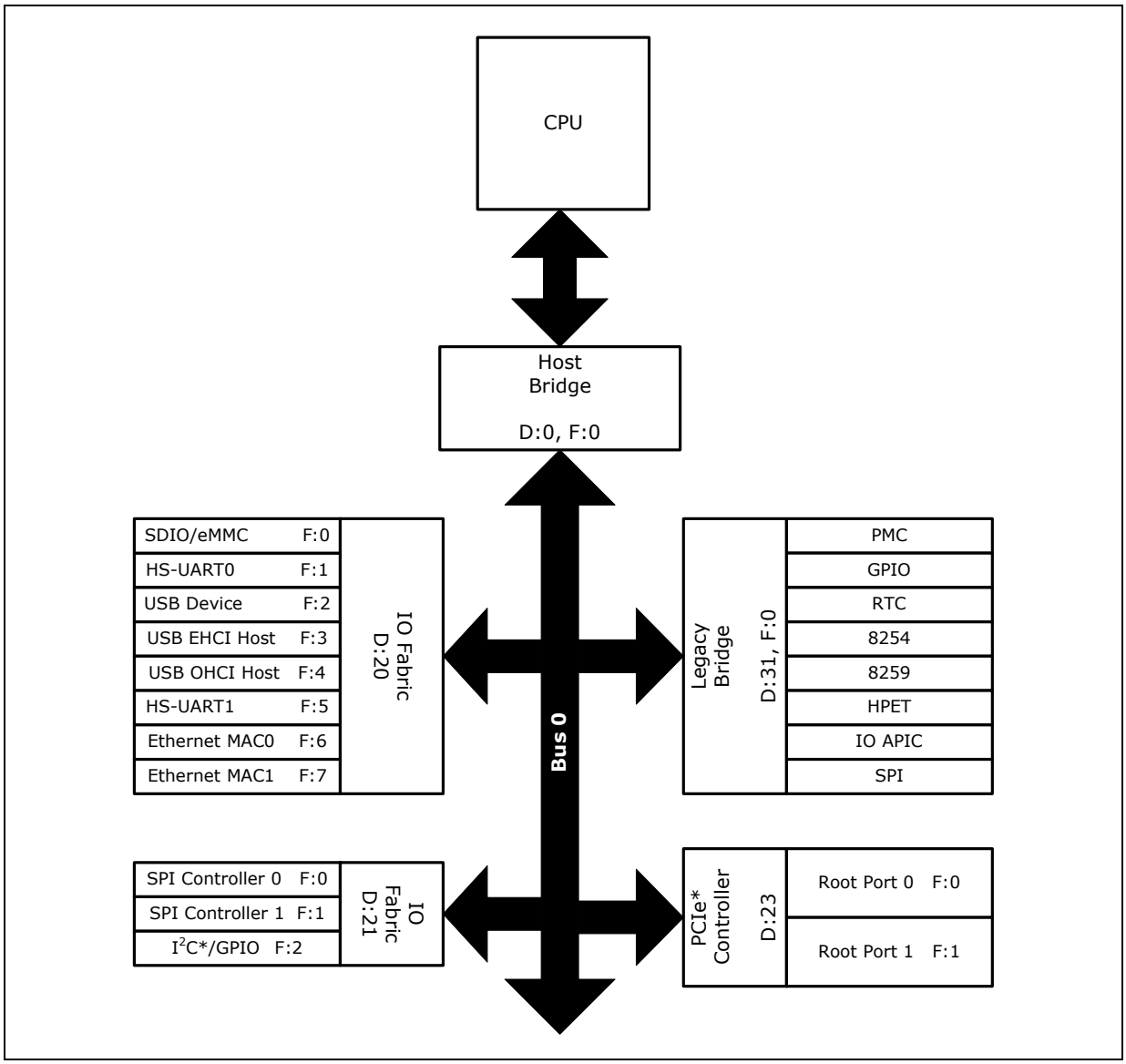
Vendor ID[1]	Device ID[2]	Revision ID[3]	Stepping
8086h	0958h	00h	A0h

Notes:

1. The Vendor ID corresponds to bits 15-0 of the Vendor ID Register located at offset 00-01h in the PCI configuration space of the device.
2. The Device ID corresponds to bits 15-0 of the Device ID Register located at offset 02-03h in the PCI configuration space of the device.
3. The Revision ID corresponds to bits 7-0 of the Revision ID Register located at offset 08h in the PCI configuration space of the device.

The SoC incorporates a variety of PCI functions as listed in Table 3. All devices reside on PCI Bus 0 as shown in Figure 2.

Figure 2. Intel® Quark SoC X1000 PCI View



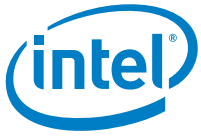
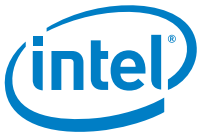


Table 3. Intel® Quark SoC X1000 Device ID

Device Function	Description	Device ID	A0 SRID
D0:F0	Host Bridge	0958h	00h
D31:F0	Legacy Bridge	095Eh	00h
D23:F0	PCIe* Root Port 0	11C3h	00h
D23:F1	PCIe* Root Port 1	11C4h	00h
D20:F0	SDIO / eMMC Controller	08A7h	10h
D20:F1	HS-UART 0	0936h	10h
D20:F2	USB 2.0 Device	0939h	10h
D20:F3	USB EHCI Host Controller	0939h	10h
D20:F4	USB OHCI Host Controller	093Ah	10h
D20:F5	HS-UART 1	0936h	10h
D20:F6	10/100 Ethernet MAC 0	0937h	10h
D20:F7	10/100 Ethernet MAC 1	0937h	10h
D21:F0	SPI Controller 0	0935h	10h
D21:F1	SPI Controller 1	0935h	10h
D21:F2	I^2C* Controller and GPIO Controller	0934h	10h

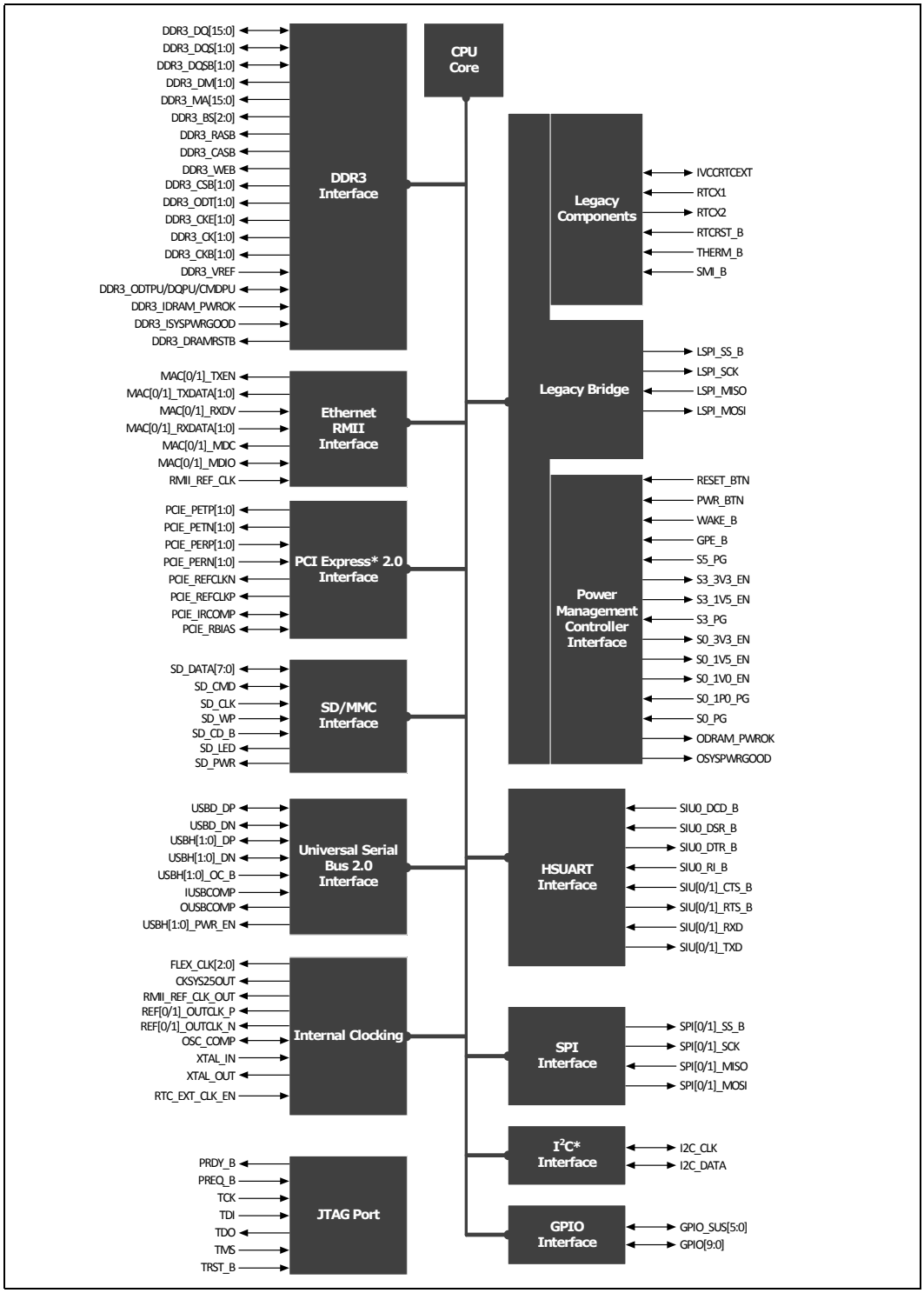
§ §



2.0 Physical Interfaces

Many interfaces contain physical pins. These groups of pins make up the physical interfaces. This chapter summarizes the physical interfaces.

Figure 3. Signals In Default System Pin List



2.1 Pin States Through Reset

This chapter describes the each signal state before, during, and directly after reset. Additionally, some signals have internal pull-up/pull-down termination resistors, and their values are also provided.

Table 4. I/O Power Well Definitions

Power Type	Power Well Description
CORE	Core I/O, and everything else uses the CORE power well.
SUS	Devices outside of memory that must remain on in the S3 state use the SUS power well.
RTC	Devices that must be on in the S4/S5 state use the RTC power well.

Table 5. Buffer Type Definitions

Buffer Type	Buffer Description
PCIe*	PCIe*, differential buffer type
SSTL-15	DDR3, 1.5V tolerant SSTL buffer type
DDI	DDR (TMDS, DP) 1.0V tolerant differential buffer type
CMOS[*Voltage*]	CMOS buffer type. [*Voltage*] can be of the following types: 1.05, 1.5, 1.8, and 3.3.
CMOS[*Voltage*]_OD	Open drain CMOS buffer type [*Voltage*] can be of the following types: 1.05, 1.5, 1.8 and 3.3.
Analog	Analog pins that do not have specific digital requirements. Often used for circuit calibration or monitoring.

Table 6. Default Buffer State Definitions

Buffer State	Description
High-Z	The SoC places this output in a high-impedance state. For inputs, external drivers are not expected.
Do Not Care	The state of the input (driven or tristated) does not affect the SoC. For outputs, it is assumed that the output buffer is in a high-impedance state.
V_{OH}	The SoC drives this signal high.
V_{OL}	The SoC drives this signal low.
Unknown	The SoC drives or expects an indeterminate value.
V_{IH}	The SoC expects/requires the signal to be driven high.
V_{IL}	The SoC expects/requires the signal to be driven low.
Pull-up	This signal is pulled high by a pull-up resistor (internal or external — internal value specified in “Term” column).
Pull-down	This signal is pulled low by a pull-down resistor (internal or external — internal value specified in “Term” column).
Running	The clock is toggling, or the signal is transitioning.
Off	The power plane for this signal is powered down. The SoC does not drive outputs, and inputs should not be driven to the SoC. (VSS on output)

2.2 System Memory Signals

See Section 6.0 for more details of the DDR3 interface signals. Termination not listed.

Table 7. System Memory Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
DDR3_BS[2:0]	O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_CASB	O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_RASB	O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_WEB	O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_MA[15:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_CK[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_CKB[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_CKE[1:0]	I/O	-	1.5V	SSTL-15	Off	V_{OL}	V_{OL}	V_{OL}
DDR3_CSB[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	V_{OH}
DDR3_ODT[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	V_{OL}
DDR3_DQ[15:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_DM[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_DQS[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_DQSB[1:0]	I/O	-	1.5V	SSTL-15	Off	High-Z	High-Z	High-Z
DDR3_IDRAM_PWROK	I	Ext	1.5V	CMOS-15	V_{IL}	V_{IH}	Pull-up	V_{IH}
DDR3_ISYSPWRGOOD	I	Ext	1.5V	CMOS-15	V_{IL}	V_{IL}	Pull-up	V_{IH}
DDR3_DRAMRSTB	O	-	1.5V	CMOS-15	Off	V_{OH}	V_{OL}	V_{OL}/V_{OH}(S3 Exit)
DDR3_VREF	I/O	-	1.5V	Reference	Off	Reference	Reference	Reference
DDR3_ODTPU	I/O	-	1.5V	Analog	Off	Analog	Analog	Analog
DDR3_DQPU	I/O	-	1.5V	Analog	Off	Analog	Analog	Analog
DDR3_CMDPU	I/O	-	1.5V	Analog	Off	Analog	Analog	Analog

2.3 PCI Express* 2.0 Signals

See "PCI Express* 2.0" on page 259 for more details of the interface signals.

Table 8. PCI Express* 2.0 Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
PCIE_REFCLKP	I	-	1.05V	PCIe	Off	Off	Running/ Unknown	Running/ Unknown
PCIE_REFCLKN	I	-	1.05V	PCIe	Off	Off	Running/ Unknown	Running/ Unknown
PCIE_PETP[1:0]	O	-	1.05V	PCIe	Off	Off	V_{OL}	V_{OL}
PCIE_PETN[1:0]	O	-	1.05V	PCIe	Off	Off	V_{OL}	V_{OL}
PCIE_PERP[1:0]	I	-	1.05V	PCIe	Off	Off	High-Z	High-Z
PCIE_PERN[1:0]	I	-	1.05V	PCIe	Off	Off	High-Z	High-Z
PCIE_IRCOMP	I/O	-	1.5V	Analog	Off	Off	Analog	Analog
PCIE_RBIAS	I	-	1.5V	Analog	Off	Off	Analog	Analog

2.4 Ethernet Interface Signals

See Chapter 15.0, "10/100 Mbps Ethernet" for more details of the Ethernet interface signals.

Table 9. Ethernet Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
RMII_REF_CLK	I	-	3.3V	CMOS3.3	Off	Off	Running/ Unknown	Running
MAC0_TXEN	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
MAC0_TXDATA[1:0]	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
MAC0_RXDV	I	20k(L)	3.3V	CMOS3.3	Off	Off	Pull-down	Pull-down
MAC0_RXDATA[1:0]	I	-	3.3V	CMOS3.3	Off	Off	Unknown	Unknown
MAC0_MDC	O	Ext	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
MAC0_MDIO	I/O	Ext	3.3V	CMOS3.3_OD	Off	Off	Pull-up	Pull-up
MAC1_TXEN	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
MAC1_TXDATA[1:0]	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
MAC1_RXDV	I	20k(L)	3.3V	CMOS3.3	Off	Off	Pull-down	Pull-down
MAC1_RXDATA[1:0]	I	-	3.3V	CMOS3.3	Off	Off	Unknown	Unknown
MAC1_MDC	O	Ext	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
MAC1_MDIO	I/O	Ext	3.3V	CMOS3.3_OD	Off	Off	Pull-up	Pull-up

2.5 USB 2.0 Interface Signals

See Chapter 16.0, "USB 2.0" for more details of the USB 2.0 interface signals.

Table 10. USB 2.0 Interface Signals (Sheet 1 of 2)

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
USBH0_OC_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
USBH1_OC_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
USBH0_PWR_EN	O	Ext	3.3V	CMOS3.3	Off	Off	Pull-down	Pull-down
USBH1_PWR_EN	O	Ext	3.3V	CMOS3.3	Off	Off	Pull-down	Pull-down
USBH0_DP	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USBH0_DN	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USBH1_DP	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USBH1_DN	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USBD_DP	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USBD_DN	I/O	-	3.3V	USB	Off	Off	High-Z	High-Z
USB_CLK96P	I	-	1.05V	USB	Off	Off	Running/ Unknown	Running/ Unknown

Table 10. USB 2.0 Interface Signals (Sheet 2 of 2)

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
USB_CLK96N	I	-	1.05V	USB	Off	Off	Running/ Unknown	Running/ Unknown
OUSBCOMP_P18	O	-	1.8V	Analog	Off	Off	Analog	Analog
IUSBCOMP_N18	I	-	1.8V	Analog	Off	Off	Analog	Analog

2.6 Integrated Clock Interface Signals

See Chapter 7.0, "Clocking" for more details of the Integrated Clock interface signals.

Table 11. Integrated Clock Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
XTAL_IN	I	-	1.05V	Analog	Running	Running	Running	Running
XTAL_OUT	O	-	1.05V	Analog	Running	Running	Running	Running
CKSYS25OUT	O	-	3.3V	CMOS3.3	Running	Running	Running	Running
REF0_OUTCLK_P	O	-	1.05V	CMOS1.05	V_{OL}	V_{OL}	V_{OL}	Running
REF0_OUTCLK_N	O	-	1.05V	CMOS1.05	V_{OL}	V_{OL}	V_{OL}	Running
REF1_OUTCLK_P	O	-	1.05V	CMOS1.05	V_{OL}	V_{OL}	V_{OL}	Running
REF1_OUTCLK_N	O	-	1.05V	CMOS1.05	V_{OL}	V_{OL}	V_{OL}	Running
FLEX0_CLK	O	-	3.3V	CMOS3.3	V_{OL}	V_{OL}	V_{OL}	Running
FLEX1_CLK	O	-	3.3V	CMOS3.3	V_{OL}	V_{OL}	V_{OL}	Running
FLEX2_CLK	O	-	3.3V	CMOS3.3	V_{OL}	V_{OL}	V_{OL}	Running
RMII_REF_CLK_OUT	O	-	3.3V	CMOS3.3	V_{OL}	V_{OL}	V_{OL}	Running
OSC_COMP	I	-	1.5V	Analog	Analog	Analog	Analog	Analog
HPLL_REFCLK_P	I	-	1.05V	CMOS1.05	Off	Off	Running/ Unknown	Running/ Unknown
HPLL_REFCLK_N	I	-	1.05V	CMOS1.05	Off	Off	Running/ Unknown	Running/ Unknown
PAD_BYPASS_CLK	I	-	1.05V	CMOS1.05	Off	Off	Running/ Unknown	Running/ Unknown

2.7 SDIO/SD/MMC Signals

See Chapter 17.0, "SDIO/SD/eMMC" for more details of the interface signals, including different options based on port configuration.

Table 12. SD/SDIO/MMC Signals (Sheet 1 of 2)

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
SD_DATA[7:0]	I/O	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SD_CMD	I/O	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SD_CLK	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}

Table 12. SD/SDIO/MMC Signals (Sheet 2 of 2)

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
SD_WP	I	20k(L)	3.3V	CMOS3.3	Off	Off	Pull-down	Pull-down
SD_CD_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SD_LED	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
SD_PWR	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}

2.8 High Speed UART Interface Signals

The SoC features two separate High Speed UARTs. However, only UART0 provides the Modem Control pins DCD, DSR, DTR and RI.

See Chapter 18.0, "High Speed UART" for more details of the HSUART interface signals.

Table 13. High Speed UART Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
SIU0_CTS_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU0_DCD_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU0_DSR_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU0_DTR_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SIU0_RI_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU0_RTS_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SIU0_RXD	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU0_TXD	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SIU1_CTS_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU1_RTS_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SIU1_RXD	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SIU1_TXD	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}

2.9 I^2C* Interface Signals

See Chapter 19.0, "I2C* Controller/GPIO Controller" for more details of the I^2C Interface signals.

Table 14. I^2C* Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
I2C_DATA	I/O	Ext	3.3V	CMOS3.3_OD	Off	Off	Pull-up	Pull-up
I2C_CLK	I/O	Ext	3.3V	CMOS3.3_OD	Off	Off	Pull-up	Pull-up

2.10 Legacy Serial Peripheral Interface (SPI) Signals

See Section 21.7 for more details of the SPI signals.

Table 15. Legacy SPI Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
LSPI_MOSI	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
LSPI_MISO	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
LSPI_SS_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
LSPI_SCK	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}

2.11 Serial Peripheral Interface (SPI)

See Chapter 20.0, "SPI Interface" for more details of the SPI signals.

Table 16. SPI Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
SPI0_MOSI	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
SPI0_MISO	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SPI0_SS_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SPI0_SCK	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
SPI1_MOSI	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}
SPI1_MISO	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SPI1_SS_B	O	-	3.3V	CMOS3.3	Off	Off	V_{OH}	V_{OH}
SPI1_SCK	O	-	3.3V	CMOS3.3	Off	Off	V_{OL}	V_{OL}

2.12 Real Time Clock (RTC) Interface Signals

See Section 21.10 for more details of the RTC interface signals.

Table 17. Real Time Clock (RTC) Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
RTCX1	I	-	<1V	Analog	Running	Running	Running	Running
RTCX2	O	-	<1V	Analog	Running	Running	Running	Running
IVCCRTCEXT	I/O	-	1.5V	Analog	Analog	Analog	Analog	Analog
RTCRST_B	I	-	3.3V	CMOS3.3	V_{iH}	V_{iH}	V_{iH}	V_{iH}
RTC_EXT_CLK_EN_B	I	-	3.3V	CMOS3.3	V_{iH}/V_{iL}	V_{iH}/V_{iL}	V_{iH}/V_{iL}	V_{iH}/V_{iL}

2.13 Power Management Signals

See Chapter 8.0, "Power Management" for more details of the Power Management interface signals.

Table 18. Power Management Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
PWR_BTN	I	-	3.3V	CMOS3.3	V_{iH}	V_{iH}	V_{iH}	V_{iH}
RESET_BTN	I	20k(H)	3.3V	CMOS3.3	Off	Pull-up	Pull-up	Pull-up
S5_PG	I	-	3.3V	CMOS3.3	V_{iH}	V_{iH}	V_{iH}	V_{iH}
S3_PG	I	-	3.3V	CMOS3.3	V_{iL}	V_{iH}	V_{iH}	V_{iH}
S0_PG	I	-	3.3V	CMOS3.3	V_{iL}	V_{iL}	V_{iH}	V_{iH}
S0_1P0_PG	I	-	3.3V	CMOS3.3	V_{iL}	V_{iL}	V_{iH}	V_{iH}
S3_3V3_EN	O	Ext	3.3V	CMOS3.3	V_{OL}	V_{OH}	V_{OH}	V_{OH}
S3_1V5_EN	O	Ext	3.3V	CMOS3.3	V_{OL}	V_{OH}	V_{OH}	V_{OH}
S0_3V3_EN	O	Ext	3.3V	CMOS3.3	Pull-down	V_{OL}	V_{OL}	V_{OiH}
S0_1V5_EN	O	Ext	3.3V	CMOS3.3	Pull-down	V_{oL}	V_{oH}	V_{oH}
S0_1P0_EN	O	Ext	3.3V	CMOS3.3	Pull-down	V_{oL}	V_{oH}	V_{oH}
ODRAM_PWROK	O	Ext	3.3V	CMOS3.3_OD	Pull-up	Pull-up	Pull-up	Pull-up
OSYSPWRGOOD	O	Ext	3.3V	CMOS3.3_OD	Pull-up	Pull-up	Pull-up	Pull-up
VNNSENSE	I/O	-	1.05V	Analog	Analog	Analog	Analog	Analog
VSSSENSE	I/O	-	GND	Analog	Analog	Analog	Analog	Analog

2.14 JTAG and Debug Interface Signals

See Chapter 22.0, "Debug Port and JTAG/TAP" for more details of the JTAG interface signals.

Table 19. JTAG and Debug Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
TCK	I	20k(L)	3.3V	CMOS3.3	Pull-down	Pull-down	Pull-down	Pull-down
TDI	I	20k(H)	3.3V	CMOS3.3	Pull-up	Pull-up	Pull-up	Pull-up
TDO	O	Ext	3.3V	CMOS3.3	Pull-up	Pull-up	Pull-up	Pull-up
TMS	I	20k(H)	3.3V	CMOS3.3	Pull-up	Pull-up	Pull-up	Pull-up
TRST_B	I	20k(H)	3.3V	CMOS3.3	Pull-up	Pull-up	Pull-up	Pull-up
PRDY_B	O	-	3.3V	CMOS3.3	Off	Off	V_{oH}	V_{oH}
PREQ_B	I	20k(H)	3.3V	CMOS3.3	Pull-up	Pull-up	Pull-up	Pull-up

2.15 Legacy Interface Signals

See Chapter 19.0, "I2C* Controller/GPIO Controller" and Chapter 21.0, "Legacy Bridge" for more details of the legacy interface signals.

Table 20. Legacy Interface Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
CLK14	I	-	3.3V	CMOS3.3	Off	Off	Unknown/ Running	Unknown/ Running
WAKE_B	I	20k(H)	3.3V	CMOS3.3_OD	Off	Pull-up	Pull-up	Pull-up
GPE_B	I	20k(H)	3.3V	CMOS3.3_OD	Off	Pull-up	Pull-up	Pull-up
THRM_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up
SMI_B	I	20k(H)	3.3V	CMOS3.3	Off	Off	Pull-up	Pull-up

2.16 General Purpose I/O Interface Signals

All GPIOs default to inputs. GPIO_SUS[5:0] are suspend well GPIOs and remain available in S3. The default buffer state of these GPIOs while in S3 and when entering S0 from S3 is configuration dependent.

Table 21. General Purpose I/O Signals

					Default Buffer State			
Signal Name	**Dir**	**Term**	**Power**	**Type**	**S4/S5**	**S3**	**Reset**	**Enter S0**
GPIO_SUS[5:0]	I/O	-	3.3V	CMOS3.3	Off	Unknown	V_{iX}	V_{iX}/Unknown
GPIO[9:0]	I/O	-	3.3V	CMOS3.3	Off	Off	V_{iX}	V_{iX}

2.17 Power And Ground Pins

Table 22. Power and Ground Pins (Sheet 1 of 2)

Signal Name	Nominal Voltage	Lowest Active State	Description/Notes
OVOUT_1P0_S5	1.0V	S5	S5 1.0V Rail Standby LDO Output
VCC1P0_S5			Standard 1.0V Rail for S5 Logic
VCCAICLKCB_1P0			ICLK Control Supply
VCCAICLKDBUFF_1P0			ICLK Differential Output Buffer Supply
VCCAICLKSSC1_1P0			ICLK SSC Supply
VCCDICLKDIG_1P0			ICLK Digital Supply
VCCAICLKSFR_1P5	1.5V		ICLK SFR (for Oscillator, IPLL)
OVOUT_1P8_S5	1.8V		S5 1.8V Rail Standby LDO Output
VCC1P8_S5			S5 1.8V CFIO Supply
VCC3P3_S5	3.3V		S5 3.3V Rail Standby LDO Input
VCCAICLKSE_3P3			ICLK Single Ended Output Buffer Supply
OVOUT_1P0_S3	1.0V	S3	S3 1.0V Rail Standby LDO Output
VCC1P0_S3			Standard 1.0V Rail for S3 logic
VCCCLKDDR_1P5	1.5V		DDR IO Clock Analog Thick Gate Isolated Quiet Supply
VCCDDR_1P5			DDR IO Analog Thick Gate Supply
OVOUT_1P8_S3	1.8V		S3 1.8V Rail Standby LDO Output
VCC1P8_S3			S3 1.8V CFIO Supply
VCCAUSB_1P8_S3			USB 1.8V Supply
VCC3P3_S3	3.3V		S3 3.3V Rail Standby LDO Input
VCC3P3_USB_S3			USB 3.3V Supply The USB PHY resides in the S3 power domain. However, from a functional point of view USB is only active in S0.

Table 22. Power and Ground Pins (Sheet 2 of 2)

Signal Name	Nominal Voltage	Lowest Active State	Description/Notes
VCC1P0_S0	1.0V	S0	Standard 1.0V Rail for HPLL (Host PLL) and USB Logic
VCCACLKDDR_1P0			DDR IO Digital Clock Isolated Quiet Supply
VCCADDR_1P0			DDR IO Digital Supply
VCCADLLDDR_1P0			DDR IO Digital Isolated Quiet Supply
VCCAPCIE_1P0			PCIe Analog Supply
VCCAVISA_1P0			VISA IO Analog Supply
VCCPLLDDR_1P0			DDR IO Digital PLL High Voltage
VNN			Default 1.0V Standard Cell Rail including Core and Uncore Logic
OVOUT_1P05_S0	1.05V		S0 1.05V Rail Standby LDO Output
VCCFHVSOC_1P05			SoC Fuses Supply
VCCFSOC_1P05			Fuse Digital Sensing
VCC1P5_S0	1.5V		PCIe Band-Gap Supply
VCCSFRPLLDDR_1P5			DDR IO PLL High Voltage
OVOUT_1P8_S0	1.8V		S0 1.8V Rail Standby LDO Output
OVOUT_1P8_SLDO			S0 1.8V Rail Standby LDO Output (Currently Unused)
VCC1P8_S0			S0 1.8V CFIO Supply
VCCAA_1P8			1.8V Analog Supply
VCCAUSB_1P8			USB 1.8V Analog Supply
VCC3P3_A	3.3V		S0 3.3V Rail Standby LDO Input (Currently Unused)
VCC3P3_S0			S0 3.3V Rail Standby LDO Input
VCCRTC3P3	3.3V	G3	RTC Well Supply To be active in G3, the source supply for VCCRTC3P3 must be a 3.0V coin cell battery or equivalent. If the target application does not require the RTC to be operational in G3, VCCRTC3P3 should be supplied from a 3.3V supply that is active in S5.
VSS	0V	-	Ground
VSSA_USB			USB Low-Noise Ground

2.18 Hardware Straps

The pins used for hardware straps are output pins in functional mode.Initially during a cold boot, these strap pins are configured as inputs. These pins remain inputs until the external pull up or pull down values are sampled during S0 Power OK. Once sampled, the pins are enabled as outputs only.

Table 23. Hardware Straps

Signal Name	Default	Strap Description
SPI0_MOSI	1b	Defines Memory Device Width 0b = x16 devices 1b = x8 devices
{SPI0_SCK, SPI1_MOSI}	11b	Defines Memory Device Density 00b = Reserved 01b = 1Gb 10b = 2Gb 11b = 4Gb
LSPI_MOSI	1b	Defines the Number of Ranks Enabled 0b = 1 Rank 1b = 2 Ranks
{MAC0_TXDATA[1], MAC0_TXDATA[0], MAC1_TXDATA[1]	000b	Frequency SKU Power Optimize Mode [2:1] CPU Clock/DDR Clock 00b = Reserved 01b = 400MHz/800MHz 10b = 200MHz/800MHz 11b = 100MHz/800MHz [0] 0b = Low Latency 1b = Low Power
MAC1_TXDATA[0]	0b	Remote Management Unit Firmware Base Address 0b = FFF0_0000h 1b = FFD0_0000h
{LSPI_SCK, SD_CLK}	00b	SDIO Slot Type 00b = Removable Card Slot 01b = Embedded Slot for One Device 10b = Shared Bus Slot 11b = Reserved
PWR_BTN	0b	Power Button Disable 0b = Power Button Disabled 1b = Power Button Enabled

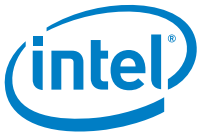
§ §

October 2013
Document Number: 329676-001US

3.0 Ballout and Package Information

Table 24. Alphabetical Ball Listing

A2	RESERVED
A4	S0_PG
A5	WAKE_B
A7	VSS
A10	VSS
A12	VSS
A14	VSS
A16	VSS
A18	VSS
A21	MAC0_TXDATA[0]
A23	VSS
A26	SD_DATA[4]
A28	VSS
A31	VSS
A32	VSS
A34	VSS
B1	RESERVED
B2	PREQ_B
B4	GPE_B
B6	I2C_DATA
B9	THRM_B
B11	MAC1_MDC
B13	MAC1_RXDATA[1]
B15	MAC1_TXDATA[1]
B17	MAC0_MDIO
B20	MAC0_RXDATA[1]
B22	SD_CMD
B24	SD_DATA[2]
B27	SD_DATA[6]
B29	VSS
B31	VSS
B33	VSS
B35	VSS
C1	VSS
C7	I2C_CLK
C10	SMI_B
C12	MAC1_TXEN
C14	MAC1_RXDV
C16	RMII_REF_CLK
C18	MAC0_RXDV
C21	MAC0_TXEN
C23	SD_DATA[0]
C26	SD_DATA[3]
C28	SD_DATA[7]
C35	VSS
D2	S0_1P0_PG
D5	VSS
D6	RESET_BTN
D9	CLK14
D11	MAC1_MDIO
D13	MAC1_RXDATA[0]
D15	MAC1_TXDATA[0]
D17	MAC0_MDC
D20	MAC0_RXDATA[0]
D22	MAC0_TXDATA[1]
D24	SD_DATA[1]
D27	SD_DATA[5]
D29	VSS
D31	VSS
D33	VSS
E1	VCCRTC_3P3
E4	S0_3V3_EN
E7	PRDY_B
E11	RTC_EXT_CLK_EN_B
E12	GPIO_SUS[1]
E15	GPIO_SUS[3]
E17	SD_LED
E19	SD_CD_B
E32	VSS
E35	VSS
F2	S0_1V5_EN
F4	S0_1V0_EN
F6	VSS
F30	VCC3P3_A
F32	VSS
F34	USBH0_PWR_EN
G1	VSS
G3	RTCX2
G7	S5_PG
G11	GPIO_SUS[0]
G12	VSS
G15	GPIO_SUS[4]
G17	SD_PWR
G19	SD_WP
G33	USBH1_PWR_EN
G35	USBH0_OC_B
H6	REF1_OUTCLK_N
H17	VSS



H30	VSSSENSE
J2	RTCX1
J4	S3_PG
J6	REF1_OUTCLK_P
J7	RTCRST_B
J11	IVCCRTCEXT
J12	GPIO_SUS[2]
J15	GPIO_SUS[5]
J19	SD_CLK
J25	VSS
J30	VNNSENSE
J32	USBH1_OC_B
J34	OUSBCOMP_P18
K1	S3_1V5_EN
K3	S3_3V3_EN
K11	OVOUT_1P8_S5
K13	VCC1P8_S5
K14	VCC3P3_S3
K16	VCC3P3_USB_S3
K18	VCC1P8_S3
K20	VCCAUSB_1P8_S3
K22	VCCAA_1P8
K24	OVOUT_1P8_SLDO
K33	IUSBCOMP_N18
K35	VSS
L2	PWR_BTN
L4	TRST_B
L32	RESERVED
L34	RESERVED
M1	VSS
M3	TCK
M5	VSS
M6	REF0_OUTCLK_N
M8	REF0_OUTCLK_P
M11	VSS
M13	VSS
M14	OVOUT_1P0_S3

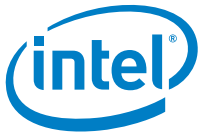
M16	VSS
M18	OVOUT_1P8_S3
M22	VSS
M28	VSS
M29	USBD_DP
M31	USBD_DN
M33	USB_CLK96N
M35	USB_CLK96P
N2	TMS
N4	TDI
N32	GPIO[0]
N34	GPIO[1]
P5	VCC3P3_S5
P7	VCCAICLKSE_3P3
P9	VSS
P11	VCCDICLKDIG_1P0
P13	VSS
P14	VCC1P0_S3
P16	VSSA_USB
P18	OVOUT_1P8_S0
P20	VCC1P8_S0
P22	VCCAUSB_1P8
P24	OVOUT_1P05_S0
P27	VSS
P29	USBH1_DP
P30	USBH1_DN
R1	TDO
R3	ODRAM_PWROK
R10	VCCAICLKSFR_1P5
R26	VCCFSOC_1P05
R33	GPIO[2]
R35	GPIO[3]
T2	OSYSPWRGOOD
T4	RMII_REF_CLK_OUT
T11	VCCAICLKCB_1P0
T13	VCC1P0_S5
T14	OVOUT_1P0_S5

T16	VSS
T18	VCCAVISA_1P0
T20	VCC1P8_S0
T22	VSS
T24	VCCFHVSOC_1P05
T32	GPIO[4]
T34	GPIO[5]
U1	VSS
U3	FLEX2_CLK
U33	GPIO[6]
U35	VSS
V2	FLEX1_CLK
V4	FLEX0_CLK
V5	HPLL_REFCLK_N
V7	HPLL_REFCLK_P
V9	VSS
V11	VCCAICLKDBUFF_1P0
V13	VCCAICLKSSC1_1P0
V14	VCC1P0_S0
V16	VCC1P0_S0
V18	VNN
V20	VNN
V22	VSS
V27	VSS
V29	USBH0_DN
V30	USBH0_DP
V32	GPIO[7]
V34	GPIO[8]
W1	VSS
W3	OSC_COMP
W5	RESERVED
W7	CKSYS25OUT
W9	TS_IREF_N
W27	VSS
W29	SIU0_DTR_B
W30	SIU0_RTS_B

W33	GPIO[9]
W35	SIU0_TXD
Y11	VSS
Y13	VSS
Y14	VCCAPCIE_1P0
Y16	VNN
Y18	VNN
Y20	VNN
Y22	VSS
AA2	XTAL_IN
AA4	RESERVED
AA10	RESERVED
AA26	VCC3P3_S0
AA32	SIU0_DSR_B
AA34	SIU0_RXD
AB1	XTAL_OUT
AB3	VSS
AB11	VSS
AB13	VCCACLKDDR_1P0
AB14	VCCADDR_1P0
AB16	VSS
AB18	VNN
AB20	VNN
AB24	VCC3P3_S0
AB33	SIU1_CTS_B
AB35	VSS
AC2	VSS
AC4	PAD_BYPASS_CLK
AC5	TS_TDC
AC7	TS_TDA
AC9	VSS
AC27	SIU0_DCD_B
AC29	VSS
AC30	VSS
AC32	SIU1_TXD
AC34	SIU1_RXD
AD1	PCIE_PETN_0

AD3	PCIE_PETP_0
AD5	VCC1P5_S0
AD6	VSS
AD8	VSS
AD11	RESERVED
AD13	VCCPLLDDR_1P0
AD14	VCCADLLDDR_1P0
AD16	VSS
AD18	VSS
AD20	VSS
AD24	VCC3P3_S0
AD28	SIU0_RI_B
AD29	SIU0_CTS_B
AD31	VCCSFRPLLDDR_1P5
AD33	SIU1_RTS_B
AD35	VSS
AE2	PCIE_REFCLKP
AE4	PCIE_REFCLKN
AE10	RESERVED
AE26	VSS
AE32	SPI0_SCK
AE34	SPI0_SS_B
AF11	RESERVED
AF13	VSS
AF20	VSS
AF22	VSS
AF24	VSS
AG1	VSS
AG3	VSS
AG6	DDR3_DQ[4]
AG30	DDR3_CK[1]
AG33	SPI0_MISO
AG35	SPI0_MOSI
AH2	PCIE_PETN_1
AH4	PCIE_PETP_1
AH7	DDR3_DQSB[0]
AH11	VSS

AH12	DDR3_DQ[6]
AH15	DDR3_DQ[3]
AH17	VSS
AH19	RESERVED
AH23	DDR3_DQPU
AH25	DDR3_ISYSPWRGOOD
AH29	DDR3_CKB[1]
AH32	SPI1_MOSI
AH34	SPI1_SCK
AJ1	PCIE_IRCOMP
AJ3	PCIE_RBIAS
AJ6	DDR3_DQ[5]
AJ30	DDR3_CK[0]
AJ33	SPI1_SS_B
AJ35	VSS
AK2	PCIE_PERP_1
AK4	PCIE_PERN_1
AK7	DDR3_DQS[0]
AK11	DDR3_DQ[1]
AK12	DDR3_DQ[7]
AK15	VSS
AK17	RESERVED
AK19	RESERVED
AK23	DDR3_CMDPU
AK25	DDR3_IDRAM_PWROK
AK29	VSS
AK32	LSPI_MOSI
AK34	SPI1_MISO
AL6	VSS
AL11	DDR3_DQ[0]
AL12	DDR3_DM[0]
AL15	DDR3_DQ[2]
AL17	RESERVED
AL19	DDR3_ODTPU
AL23	VSS
AL25	DDR3_DRAMRSTB



AL30	DDR3_CKB[0]
AM1	VSS
AM2	PCIE_PERP_0
AM4	VSS
AM7	VSS
AM29	DDR3_VREF
AM32	VSS
AM33	LSPI_SS_B
AM35	LSPI_SCK
AN1	PCIE_PERN_0
AN5	DDR3_DQ[13]
AN6	DDR3_DQS[1]
AN9	VSS
AN11	DDR3_CKE[0]
AN13	VSS
AN15	DDR3_MA[11]
AN17	DDR3_MA[7]
AN20	DDR3_MA[5]
AN22	DDR3_MA[1]
AN24	DDR3_MA[10]
AN27	VSS
AN29	DDR3_CASB
AN31	DDR3_CSB[0]
AN35	LSPI_MISO
AP2	VSS
AP4	DDR3_DQ[9]
AP7	DDR3_DM[1]
AP10	DDR3_DQ[11]
AP12	DDR3_CKE[1]
AP14	DDR3_MA[14]
AP16	DDR3_MA[9]
AP18	VSS
AP21	DDR3_MA[3]
AP23	VSS
AP26	DDR3_BS[0]
AP28	DDR3_WEB
AP31	DDR3_CSB[1]

AP33	VSS
AR1	RESERVED
AR6	DDR3_DQSB[1]
AR9	DDR3_DQ[10]
AR11	DDR3_DQ[15]
AR13	DDR3_BS[2]
AR15	DDR3_MA[12]
AR17	DDR3_MA[8]
AR20	DDR3_MA[6]
AR22	DDR3_MA[2]
AR24	DDR3_MA[0]
AR27	DDR3_RASB
AR29	DDR3_MA[13]
AR35	RESERVED
AT2	RESERVED
AT3	DDR3_DQ[8]
AT5	DDR3_DQ[12]
AT7	VSS
AT10	DDR3_DQ[14]
AT12	VCCDDR_1P5
AT14	DDR3_MA[15]
AT16	VCCDDR_1P5
AT18	VCCDDR_1P5
AT21	DDR3_MA[4]
AT23	VCCDDR_1P5
AT26	DDR3_BS[1]
AT28	VCCCLKDDR_1P5
AT31	DDR3_ODT[0]

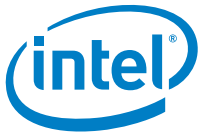
AT32	DDR3_ODT[1]
AT34	RESERVED

NOTE: The following balls are No Connect (NC):

M26
E25
J23
M24
Y24
G25
AF16
G23
M20
AD22
AF14
AB22
V24
E23
G29
AF18
J29
E29

Table 25. Alphabetical Signal Listing

CKSYS25OUT	W7
CLK14	D9
DDR3_BS[0]	AP26
DDR3_BS[1]	AT26
DDR3_BS[2]	AR13
DDR3_CASB	AN29
DDR3_CK[0]	AJ30
DDR3_CK[1]	AG30
DDR3_CKB[0]	AL30
DDR3_CKB[1]	AH29
DDR3_CKE[0]	AN11
DDR3_CKE[1]	AP12
DDR3_CMDPU	AK23
DDR3_CSB[0]	AN31
DDR3_CSB[1]	AP31
DDR3_DM[0]	AL12
DDR3_DM[1]	AP7
DDR3_DQ[0]	AL11
DDR3_DQ[1]	AK11
DDR3_DQ[10]	AR9
DDR3_DQ[11]	AP10
DDR3_DQ[12]	AT5
DDR3_DQ[13]	AN5
DDR3_DQ[14]	AT10
DDR3_DQ[15]	AR11
DDR3_DQ[2]	AL15
DDR3_DQ[3]	AH15
DDR3_DQ[4]	AG6
DDR3_DQ[5]	AJ6
DDR3_DQ[6]	AH12
DDR3_DQ[7]	AK12
DDR3_DQ[8]	AT3
DDR3_DQ[9]	AP4
DDR3_DQPU	AH23
DDR3_DQS[0]	AK7
DDR3_DQS[1]	AN6
DDR3_DQSB[0]	AH7
DDR3_DQSB[1]	AR6
DDR3_DRAMRSTB	AL25
DDR3_IDRAM_PWROK	AK25
DDR3_ISYSPWRGOOD	AH25
DDR3_MA[0]	AR24
DDR3_MA[1]	AN22
DDR3_MA[10]	AN24
DDR3_MA[11]	AN15
DDR3_MA[12]	AR15
DDR3_MA[13]	AR29
DDR3_MA[14]	AP14
DDR3_MA[15]	AT14
DDR3_MA[2]	AR22
DDR3_MA[3]	AP21
DDR3_MA[4]	AT21
DDR3_MA[5]	AN20
DDR3_MA[6]	AR20
DDR3_MA[7]	AN17
DDR3_MA[8]	AR17
DDR3_MA[9]	AP16
DDR3_ODT[0]	AT31
DDR3_ODT[1]	AT32
DDR3_ODTPU	AL19
DDR3_RASB	AR27
DDR3_VREF	AM29
DDR3_WEB	AP28
FLEX0_CLK	V4
FLEX1_CLK	V2
FLEX2_CLK	U3
GPE_B	B4
GPIO_SUS[0]	G11
GPIO_SUS[1]	E12
GPIO_SUS[2]	J12
GPIO_SUS[3]	E15
GPIO_SUS[4]	G15
GPIO_SUS[5]	J15
GPIO[0]	N32
GPIO[1]	N34
GPIO[2]	R33
GPIO[3]	R35
GPIO[4]	T32
GPIO[5]	T34
GPIO[6]	U33
GPIO[7]	V32
GPIO[8]	V34
GPIO[9]	W33
HPLL_REFCLK_N	V5
HPLL_REFCLK_P	V7
I2C_CLK	C7
I2C_DATA	B6
IUSBCOMP_N18	K33
IVCCRTCEXT	J11
LSPI_MISO	AN35
LSPI_MOSI	AK32
LSPI_SCK	AM35
LSPI_SS_B	AM33
MAC0_MDC	D17
MAC0_MDIO	B17
MAC0_RXDATA[0]	D20
MAC0_RXDATA[1]	B20
MAC0_RXDV	C18
MAC0_TXDATA[0]	A21
MAC0_TXDATA[1]	D22
MAC0_TXEN	C21
MAC1_MDC	B11
MAC1_MDIO	D11
MAC1_RXDATA[0]	D13
MAC1_RXDATA[1]	B13
MAC1_RXDV	C14
MAC1_TXDATA[0]	D15



MAC1_TXDATA[1]	B15
MAC1_TXEN	C12
ODRAM_PWROK	R3
OSC_COMP	W3
OSYSPWRGOOD	T2
OUSBCOMP_P18	J34
OVOUT_1P0_S3	M14
OVOUT_1P0_S5	T14
OVOUT_1P05_S0	P24
OVOUT_1P8_S0	P18
OVOUT_1P8_S3	M18
OVOUT_1P8_S5	K11
OVOUT_1P8_SLDO	K24
PAD_BYPASS_CLK	AC4
PCIE_IRCOMP	AJ1
PCIE_PERN_0	AN1
PCIE_PERN_1	AK4
PCIE_PERP_0	AM2
PCIE_PERP_1	AK2
PCIE_PETN_0	AD1
PCIE_PETN_1	AH2
PCIE_PETP_0	AD3
PCIE_PETP_1	AH4
PCIE_RBIAS	AJ3
PCIE_REFCLKN	AE4
PCIE_REFCLKP	AE2
PRDY_B	E7
PREQ_B	B2
PWR_BTN	L2
REF0_OUTCLK_N	M6
REF0_OUTCLK_P	M8
REF1_OUTCLK_N	H6
REF1_OUTCLK_P	J6
RESERVED	AR35
RESERVED	AT34
RESERVED	AH19
RESERVED	AK19

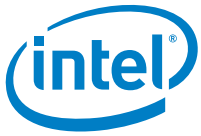
RESERVED	AL17
RESERVED	AK17
RESERVED	AT2
RESERVED	AR1
RESERVED	AD11
RESERVED	AF11
RESERVED	A2
RESERVED	B1
RESERVED	AA10
RESERVED	AE10
RESERVED	AA4
RESERVED	W5
RESERVED	L32
RESERVED	L34
RESET_BTN	D6
RMII_REF_CLK	C16
RMII_REF_CLK_OUT	T4
RTC_EXT_CLK_EN_B	E11
RTCRST_B	J7
RTCX1	J2
RTCX2	G3
S0_1P0_PG	D2
S0_1V0_EN	F4
S0_1V5_EN	F2
S0_3V3_EN	E4
S0_PG	A4
S3_1V5_EN	K1
S3_3V3_EN	K3
S3_PG	J4
S5_PG	G7
SD_CD_B	E19
SD_CLK	J19
SD_CMD	B22
SD_DATA[0]	C23
SD_DATA[1]	D24
SD_DATA[2]	B24
SD_DATA[3]	C26

SD_DATA[4]	A26
SD_DATA[5]	D27
SD_DATA[6]	B27
SD_DATA[7]	C28
SD_LED	E17
SD_PWR	G17
SD_WP	G19
SIU0_CTS_B	AD29
SIU0_DCD_B	AC27
SIU0_DSR_B	AA32
SIU0_DTR_B	W29
SIU0_RI_B	AD28
SIU0_RTS_B	W30
SIU0_RXD	AA34
SIU0_TXD	W35
SIU1_CTS_B	AB33
SIU1_RTS_B	AD33
SIU1_RXD	AC34
SIU1_TXD	AC32
SMI_B	C10
SPI0_MISO	AG33
SPI0_MOSI	AG35
SPI0_SCK	AE32
SPI0_SS_B	AE34
SPI1_MISO	AK34
SPI1_MOSI	AH32
SPI1_SCK	AH34
SPI1_SS_B	AJ33
TCK	M3
TDI	N4
TDO	R1
THRM_B	B9
TMS	N2
TRST_B	L4
TS_IREF_N	W9
TS_TDA	AC7
TS_TDC	AC5

USB_CLK96N	M33
USB_CLK96P	M35
USBD_DN	M31
USBD_DP	M29
USBH0_DN	V29
USBH0_DP	V30
USBH0_OC_B	G35
USBH0_PWR_EN	F34
USBH1_DN	P30
USBH1_DP	P29
USBH1_OC_B	J32
USBH1_PWR_EN	G33
VCC1P0_S0	V14
VCC1P0_S0	V16
VCC1P0_S3	P14
VCC1P0_S5	T13
VCC1P5_S0	AD5
VCC1P8_S0	P20
VCC1P8_S0	T20
VCC1P8_S3	K18
VCC1P8_S5	K13
VCC3P3_A	F30
VCC3P3_S0	AA26
VCC3P3_S0	AB24
VCC3P3_S0	AD24
VCC3P3_S3	K14
VCC3P3_S5	P5
VCC3P3_USB_S3	K16
VCCAA_1P8	K22
VCCACLKDDR_1P0	AB13
VCCADDR_1P0	AB14
VCCADLLDDR_1P0	AD14
VCCAICLKCB_1P0	T11
VCCAICLKDBUFF_1P0	V11
VCCAICLKSE_3P3	P7
VCCAICLKSFR_1P5	R10
VCCAICLKSSC1_1P0	V13

VCCAPCIE_1P0	Y14
VCCAUSB_1P8	P22
VCCAUSB_1P8_S3	K20
VCCAVISA_1P0	T18
VCCCLKDDR_1P5	AT28
VCCDDR_1P5	AT12
VCCDDR_1P5	AT16
VCCDDR_1P5	AT18
VCCDDR_1P5	AT23
VCCDICLKDIG_1P0	P11
VCCFHVSOC_1P05	T24
VCCFSOC_1P05	R26
VCCPLLDDR_1P0	AD13
VCCRTC_3P3	E1
VCCSFRPLLDDR_1P5	AD31
VNN	V18
VNN	V20
VNN	Y16
VNN	Y18
VNN	Y20
VNN	AB18
VNN	AB20
VNNSENSE	J30
VSS	A7
VSS	A10
VSS	A12
VSS	A14
VSS	A16
VSS	A18
VSS	A23
VSS	A28
VSS	A31
VSS	A32
VSS	A34
VSS	B29
VSS	B31
VSS	B33

VSS	B35
VSS	C1
VSS	C35
VSS	D5
VSS	D29
VSS	D31
VSS	D33
VSS	E32
VSS	E35
VSS	F6
VSS	F32
VSS	G1
VSS	G12
VSS	H17
VSS	J25
VSS	K35
VSS	M1
VSS	M5
VSS	M11
VSS	M13
VSS	M16
VSS	M22
VSS	M28
VSS	P9
VSS	P13
VSS	P27
VSS	T16
VSS	T22
VSS	U1
VSS	U35
VSS	V9
VSS	V22
VSS	V27
VSS	W1
VSS	W27
VSS	Y11
VSS	Y13



VSS	Y22
VSS	AB3
VSS	AB11
VSS	AB16
VSS	AB35
VSS	AC2
VSS	AC9
VSS	AC29
VSS	AC30
VSS	AD6
VSS	AD8
VSS	AD16
VSS	AD18
VSS	AD20
VSS	AD35
VSS	AE26
VSS	AF13
VSS	AF20
VSS	AF22
VSS	AF24
VSS	AG1
VSS	AG3
VSS	AH11
VSS	AH17
VSS	AJ35
VSS	AK15
VSS	AK29
VSS	AL6
VSS	AL23
VSS	AM1
VSS	AM4
VSS	AM7
VSS	AM32
VSS	AN9
VSS	AN13
VSS	AN27
VSS	AP2

VSS	AP18
VSS	AP23
VSS	AP33
VSS	AT7
VSSA_USB	P16
VSSSENSE	H30
WAKE_B	A5
XTAL_IN	AA2
XTAL_OUT	AB1

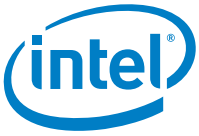
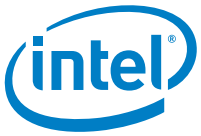
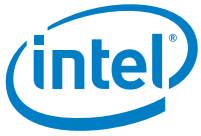


Table 26.





4.0 Electrical Characteristics

This chapter contains the DC and AC characteristics for Intel® Quark SoC X1000. AC timing diagrams are included.

4.1 Absolute Maximum Ratings

Table 27 specifies the absolute maximum and minimum ratings of the Intel® Quark SoC X1000 processor. At conditions outside of the functional operating condition limits, but within the absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. If a device is returned to conditions within the functional operating limits after having been subjected to conditions outside these limits (but within the absolute maximum and minimum ratings) the device may be functional, but with its lifetime degraded depending on exposure to conditions exceeding the functional operating condition limits.

At conditions exceeding the absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. Moreover, if a device is subjected to these conditions for any length of time, it will either not function or its reliability will be severely degraded when returned to conditions within the functional operating condition limits.

Although the SoC contains protective circuitry to resist damage from Electrostatic Discharge (ESD), precautions should always be taken to avoid high static voltages or electric fields.

All voltage values are given with respect to VSS.

Table 27. Intel® Quark SoC X1000 Absolute Maximum Voltage Ratings (Sheet 1 of 2)

Parameter	Minimum Limits	Maximum Limits
Temperature		
Junction Temperature (C)	0	110
Storage Temperature Range (C)	-55	125
Supplies		
3.3V Supply Voltage (V)	-0.5	3.7
1.8V Supply Voltage (V)	-0.3	2.0
1.5V Supply Voltage (V)	-0.2	1.65
1.05V Supply Voltage (V)	-0.2	1.3
1.0V Supply Voltage	-0.2	1.3
Signals		
Voltage on any 3.3 V Pin (V)	-0.5	3.7 V

Table 27. Intel® Quark SoC X1000 Absolute Maximum Voltage Ratings (Sheet 2 of 2)

Parameter	Minimum Limits	Maximum Limits
Voltage on any 1.8 V Pin (V)	-0.5	2.0
Voltage on any 1.5 V Pin (V)	-0.5	1.65
Voltage on any 1.0V Tolerant Pin (V)	-0.5	1.3

4.2 Recommended Power Supply Ranges

Table 28 shows the recommended operating voltage ranges for each SoC supply pin. Typically minimum and maximum values are +/- 5% of the nominal value unless otherwise stated.

Table 28. Power Supply Rail Ranges (Sheet 1 of 2)

Package Ball	Description	Min (V)	Nom (V)	Max (V)	Notes
VCC3P3_S5	S5 3.3V rail Standby LDO input	3.20	3.30	3.40	+/-3%
VCC3P3_S3	S3 3.3V rail Standby LDO input	3.20	3.30	3.40	+/-3%
VCC3P3_S0	S0 3.3V rail Standby LDO input	3.20	3.30	3.40	+/-3%
VCC3P3_A	S0 3.3V rail Standby LDO input	3.20	3.30	3.40	+/-3%
VNN	Default 1.0V standard cell rail including Core and Uncore logic	0.95	1.00	1.10	Active in state S0 only
VCC1P0_S3	Standard 1.0V rail for S3 logic	0.95	1.00	1.05	
VCC1P0_S5	Standard 1.0V rail for S5 logic	0.95	1.00	1.05	
VCCPLLDDR_1P0	DDR IO digital PLL high voltage	0.95	1.00	1.05	
VCCADLLDDR_1P0	DDR IO digital isolated quiet supply	0.95	1.00	1.05	
VCCACLKDDR_1P0	DDR IO digital clock isolated quiet supply	0.95	1.00	1.05	
VCCADDR_1P0	DDR IO Digital supply	0.95	1.00	1.05	
VCCDDR_1P5	DDR IO analog thick gate supply	1.42	1.50	1.57	
VCCSFRPLLDDR_1P5	DDR IO PLL high voltage	1.42	1.50	1.57	
VCCCLKDDR_1P5	DDR IO clock analog thick gate isolated quiet supply	1.42	1.50	1.57	
VCCAPCIE_1P0	PCIe analog supply	0.95	1.00	1.05	
VCC1P5_S0	PCIe band-gap supply	1.42	1.50	1.57	
VCC1P8_S0	S0 1.8V CFIO supply	1.71	1.80	1.89	
VCC1P8_S3	S3 1.8V CFIO supply	1.71	1.80	1.89	
VCC1P8_S5	S5 1.8V CFIO supply	1.71	1.80	1.89	
VCCRTC_3P3	RTC well supply	2.00	-	3.40	
VCCAUSB_1P8_S3	USB 1.8V analog supply - suspend rail	1.71	1.80	1.89	
VCC3P3_USB_S3	USB 3.3V supply - suspend rail	3.13	3.30	3.46	
VCCAUSB_1P8	USB 1.8V supply	1.71	1.80	1.89	
VSSA_USB	USB low-noise ground	0.00	0.00	0.00	
VCCAA_1P8	1.8V analog supply	1.71	1.80	1.89	
VCC1P0_S0	Standard 1.0V rail for HPLL (host PLL) and USB logic	0.95	1.00	1.05	Active in S0-only; can in general be connected to VNN

Table 28. Power Supply Rail Ranges (Sheet 2 of 2)

Package Ball	Description	Min (V)	Nom (V)	Max (V)	Notes
VCCAVISA_1P0	VISA IO analog supply	0.95	1.00	1.05	
VCCFHVSOC_1P05	SoC Fuses supply (sensing)	1.0	1.05	1.10	
VCCFSOC_1P05	Fuse digital sensing	1.0	1.05	1.10	
VCCAICLKCB_1P0	ICLK control supply	0.95	1.00	1.05	
VCCAICLKSSC1_1P0	ICLK SSC supply	0.95	1.00	1.05	
VCCAICLKDBUFF_1P0	ICLK differential output buffer supply	0.95	1.00	1.05	
VCCDICLKDIG_1P0	ICLK digital supply	0.95	1.00	1.05	
VCCAICLKSFR_1P5	ICLK SFR (for oscillator, IPLL)	1.42	1.50	1.57	
VCCAICLKSE_3P3	ICLK single ended output buffer supply	3.13	3.30	3.46	

4.3 Maximum Supply Current

Table 29. Maximum Supply Current: ICC Max (Sheet 1 of 2)

Voltage Rail	Voltage (V)	I_{CC}Max (mA)[3]
VCC3P3_S5	3.3	180
VCC3P3_S3	3.3	280
VCC3P3_S0	3.3	680
VCC3P3_A	3.3	5
VNN	1.0V	480
VCC1P0_S3	1.0	70
VCC1P0_S5	1.0	40
VCCPLLDDR_1P0	1.0	25
VCCADLLDDR_1P0	1.0	100
VCCACLKDDR_1P0	1.0	15
VCCADDR_1P0	1.0	300
VCCDDR_1P5	1.5	580
VCCSFRPLLDDR_1P5	1.5	35
VCCCLKDDR_1P5	1.5	90
VCCAPCIE_1P0	1.0	330
VCC1P5_S0	1.5	120
VCC1P8_S0	1.8	50
VCC1P8_S3	1.8	110
VCC1P8_S5	1.8	15
VCCRTC_3P3	3.3	3
VCCRTC_3P3 (Battery)	3.0	6uA
VCCAUSB_1P8_S3	1.8	20
VCC3P3_USB_S3	3.3	40
VSSA_USB	0.0	20

Table 29. Maximum Supply Current: ICC Max (Sheet 2 of 2)

Voltage Rail	Voltage (V)	I_{CC}Max (mA)[3]
VCCAA_1P8	1.8	10
VCC1P0_S0	1.0	120
VCCAVISA_1P0	1.0	280
VCCFHVSOC_1P05	1.05	100
VCCFSOC_1P05	1.05	40
VCCAICLKCB_1P0	1.0	40
VCCAICLKSSC1_1P0	1.0	15
VCCAICLKDBUFF_1P0	1.0	30
VCCDICLKDIG_1P0	1.0	15
VCCAICLKSFR_1P5	1.5	40
VCCAICLKSE_3P3	3.3	15
VCCAUSB_1P8	1.8	20

4.4 Configurable IO Characteristics

The signals in Table 30 are brought on- and off-chip via standard configurable IO groups. This section describes the DC and AC characteristics associated with these signals.

Table 30. CFIO AC Characteristics

Type	Symbol	Parameter	Min	Max	Units	Conditions
This data applies to all signals in Table 9. To determine the correct V_{CC} supply for a signal use the related supply shown in Table 9						
	V_{CC}	Supply Voltage Reference	3.13	3.49	V	
Output	SR_{RISE}	Slew Rate Rise	0.5	3.0	V/ns	C_{LOAD} = 10pF
	SR_{FALL}	Slew Rate Fall	0.5	3.0	V/ns	C_{LOAD} = 10pF
	T_{RISE}	Output Rise Time	0.88	5.28	ns	$0.10*V_{CC}$ - $0.90*V_{CC}$
	T_{FALL}	Output Fall Time	0.88	5.28	ns	$0.90*V_{CC}$ - $0.10*V_{CC}$

4.5 RTC DC Characteristics

Table 31. RTC DC Characteristics (Sheet 1 of 2)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Associated Signals: RTCRST_B ***Related Supply (V_{CC}):*** VCCRTC_3P3							
Input	V_{CC}	Supply Voltage Reference	2.00	3.40	V		
	V_{IH}	Input High Voltage	2.3	V_{CC} + 0.5	V		2
	V_{IL}	Input Low Voltage	-0.5	0.78	V		2
	C_{IN}	Input Pin Capacitance	—	3	pF		

Table 31. RTC DC Characteristics (Sheet 2 of 2)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Associated Signals: S5_PG, RTC_EXT_CLK_EN_B, S0_PG ***Related Supply (V_{CC}):*** VCCRTC_3P3							
Input	V_{CC}	Supply Voltage Reference	2.0	3.40	V		
	V_{IH}	Input High Voltage	2.0	V_{CC} + 0.5	V		2
	V_{IL}	Input Low Voltage	-0.5	0.78	V		2
	C_{IN}	Input Pin Capacitance	—	3	pF		

4.6 PCI Express* 2.0 DC/AC Characteristics

Table 32. PCI Express* 2.0 Differential Signal DC Characteristics

Symbol	Parameter	Min	Max	Unit	Figures	Notes
Associated Signals: PCIE_PERN[1], PCIE_PERP[1], PCIE_PETN[1], PCIE_PETP[1], PCIE_PERN[0], PCIE_PERP[0], PCIE_PETN[0], PCIE_PETP[0] ***Related Supply (V_{CC}):*** VCCAPCIE_1P0						
$V_{TX-DIFF\ P-P}$	Differential Peak to Peak Output Voltage	0.8	1.2	V		1
$V_{TX-DIFF\ P-P\ -\ Low}$	Low power differential Peak to Peak Output Voltage	0.4	1.2	V		
V_{TX_CM-ACp}	TX AC Common Mode Output Voltage (2.5GT/s)	—	20	mV		
$Z_{TX-DIFF-DC}$	DC Differential TX Impedance	80	120	Ω		
$V_{RX-DIFF\ p-p}$	Differential Input Peak to Peak Voltage	0.175	1.2	V		1
V_{RX_CM-ACp}	AC peak Common Mode Input Voltage	—	150	mV		

Notes:
1. PCI Express mVdiff p-p = 2*|PETP[x] – PETN[x]|; PCI Express mVdiff p-p = 2*|PERP[x] – PERN[x]|

Table 33. PCI Express* 2.0 Interface Timings (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Figures	Notes
Transmitter and Receiver Timings						
UI	Unit Interval – PCI Express*	399.88	400.12	ps		5,6
T_{TX-EYE}	Minimum Transmission Eye Width	0.7	—	UI	4	1,2,6

Table 33. PCI Express* 2.0 Interface Timings (Sheet 2 of 2)

$T_{TX-RISE/Fall}$	D+/D- TX Out put Rise/Fall time	0.125		UI		1,2,6
T_{RX-EYE}	Minimum Receiver Eye Width	0.40	—	UI	5	3,4,6

Notes:

1. Specified at the measurement point into a timing and voltage compliance test load and measured over any 250 consecutive TX UIs. (Also refer to the Transmitter compliance eye diagram)
2. A T_{TX-EYE} = 0.70 UI provides for a total sum of deterministic and random jitter budget of $T_{TXJITTER-MAX}$ = 0.30 UI for the Transmitter collected over any 250 consecutive TX UIs. The $T_{TXEYE-MEDIAN-to-MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total TX jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value.
3. Specified at the measurement point and measured over any 250 consecutive UIs. The test load documented in the PCI Express* specification 2.0 should be used as the RX device when taking measurements (also refer to the Receiver compliance eye diagram). If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as a reference for the eye diagram.
4. A T_{RX-EYE} = 0.40 UI provides for a total sum of 0.60 UI deterministic and random jitter budget for the Transmitter and interconnect collected any 250 consecutive UIs. The $_{TRX-EYE-MEDIAN-to--MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total 0.6 UI jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value. If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as the reference for the eye diagram.
5. Nominal Unit Interval is 400 ps for 2.5 GT/s.
6. Intel® Quark SoC X1000 supports PCI Gen 1 timing only: 2.5 GT/s

Figure 4. PCI Express Transmitter Eye

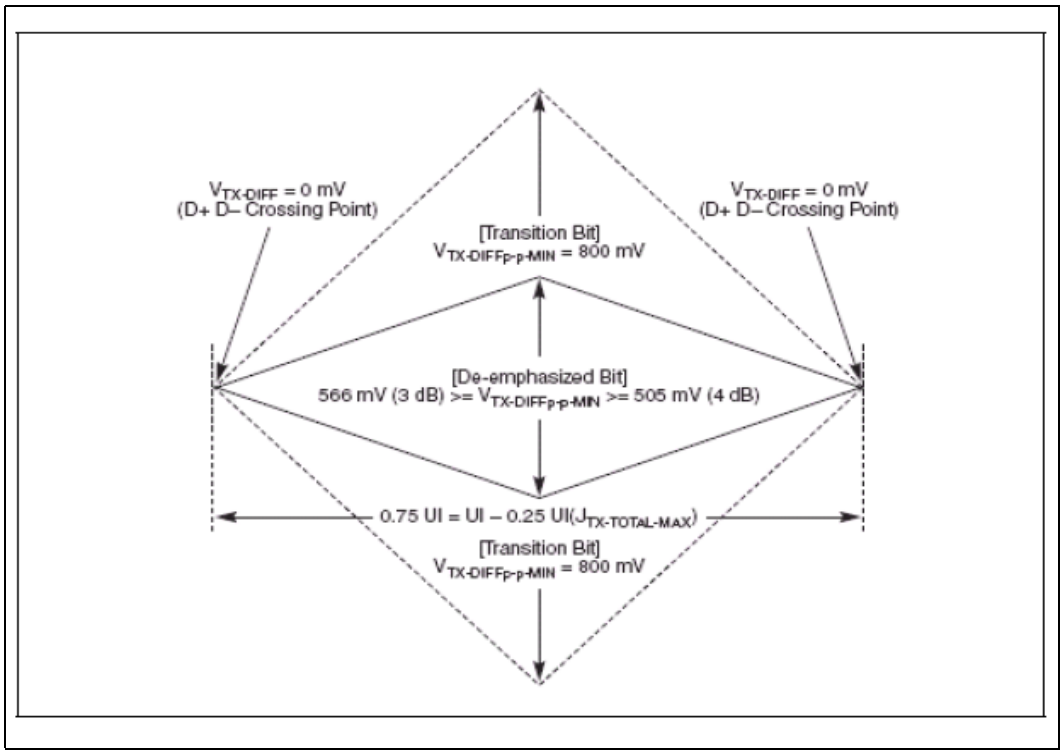
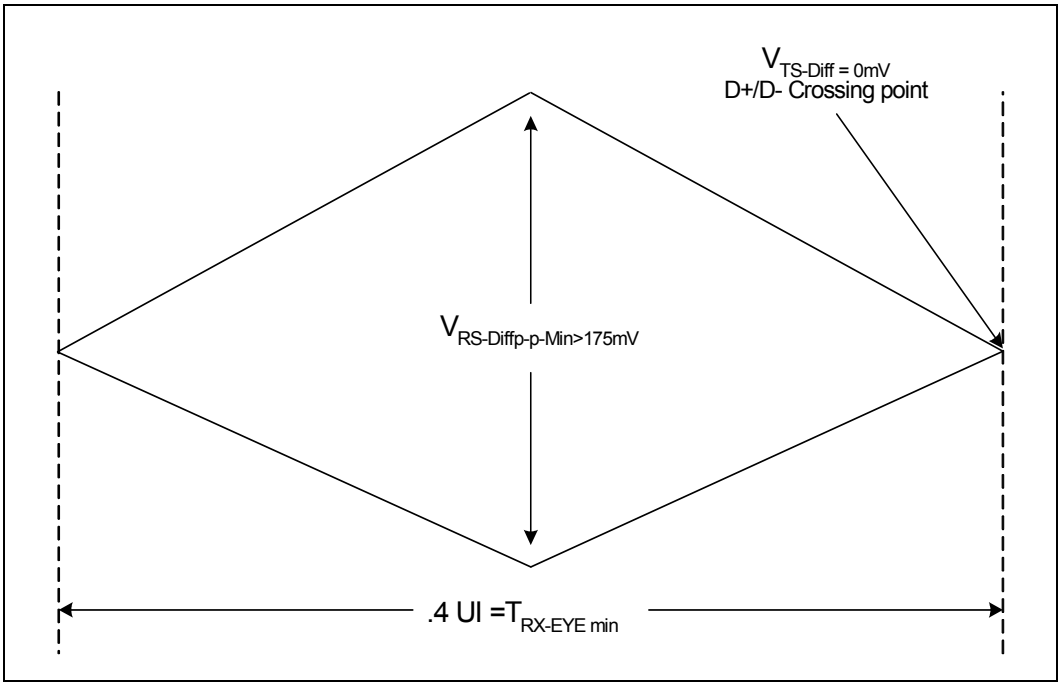


Figure 5. PCI Express Receiver Eye



4.7 USB 2.0 DC/AC Characteristics

Table 34. USB 2.0 Differential Signal DC Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Conditions	Notes
Associated Signals: USBH2_DP, USBH2_DN, USBH1_DP, USBH1_DN, USBH0_DP, USBH0_DN ***Related Supply (V_{CC}):*** VCCUSBSUS_3P3						
VDI	Differential Input Sensitivity	0.2	—	V		1,3
VCM	Differential Common Mode Range	0.8	2.5	V		2,3
VSE	Single-Ended Receiver Threshold	0.8	2	V		3
VCRS	Output Signal Crossover Voltage	1.3	2	V		3
VOL	Output Low Voltage	—	0.4	V	Iol=5 mA	3
VOH	Output High Voltage	V_{CC} – 0.5	—	V	Ioh=-2mA	3
VHSSQ	HS Squelch Detection Threshold	100	150	mV		4
VHSDSC	HS Disconnect Detection Threshold	525	625	mV		4
VHSCM	HS Data Signaling Common Mode Voltage Range	-50	500	mV		4
VHSOI	HS Idle Level	-10	10	mV		4
VHSOH	HS Data Signaling High	360	440	mV		4
VHSOL	HS Data Signaling Low	-10	10	mV		4

Table 34. USB 2.0 Differential Signal DC Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Conditions	Notes
VCHIRPJ	Chirp J Level	700	1100	mV		4
VCHIRPK	Chirp K Level	-900	-500	mV		4

Notes:
1. V_{DI} = | USBHx_DP – USBHx_DN |
2. Includes VDI range
3. Applies to Low-Speed/Full-Speed USB
4. Applies to High-Speed USB 2.0

Table 35. USB 2.0 Interface Timings (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Units	Notes	Fig
Full-speed Source (Note 7)						
t100	USBHx_DP, USBHx_DN - Driver Rise Time	4	20	ns	1,6 C_L = 50 pF	7
t101	USBHx_DP, USBHx_DN Driver Fall Time	4	20	ns	1,6 C_L = 50 pF	7
t102	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	–3.5 –4	3.5 4	ns ns	2, 3	8
t103	Source SE0 interval of EOP	160	175	ns	4	9
t104	Source Jitter for Differential Transition to SE0 Transition	–2	5	ns	5	
t105	Receiver Data Jitter Tolerance - T o Next Transition - For Paired Transitions	–18.5 –9	18.5 9	ns ns	3	8
t106	EOP Width: Must accept as EOP	82	—	ns	4	9
t107	Width of SE0 interval during differential transition	—	14	ns		
Low-speed Source (Note 8)						
t108	USBHx_DP, USBHx_DN - Driver Rise Time	75	300	ns	1, 6 C_L = 200 pF C_L = 600 pF	7
t109	USBHx_DP, USBHx_DN Driver Fall Time	75	300	ns	1,6 C_L = 200 pF C_L = 600 pF	7
t110	Source Differential Driver Jitter To Next Transition For Paired Transitions	–25 –14	25 14	ns ns	2, 3	8
t111	Source SE0 interval of EOP	1.25	1.50	µs	4	9
t112	Source Jitter for Differential Transition to SE0 Transition	–40	100	ns	5	
t113	Receiver Data Jitter Tolerance - To Next Transition - For Paired Transitions	–152 –200	152 200	ns ns	3	8

Table 35. USB 2.0 Interface Timings (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Units	Notes	Fig
Full-speed Source (Note 7)						
t114	EOP Width: Must accept as EOP	670	—	ns	4	9
t115	Width of SE0 interval during differential transition	—	210	ns		

Notes:
1. Driver output resistance under steady state drive is specified at 28 Ω at minimum and 43 Ω at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. Full-speed Data Rate has minimum of 11.97 Mb/s and maximum of 12.03 Mb/s.
8. Low-speed Data Rate has a minimum of 1.48 Mb/s and a maximum of 1.52 Mb/s.

Figure 6. USB Rise and Fall Time

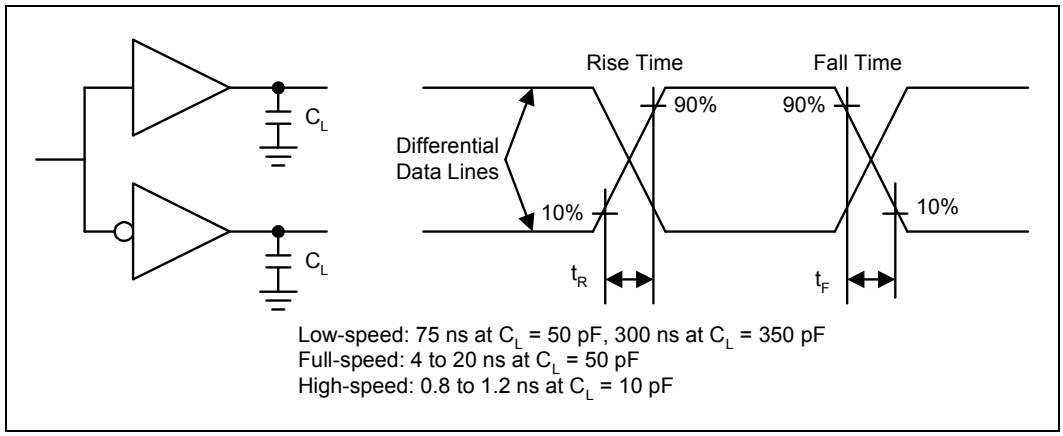


Figure 7. USB Jitter

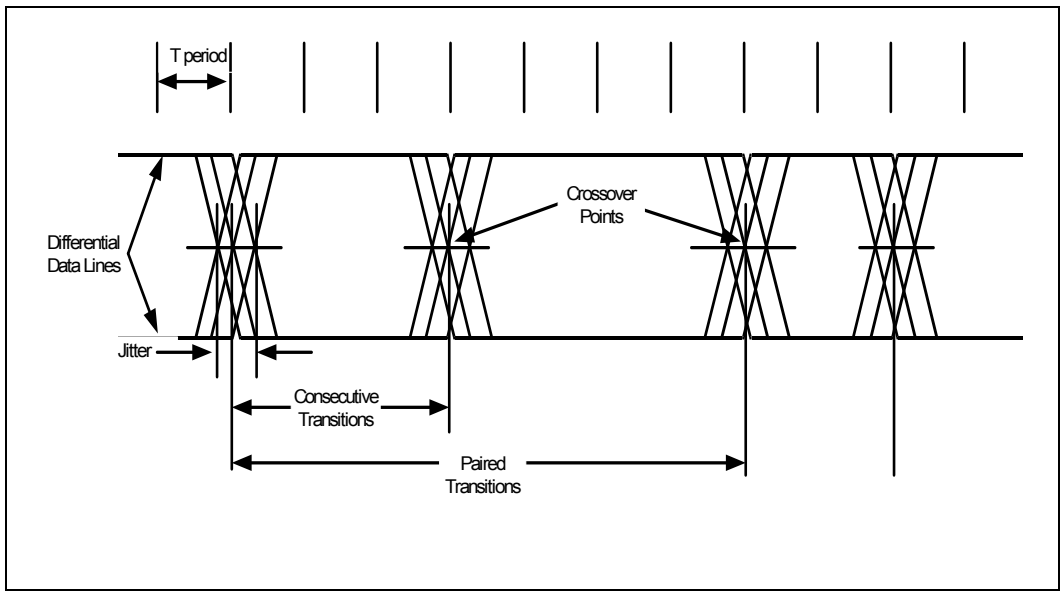
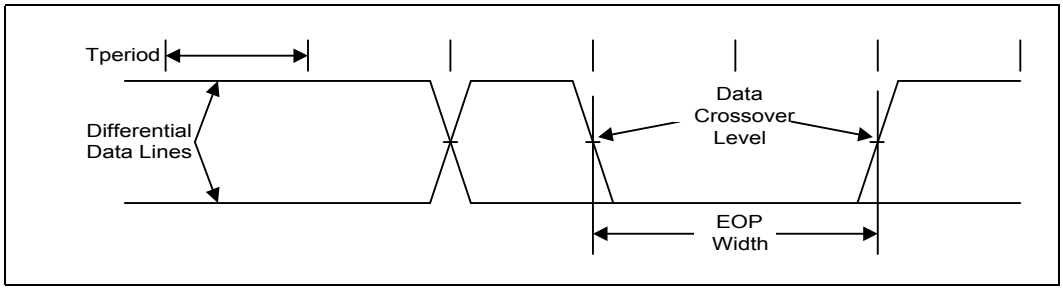


Figure 8. USB EOP Width



4.8 General Interface Timing

4.8.1 Legacy SPI Interface Timing

Table 36. Legacy SPI Interface Timings (20 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
F	Serial Clock Frequency - 20M Hz Operation	—	20	MHz		
T_{CH}	LSPI_SCK high time	20	—	ns		9
T_{CL}	LSPI_SCK low time	30	—	ns		9
T_{DSCR}	Setup of LSPI_MISO with respect to LPSI_SCK rising edge	11.7	—	ns		9
T_{CRDH}	Hold time of LSPI_MISO with respect to LPSI_SCK rising edge	-3.0	—	ns		9
T_{CFDV}	LSPI_SCK falling edge to LSPI_MOSI valid	-1.9	2.5	ns		9
T_{CFSF}	LSPI_SCK falling edge to LSPI_SS_B low	-1.6	2.4	ns		9
T_{CFSR}	LSPI_SCK falling edge to LSPI_SS_B high	-1.7	2.3	ns		9

Note:
1. All input signals have a slope of 1.0ns measured between 20%and 80% V_{CC} values
2. All output signals are loaded with 20pF
3. Measurements are made at 50% V_{CC} levels

4.8.2 SPI0/1 Interface Timing

Table 37. SPI0/1 Interface Timings (25 MHz)

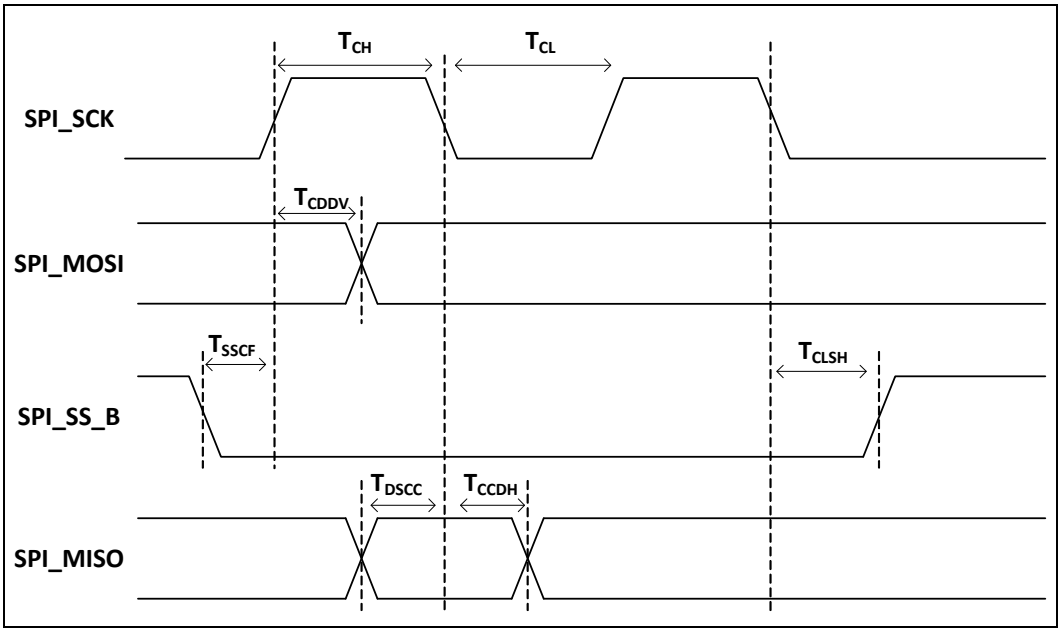
Sym	Parameter	Min	Max	Units	Notes	Fig
F	Serial Clock Frequency - 25MHz Operation	—	25	MHz		
T_{CH}	SPI0/1_SCK high time	20	—	ns		9
T_{CL}	SPI0/1_SCK low time	20	—	ns		9
T_{DSCR}	Setup of SPI0/1_MISO with respect to SPI0/1_SCK capturing edge	10.7	—	ns		9
T_{CRDH}	Hold time of SPI0/1_MISO with respect to SPI0/1_SCK capturing edge	-2.5	—	ns		9
T_{CFDV}	SPI0/1_SCK driving edge to SPI0/1_MOSI valid	-0.8	3.5	ns		9

Table 37. SPI0/1 Interface Timings (25 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
T_{SSCF}	Setup of SPI0/1_SS_B with respect to first edge out of inactive state of SPI0/1_SCK	20	—	ns		9
T_{CLSH}	Hold of SPI0/1_SS_B with respect to last edge into inactive state of SPI0/1_SCK	20	—	ns		9

Note:
1. All input signals have a slope of 1.0ns measured between 20%and 80% V_{CC} values
2. All output signals are loaded with 20pF
3. Measurements are made at 50% V_{CC} levels
4. Driving edge and capturing edge of SPI0/1_SCK are determined by SPI Control Register 1 settings SSCR1.SPH and SSCR1.SPO; Figure 9 shows SPI_SCK rising edge as the driving edge and SPI_SCK falling edge as the capturing edge by way of example

Figure 9. SPI Interface Timing



4.8.3 SDIO Interface Timing

Table 38. SDIO Timing (Sheet 1 of 2)

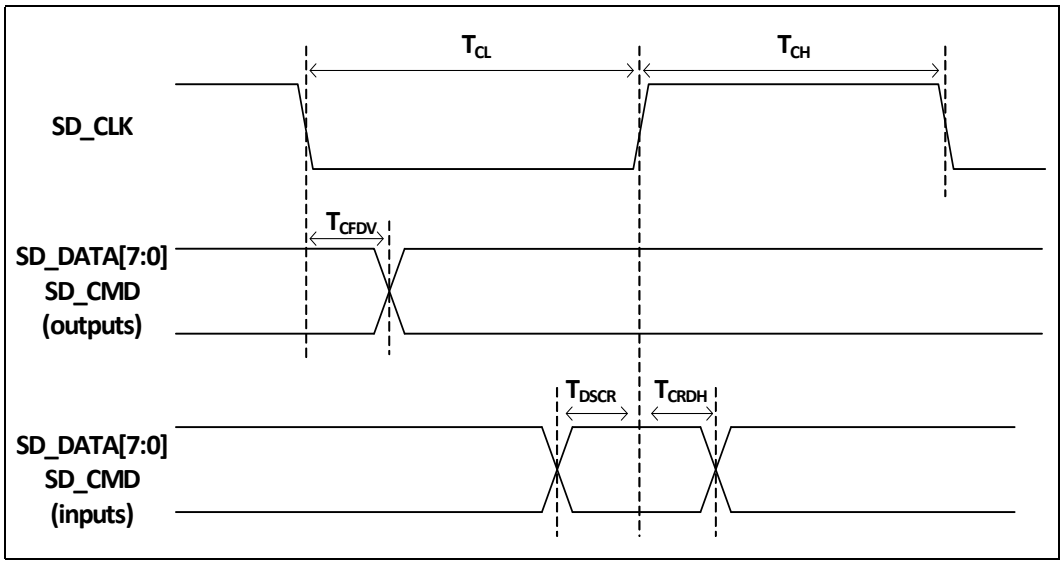
Sym	Parameter	Min	Max	Units	Notes	Fig
F	Operating Frequency SD_CLK	-	50	MHz		
T_{CH}	Clock High Time SD_CLK	10	-	ns		10
T_{CL}	Clock Low Time SD_CLK	10	-	ns		10
T_{DSCR}	SD_DATA[7:0]/SD_CMD setup time with respect to SD_CLK rising	0.6	-	ns		10

Table 38. SDIO Timing (Sheet 2 of 2)

Sym	Parameter	Min	Max	Units	Notes	Fig
T_{CRDH}	SD_DATA[7:0]/SD_CMD hold time with respect to SD_CLK rising	1.6	-			10
T_{CFDV}	SD_CLK falling to data valid on SD_DATA[7:0]/SD_CMD	-2.0	2.7	ns		10

Note:
1. All input signals have a slope of 1.0ns measured between 20%and 80% V_{CC} values
2. All output signals are loaded with 20pF
3. Measurements are made at 50% V_{CC} levels

Figure 10. SDIO Interface Timing



4.9 Clock AC Timing

4.9.1 Reference Clock AC Characteristics

Table 39. Reference Clocks AC Characteristics (Sheet 1 of 2)

Parameter	Description	Min	Max	Units	Notes	Fig
Associated Signals: REF0_OUTCLK_P, REF0_OUTCLK_N, REF1_OUTCLK_P, REF1_OUTCLK_N ***Related Supply:*** VCCAICLKDBUFF_1P0						
T_{SLEW_RISE}	Rising slew rate	1.5	8.0	V/ns	2,3,10	11
T_{SLEW_FALL}	Falling slew rate	1.5	8.0	V/ns	2,3,10	11
P_{SLEW_VAR}	Slew rate matching	-	20	%	1,9,10	11
V_{SWING}	Differential output swing	300	-	mV	2,11	11
V_{CROSS}	Crossing point voltage	300	550	mV	1,4,5,11	11
V_{CROSS_DELTA}	V_{CROSS} variation	-	140	mV	1,4,8,11	11

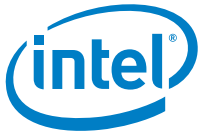


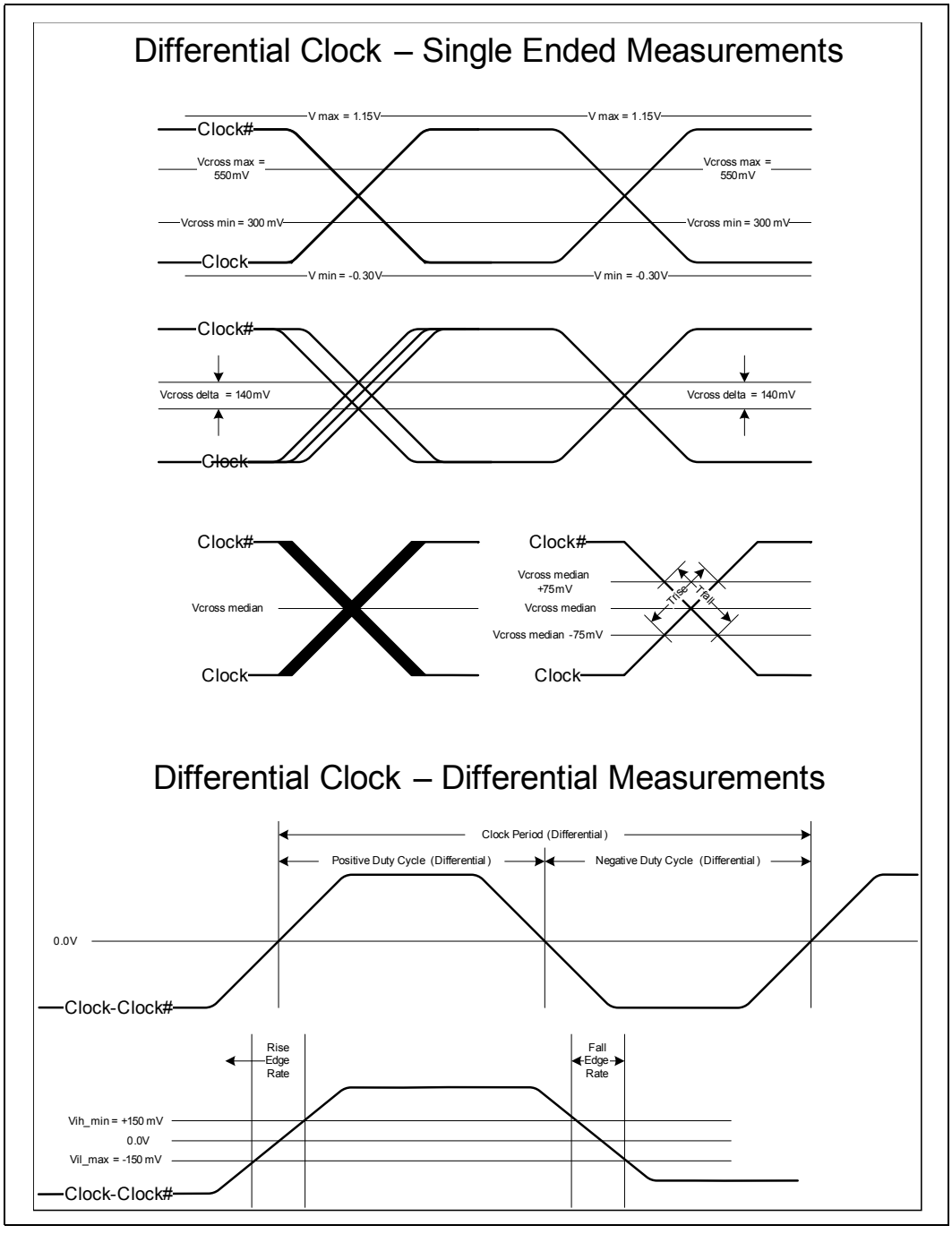
Table 39. Reference Clocks AC Characteristics (Sheet 2 of 2)

Parameter	Description	Min	Max	Units	Notes	Fig
V_{MAX}	Maximum output voltage	-	1.15	V	1,6,11	11
V_{MIN}	Minimum output voltage	-0.3	-	V	1,7,11	11
P_{DTY_CYC}	Duty Cycle	40	60	%	2,10	11

Note:

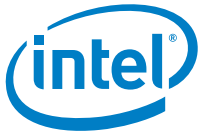
1. Measurement taken from a single-ended waveform on a component test board
2. Measurement taken from a differential waveform on a component test board
3. Slew rate measured through V_{SWING} voltage measured at differential zero
4. VCROSS is defined as the voltage where CLK_P = CLK_N
5. Only applies to differential rising edge (CLK_P rising, CLK_N falling)
6. The maximum voltage including over-shoot
7. The minimum voltage including under-shoot
8. The total variation of all V_{CROSS} measurements in any particular system
9. Matching applies to rising edge rate for CLK_P and falling edge rate for CLK_N; It is measured using a ±75mV window centered on the average cross point where CLK_P rising meets CLK_N falling. The median cross point is used to calculate the voltage thresholds the oscilloscope is to use for the edge rate calculations
10. Average measurement
11. Instantaneous measurement

Figure 11. Measurement Points for Differential Clocks[1]



1. Clock == CLK_P; Clock# == CLK_N

§ §



5.0 Register Access Methods

There are six different common register access methods:

- Fixed I/O
- Fixed Memory
- I/O Referenced
- Memory Referenced
- PCI Configuration (Indirect - via Memory or I/O registers)
- Message Bus (Indirect - via PCI Configuration Registers)

5.1 Fixed I/O Register Access

Fixed I/O registers are accessed by specifying their 16-bit address in a PORT IN and/or PORT OUT transaction from the CPU core. This allows direct manipulation of the registers. Fixed I/O registers are unmovable registers in I/O space.

Table 40. Fixed I/O Register Access Method Example (NSC Register)

Type: I/O Register (Size: 8 bits)	**NSC:** 61h

5.2 Fixed Memory Mapped Register Access

Fixed Memory Mapped I/O (MMIO) registers are accessed by specifying their 32-bit address in a memory transaction from the CPU core. This allows direct manipulation of the registers. Fixed MMIO registers are unmovable registers in memory space.

Table 41. Fixed Memory Mapped Register Access Method Example (IDX Register)

Type: Memory Mapped I/O Register (Size: 32 bits)	**IDX:** FEC00000h

5.3 I/O Referenced Register Access

I/O referenced registers use programmable base address registers (BARs) to select a range of I/O addresses that it uses to decode PORT IN and/or PORT OUT transactions from the CPU to directly access a register. Thus, the I/O BARs act as pointers to blocks of actual I/O registers. To access an I/O referenced register for a specific I/O base address, start with that base address and add the register's offset. Example pseudocode for an I/O referenced register read is shown below:

```
Register_Snapshot = IOREAD([IO_BAR]+Register_Offset)
```

Base address registers are often located in the PCI configuration space and are programmable by the BIOS/OS. Other base address register types may include fixed memory registers, fixed I/O registers or message bus registers.

Table 42. Referenced I/O Register Access Method Example (PM1S Register)

Type: I/O Register (Size: 16 bits)	**PM1S:** [PM1BLK] + 0h **PM1BLK Type:** PCI Configuration Register (Size: 32 bits) **PM1BLK Reference:** [B:0, D:31, F:0] + 48h

5.4 Memory Referenced Register Access

The SoC uses programmable base address registers (BARs) to set a range of physical address (memory) locations that it uses to decode memory reads and writes from the CPU to directly access a register. These BARs act as pointers to blocks of actual memory mapped I/O (MMIO) registers. To access a memory referenced register for a specific base address, start with that base address and add the register's offset. Example pseudocode for a read is shown below:

```
Register_Snapshot = MEMREAD([Mem_BAR]+Register_Offset)
```

Base address registers are often located in the PCI configuration space and are programmable by the BIOS/OS. Other common base address register types include fixed memory registers and I/O registers that point to MMIO register blocks.

Table 43. Memory Mapped Register Access Method Example (ESD Register)

Type: Memory Mapped I/O Register (Size: 32 bits)	**ESD:** [RCBA] + 4h **RCBA Type:** PCI Configuration Register (Size: 32 bits) **RCBA Reference:** [B:0, D:31, F:0] + F0h

5.5 PCI Configuration Register Access

Access to PCI configuration space registers is performed through one of two different configuration access methods (CAMs):

- I/O indexed - PCI CAM
- Memory mapped - PCI Enhanced CAM (ECAM)

Each PCI function (see Section 6.3, "PCI Configuration Space" on page 95) has a standard PCI header consisting of 256 bytes for the I/O access scheme (CAM), or 4096 bytes for the enhanced memory access method (ECAM). Invalid read accesses return binary strings of 1s.

Table 44. PCI Register Access Method Example (PCI_DEVICE_VENDOR Register)

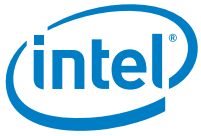
Type: PCI Configuration Register (Size: 32 bits)	**PCI_DEVICE_VENDOR:** [B:0, D:31, F:0] + 0h

5.5.1 PCI Configuration Access - CAM: I/O Indexed Scheme

Accesses to configuration space using the I/O method rely on two 32-bit I/O registers:

- **CONFIG_ADDRESS** - I/O Port CF8h
- **CONFIG_DATA** - I/O Port CFCh

These two registers are both 32-bit registers in I/O space. Using this indirect access mode, software uses CONFIG_ADDRESS (CF8h) as an index register, indicating which configuration space register to access, and CONFIG_DATA (CFCh) acts as a window to



the register pointed to in CONFIG_ADDRESS. Accesses to CONFIG_ADDRESS (CF8h) are internally captured. Upon a read or write access to CONFIG_DATA (CFCh), configuration cycles are generated to the PCI function specified by the address captured in CONFIG_ADDRESS. The format of the address is shown in Table 45.

Table 45. PCI CONFIG_ADDRESS Register (I/O PORT CF8h) Mapping

Field	CONFIG_ADDRESS Bits
Enable PCI Config. Space Mapping	31
Reserved	30:24
Bus Number	23:16
Device Number	15:11
Function Number	10:08
Register/Offset Number	07:02

Note: Bit 31 of CONFIG_ADDRESS must be set for a configuration cycle to be generated.

Pseudocode for a PCI register read is shown below:

```
MyCfgAddr[23:16] = bus; MyCfgAddr[15:11] = device; MyCfgAddr[10:8] = funct;

MyCfgAddr[7:2] = dWordMask(offset); MyCfgAddr[31] = 1;

IOWRITE(0xCF8, MyCfgAddr)

Register_Snapshot = IOREAD(0xCFC)
```

5.5.2 PCI Configuration Access - ECAM: Memory Mapped Scheme

A flat, 256 Mbyte memory space may also be allocated to perform configuration transactions. This is enabled through the HECREG message bus register (Port: 3h, Register: 09h) found in the Host Bridge. HECREG allows remapping this 256 Mbyte region anywhere in physical memory space. Memory accesses within the programmed MMIO range result in configuration cycles to the appropriate PCI devices specified by the memory address as shown below.

Table 46. PCI Configuration Memory Bar Mapping

ECAM Memory Address Field	ECAM Memory Address Bits
Use from BAR: HECREG[31:28]	31:28
Bus Number	27:20
Device Number	19:15
Function Number	14:12
Register Number	11:02

Note: ECAM accesses are only possible when HECREG.EC_ENABLE (bit 0) is set.

Pseudocode for an enhanced PCI configuration register read is shown below:

```
MyCfgAddr[27:20] = bus; MyCfgAddr[19:15] = device; MyCfgAddr[14:12] = funct;

MyCfgAddr[11:2] = dw_offset; MyCfgAddr[31:28] = HECREG[31:28];

Register_Snapshot = MEMREAD(MyCfgAddr)
```

5.6 Message Bus Register Access

Accesses to the message bus space are through the Host Bridge's PCI configuration registers. This unit relies on three 32-bit PCI configuration registers to generate messages:

- Message Bus Control Register (MCR) - PCI[B:0,D:0,F:0] + D0h
- Message Data Register (MDR) - PCI[B:0,D:0,F:0] + D4h
- Message Control Register eXtension (MCRX) - PCI[B:0,D:0,F:0] + D8h

This indirect access mode is similar to PCI CAM. Software uses the MCR/MCRX as an index register, indicating which message bus space register to access (MCRX only when required), and MDR as the data register. Writes to the MCR trigger message bus transactions.

Writes to MCRX and MDR are captured. Writes to MCR generates an internal 'message bus' transaction with the opcode and target (port, offset, byte enable) specified in the MCR and the captured MCRX. When a write opcode is specified in MCR, the data that was captured by MDR is used for the write. When a data read opcode is specified in MCR, the data is available in the MDR register after the MCR write completes (non-posted). The format of MCR and MCRX are shown in Table 47 and Table 48.

Table 47. MCR Description

Field	MBPR Bits
OpCode (typically 10h for read, 11h for write)	31:24
Port	23:16
Offset/Register	15:08
Byte Enable	07:04

Table 48. MCRX Description

Field	MBPER Bits
Offset/Register Extension. This is used for messages sent to end points that require more than 8 bits for the offset/register. These bits are a direct extension of MCR[15:8].	31:08

Most message bus registers are located in the Host Bridge. The default opcode messages for those registers are as follows:

- Message 'Read Register' Opcode: 10h
- Message 'Write Register' Opcode: 11h

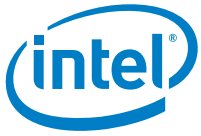
Registers with different opcodes are specified as applicable. Pseudocode of a message bus register read is shown below (where ReadOp==0x10):

```
MyMCR[31:24] = ReadOp; MyMCR[23:16] = port; MyMCR[15:8] = offset;

MyMCR[7:4] = 0xf

PCIWRITE(0, 0, 0, 0xD0, MyMCR)

Register_Snapshot = PCIREAD(0, 0, 0, 0xD4)
```

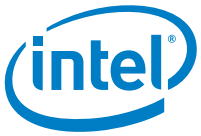


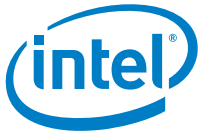
5.7 Register Field Access Types

Table 49. Register Access Types and Definitions

Access Type	Meaning	Description
RO	Read Only	In some cases, if a register is read only, writes to this register location have no effect. However, in other cases, two separate registers are located at the same location where a read accesses one of the registers and a write accesses the other register. See the I/O and memory map tables for details.
WO	Write Only	In some cases, if a register is write only, reads to this register location have no effect. However, in other cases, two separate registers are located at the same location where a read accesses one of the registers and a write accesses the other register. See the I/O and memory map tables for details.
R/W	Read/Write	A register with this attribute can be read and written.
R/WC	Read/Write Clear	A register bit with this attribute can be read and written. However, a write of 1 clears (sets to 0) the corresponding bit and a write of 0 has no effect.
R/WO	Read/Write-Once	A register bit with this attribute can be written only once after power up. After the first write, the bit becomes read only.
R/WL	Read/Write Lockable	A register bit with the attribute can be read at any time but writes may only occur if the associated lock bit is set to unlock. If the associated lock bit is set to lock, this register bit becomes RO unless otherwise indicated.
R/WLO	Read/Write, Lock-Once	A register bit with this attribute can be written to the non-locked value multiple times, but to the locked value only once. After the locked value has been written, the bit becomes read only.
R/W/SN	Read/Write	Read/Write register initial value loaded from NVM.
Reserved	Reserved	The value of reserved bits must never be changed.
Default	Default	When the processor is reset, it sets its registers to predetermined default states. The default state represents the minimum functionality feature set required to successfully bring up the system. Hence, it does not represent the optimal system configuration. It is the responsibility of the system initialization software to determine configuration, operating parameters, and optional system features that are applicable, and to program the processor registers accordingly.

§ §





6.0 Mapping Address Spaces

The Intel® Quark SoC X1000 supports four different address spaces:

- Physical Address Space (Memory Space)
- I/O Space
- PCI Configuration Space
- Message Bus Space

The CPU core can only directly access *memory space* through memory reads and writes and *I/O space* through the `IN` and `OUT` I/O port instructions. *PCI configuration space* is indirectly accessed through I/O or memory space, and the *Message Bus space* is accessed through PCI configuration space. See Chapter 5.0, "Register Access Methods" for details.

This chapter describes how the memory, I/O, PCI, and Message Bus spaces are mapped to interfaces in the SoC.

Note: See Chapter 12.0, "Host Bridge" for registers specified in the chapter.

6.1 Physical Address Space Mappings

There are 4 Gbyte (32-bits) of physical address space that can be used as:

- Memory Mapped I/O (MMIO - I/O fabric)
- Physical Memory (DRAM)

The CPU core can access the full physical address space, while downstream devices can only access SoC DRAM, and the CPU core's local APIC. Peer to peer transactions are not supported.

Most devices map their registers and memory to the physical address space. This chapter summarizes the possible mappings.

6.1.1 Bridge Memory Map

The Host Bridge maps the physical address space as follows:

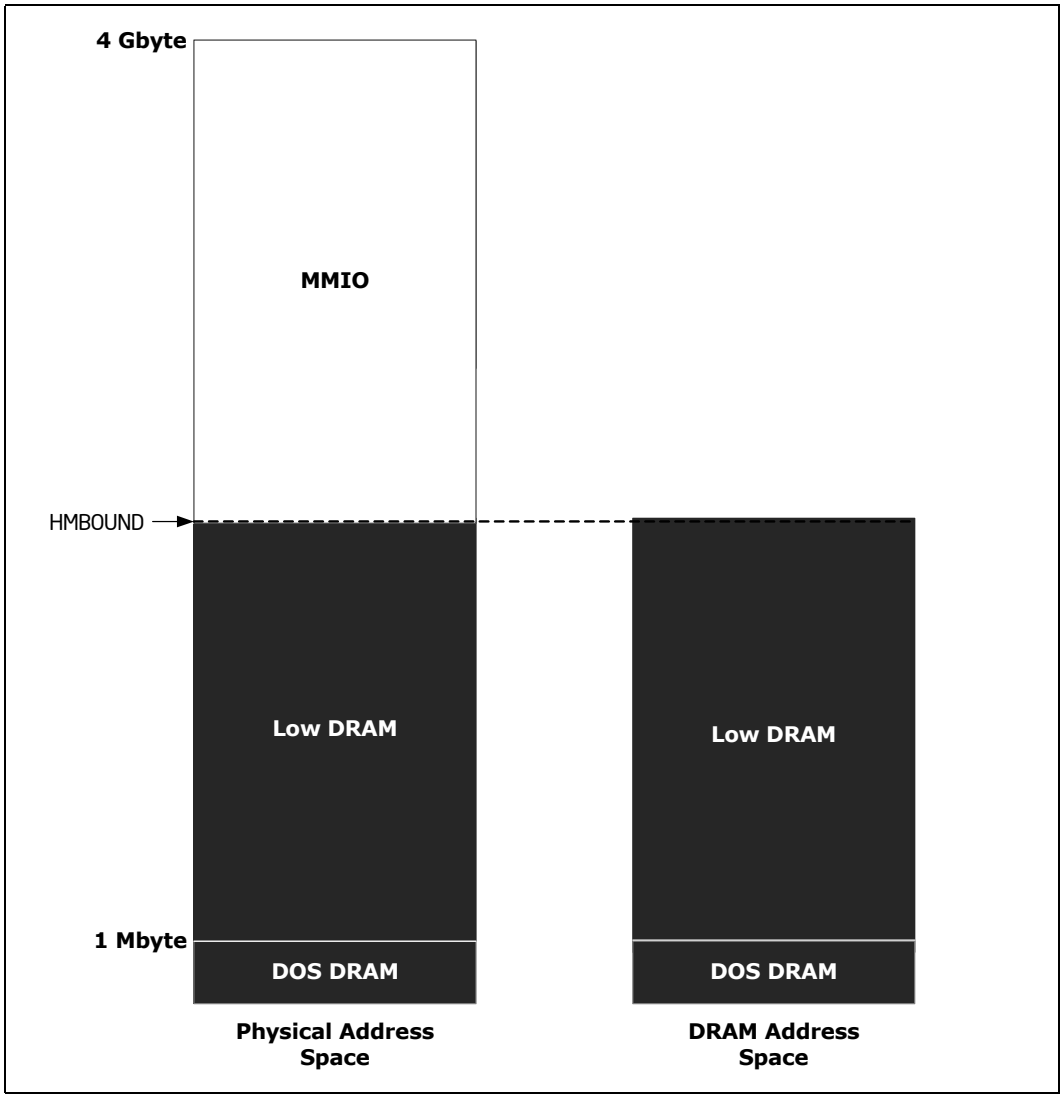
- CPU core to DRAM
- CPU core to I/O fabric (MMIO)
- CPU core to extended PCI registers (ECAM accesses)
- I/O fabric to CPU cores (local APIC interrupts)

This SoC has the following distinct memory regions:

- DOS DRAM + Low DRAM
- MMIO

The HMBOUND register is used to create these memory regions, as shown in Figure 12.

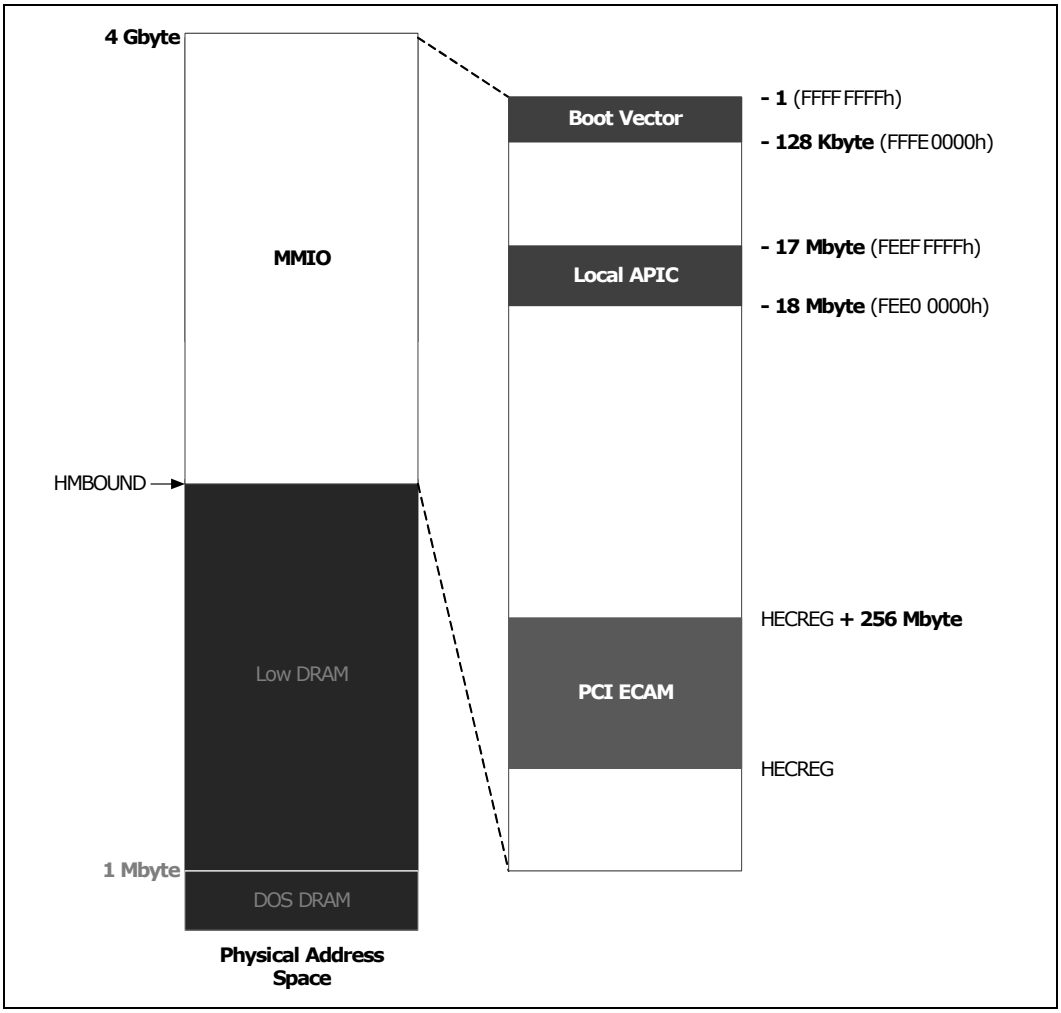
Figure 12. Physical Address Space - Low DRAM & MMIO



6.1.1.1 MMIO

The MMIO mappings are shown in Figure 13.

Figure 13. Physical Address Space - MMIO



By default, CPU core reads targeting the **Boot Vector** range (FFFFFFFFh-FFFE0000h) are sent to the Legacy Bridge, and write accesses target DRAM. For secure SKU's, reads targeting the Boot Vector are decoded and routed to a Secure Root of Trust Boot ROM. For non-secure SKU's, reads targeting this region are routed to a boot SPI flash device connected to the Legacy Bridge.

Upstream writes from the I/O fabric to the **Local APIC** range (FEE00000h-FEEFFFFFh) are sent to the CPU core's APIC.

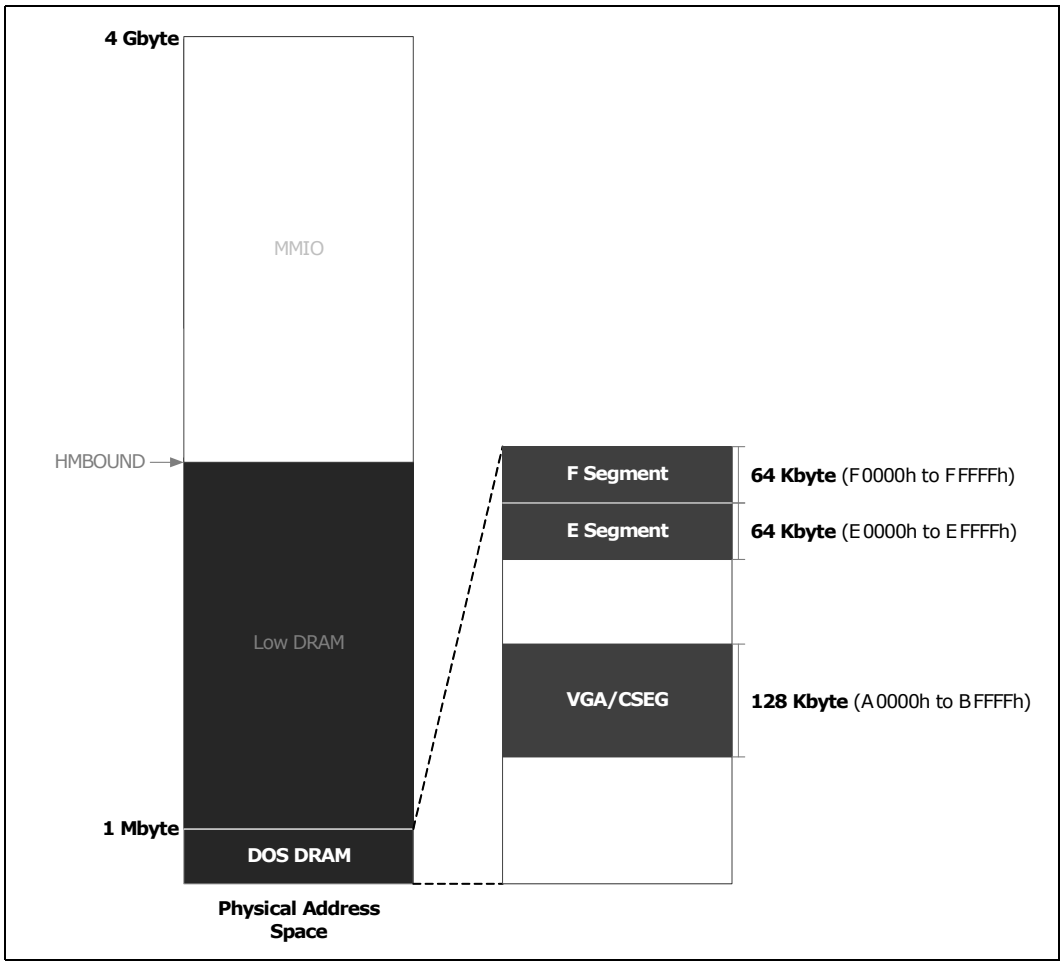
Accesses in the 256 Mbyte **PCI ECAM** range starting at HECREG generate enhanced PCI configuration register accesses when enabled (HECREG.EC_ENABLE). Unlike traditional memory writes, writes to this range are non-posted when enabled. See Chapter 5.0, "Register Access Methods" for more details.

All other downstream accesses in the MMIO range are decoded based on PCI resource allocations. The subtractive agent (for unclaimed accesses) is the I/O Fabric. The I/O Fabric returns an UNSUPPORTED REQUEST for unclaimed accesses.

6.1.1.2 DOS DRAM

The DOS DRAM is the memory space below 1 MByte. In general, accesses from a processor targeting DOS DRAM target system DRAM. Exceptions are shown in Figure 14.

Figure 14. Physical Address Space - DOS DRAM



Processor writes to the 64 Kbyte (each) **E** and **F** segments (E0000h-EFFFFh and F0000h-FFFFFh) always target DRAM. The HMISC2 register is used to direct CPU core reads in these two segments to DRAM or the I/O fabric (MMIO).

CPU core accesses to the 128 Kbyte **VGA/CSEG** range (A0000h-BFFFFh) can target DRAM or the MMIO space depending on the setting of HMISC2.ABSEG_IN_DRAM. When targeting MMIO space, requests are sent to the PCIe* port if legacy VGA is enabled in the PCIe controller.

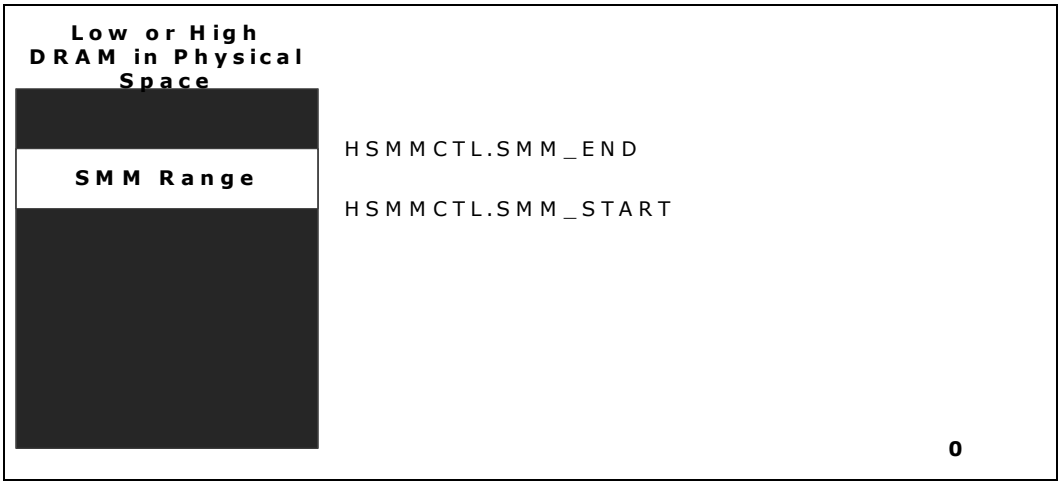
6.1.1.3 Additional Mappings

There is one additional mapping available in the Host Bridge:

- SMM range

Figure 15 shows these mappings.

Figure 15. Physical Address Space - SMM Range



SMI handlers running on a CPU core execute out of SMRAM. To protect this memory from non-CPU core access, the **SMM Range** (HSMMCTL.SMM_START - HSMMCTL.SMM_END) may be programmed anywhere in low DRAM space (1 Mbyte aligned). This range only allows accesses from the CPU core while in SMM.

6.1.2 MMIO Map

Memory accesses targeting MMIO are routed by the programmed PCI ranges.

Fixed MMIO is claimed by the Legacy Bridge. The default regions are listed below. Movable ranges are not shown. See the register maps of all Legacy Bridge components for details.

Table 50. Fixed Memory Ranges in the Legacy Bridge

Device	Start Address	End Address	Comments
Low BIOS (Flash Boot)	000E0000h	000FFFFFh	Starts 128 Kbyte below 1 Mbyte; Firmware/ BIOS
I/O APIC	FEC00000h	FEC00040h	Starts 20 Mbyte below 4 Gbyte
HPET	FED00000h	FED003FFh	Starts 19 Mbyte below 4 Gbyte
High BIOS/Boot Vector	FFFE0000h	FFFFFFFFh	Starts 128 Kbyte below 4 Gbyte; Firmware/ BIOS

PCI devices may also claim memory resources in MMIO space. For details see each device's interface chapter.

Warning: Variable memory ranges should not be set to conflict with other memory ranges. There may be unpredictable results if the configuration software allows conflicts to occur. Hardware does not check for conflicts.

6.2 I/O Address Space

There are 64 Kbyte + 3 bytes of I/O space (0h-10002h) for accessing I/O registers. Most I/O registers exists for legacy functions in the Legacy Bridge or for PCI devices, while some are claimed by the Host Bridge for the PCI configuration space access registers.

6.2.1 Host Bridge I/O Map

The Host Bridge claims I/O transactions for VGA/Extended VGA found in the display/graphics interface. It also claims the two 32-bit registers at port CF8h and CFCh used to access PCI configuration space.

6.2.2 I/O Fabric I/O Map

6.2.2.1 Legacy Bridge Fixed I/O Address Ranges

Table 51 shows the fixed I/O space ranges seen by a processor.

Table 51. Fixed I/O Ranges in the Legacy Bridge

Device	I/O Address	Comments
8259 Master	20h-3Dh	
8254s	40h-43h, 50h-53h	
NMI Controller	61h, 63h, 65h, 67h	
RTC	70h-73h	
Scratch Pad	80h-83h	
8259 Slave	A0h-BDh	
Reset Control	CF9h	Overlaps PCI I/O registers

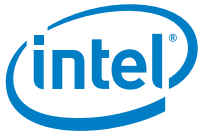
6.2.2.2 Variable I/O Address Ranges

Table 52 shows the variable I/O decode ranges. They are set using base address registers (BARs) or other similar means. Plug-and-play (PnP) software (PCI/ACPI) can use their configuration mechanisms to set and adjust these values.

Warning: The variable I/O ranges should not be set to conflict with other I/O ranges. There may be unpredictable results if the configuration software allows conflicts to occur. Hardware does not check for conflicts.

Table 52. Movable I/O Ranges Decoded by PCI Devices on the I/O Fabric

Device	Size (bytes)	Target
ACPI Power Management	16	PM1BLK: PCI[B:0,D:31,F:0] + 48h
ACPI General Purpose Event 0	64	GPE0BLK: PCI[B:0,D:31,F:0] + 4Ch
GPIO	128	GBA: PCI[B:0,D:31,F:0] + 44h
Watchdog Timer	64	WDTBA: PCI[B:0,D:31,F:0] + 84h
ACPI Processor Block	16	PMBA: Port[0x04] + 70h
SPI DMA Block	16	SPI_DMA_BAR: Port[0x04] + 7Ah



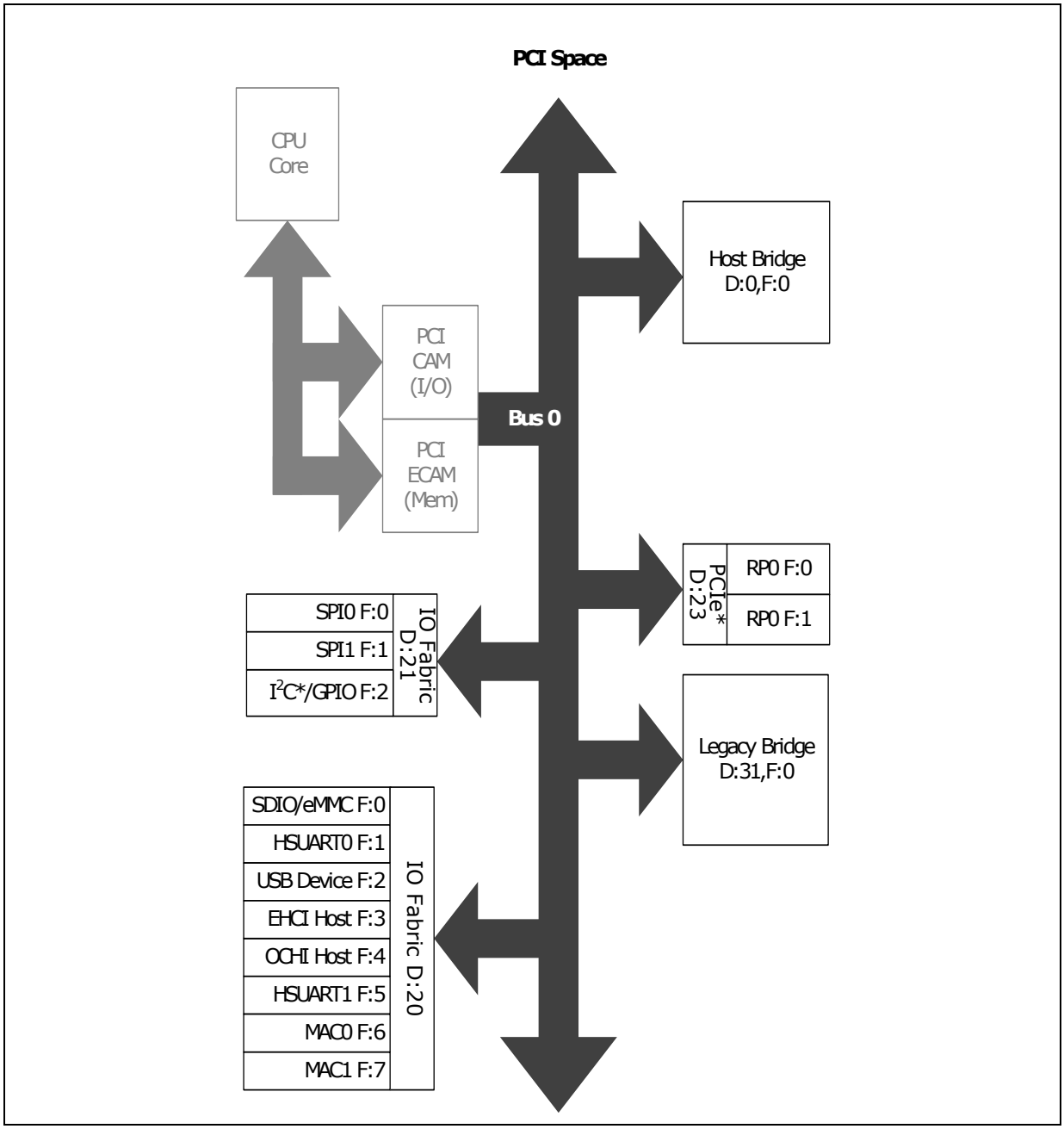
6.3 PCI Configuration Space

All PCI devices/functions are shown below.

Table 53. PCI Devices and Functions

Bus	Device	Function	Device Description	Function Description
0	0	0	Host Bridge	
	20	0	I/O Fabric	SDIO / eMMC
		1		HS-UART 0
		2		USB 2.0 Device
		3		USB EHCI
		4		USB OHCI
		5		HS-UART 1
		6		10/100 Ethernet MAC 0
		7		10/100 Ethernet MAC 1
	21	0	I/O Fabric	SPI 0
		1		SPI 1
		2		I^2C* / GPIO
	23	0	PCI Express*	Root Port 0
		1		Root Port 1
	31	0	Legacy Bridge	Legacy Components

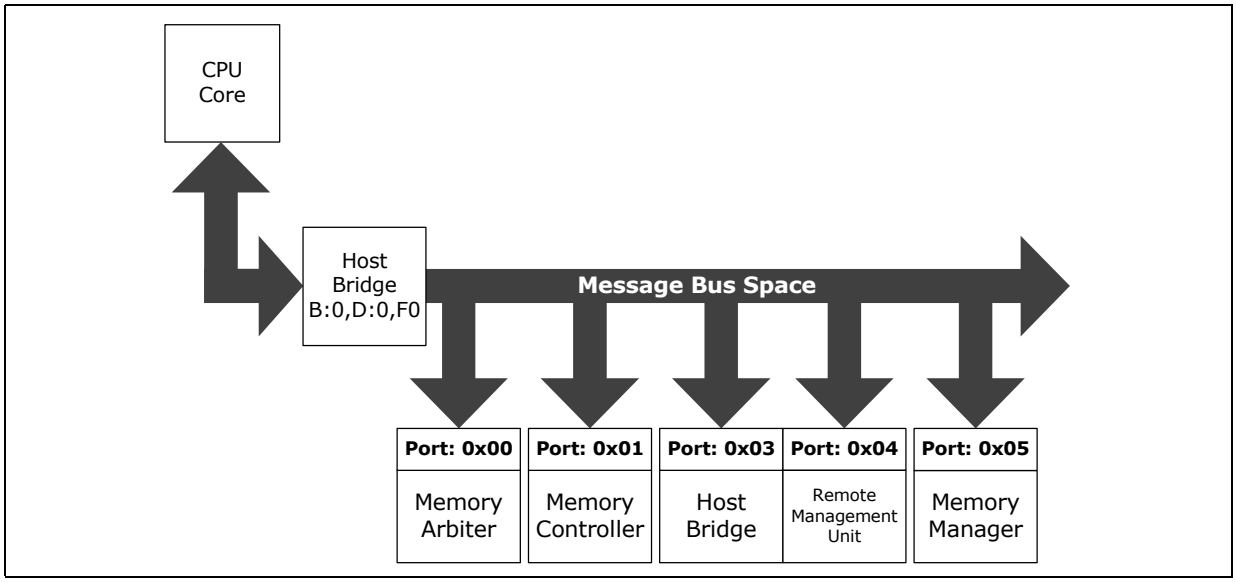
Figure 16. Bus 0 PCI Devices and Functions



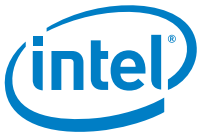
6.4 Message Bus Space

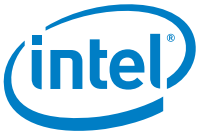
Message bus space provides access to different units within the host bridge. These units are useful in configuring the memory map, power management, and more.

Figure 17. Message Bus with PCI Space



§ §





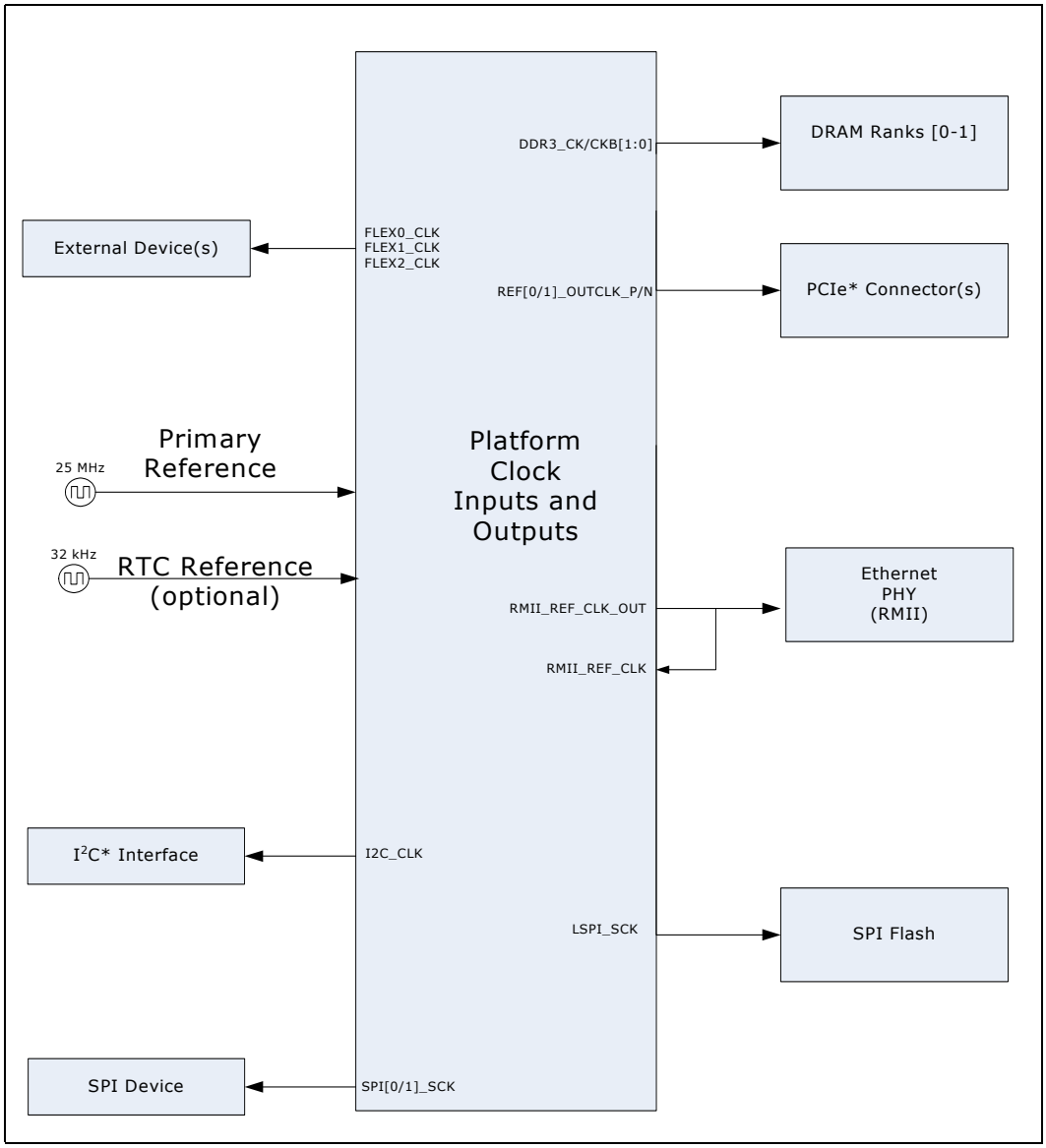
7.0 Clocking

The SoC has a variable frequency, multiple clock domain, and multiple power plane clocking system. This clock architecture achieves a low power clocking solution that supports the various clocking requirements of the different IPs on the SoC. This is achieved by using an Integrated Clock module (iClock) that supplies the clocks to the entire platform.

7.1 Clocking Features

The SoC provides a complete system clocking solution through integrated clocking. All the required platform clocks are provided by the SoC using only one input: a 25 MHz primary reference for the integrated clock block. An optional 32 KHz reference for the Real Time Clock (RTC) block may be provided if required.

Figure 18. SoC Platform Clocking



The references required for the various interface PLLS (e.g., USB/PCIe*) and the processor are internally generated by the Integrated Clocking unit.

7.2 Platform/System Clock Domains

The SoC contains multiple clock domains to support its various interfaces. **Table 54** and **Table 55** summarize the different clock inputs and outputs in the system.

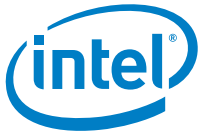


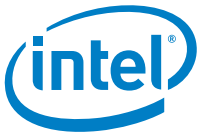
Table 54. Intel® Quark SoC X1000 Clock Inputs

Clock Domain	Signal Name	Frequency	Usage/Description
Main	XTAL_IN XTAL_OUT	25 MHz	25 MHz reference for the iCLK PLL
RTC	RTCX1 RTCX2	32 kHz	RTC Crystal I/O for RTC block. This clock is optional and may be generated internally by the iCLK PLL.
Ethernet PHY	RMII_REF_CLK	50 MHz	RMII 50MHz Clock This clock is a loopback of RMII_REF_CLK_OUT
JTAG	TCK	25 MHz	JTAG Test Clock

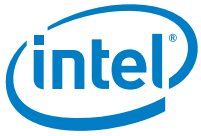
Table 55. Intel® Quark SoC X1000 Clock Outputs

Clock Domain	Signal Name	Frequency	Usage/Description
DDR	DDR3_CK[1:0] DDR3_CKB[3:0]	400 MHz	Drives the Memory ranks 0-1. Data rate is 2x the clock rate.
PCI Express*	REF[0/1]_OUTCLK_N REF[0/1]_OUTCLK_P	100 MHz	Differential Clocks supplied to external PCI Express* devices
Flex Clocks	FLEX0_CLK FLEX1_CLK FLEX2_CLK	33 MHz 33 MHz 48 MHz	Output clock for External devices
Legacy SPI	LSPI_SCK	20 MHz	Clock for external SPI Flash
SPI	SPI[0/1]_SCK	25 MHz	SPI serial clocks
Ethernet PHY	RMII_REF_CLK_OUT	50 MHz	Reference clock for RMII interface
I^2C*	I2C_CLK	400 kHz	I^2C clocks
SD	SD_CLK	50 MHz	SD Clock
Main	CKSYS25OUT	25 MHz	25 MHz Oscillator Output

§ §



October 2013
Document Number: 329676-001US



8.0 Power Management

This chapter provides information on the following power management topics:

- ACPI States
- Processor Core
- PCI Express*

8.1 Power Management Features

- ACPI 3.0 specification support
- ACPI Processor C States (C0, C1, and C2)
- ACPI Sleep State Support (S0, S3, S4, and S5)
- PCI Express L0, L1, L2, and L3

8.2 ACPI Supported States

The ACPI states supported by the processor are described in this section.

8.2.1 S-State Definition

8.2.1.1 S0 - Full On

This is the normal operating state of the processor. In S0, the core processor transitions in and out of the various processor C-States.

Note: The processor core does not support P-states.

8.2.1.2 S3 - Suspend to RAM (Standby)

S3 is a suspend state in which the core power planes of the processor are turned off and the suspend wells remain powered.

- All power wells are disabled, except for the suspend and RTC wells.
- The core processor's macro-state is saved in memory.
- Memory is held in self-refresh and the memory interface is disabled, except the CKE pin as it is powered from the memory voltage rail. CKE is driven low.

8.2.1.3 S4 - Suspend to Disk (Hibernate)

S4 is a suspend state in which most power planes of the processor are turned off, except for the suspend and RTC well. In this ACPI state, system context is saved to the SD card.

Note: This is a software based state that is the same as S5 to hardware. On S4 entry, the system saves the entire contents of data off to NVRAM. On S4 resume, the system restores the entire contents of memory after performing the a typical S5-S0 boot.

Key features:

- No activity is allowed.
- All power wells are disabled, except for the suspend and RTC well.

8.2.1.4 S5 - Soft Off

From a hardware perspective the S5 state is identical to the S4 state. The difference is purely software; software does not write system context to OS storage when entering S5.

8.2.2 System States

Table 56. General Power States for System

States/Sub-states	Legacy Name / Description
G0/S0/C0	**FULL ON:** CPU operating. Individual devices may be shut down to save power. The different CPU operating levels are defined by Cx states.
G0/S0/Cx	**Cx State:** CPU manages C-state itself.
G1/S3	**Suspend-To-RAM (STR):** The system context is maintained in system DRAM, but power is shut to non-critical circuits. Memory is retained, and refreshes continue. All external clocks are shut off; RTC clock and internal ring oscillator clocks are still toggling.
G1/S4	**Suspend-To-Disk (STD):** The context of the system is maintained on the disk. All power is shut down except power for the logic to resume.
G2/S5	**Soft-Off:** System context is not maintained. All power is shut down except power for the logic to restart. A full boot is required to restart. A full boot is required when waking.
G3	**Mechanical OFF.** System content is not maintained. All power shutdown except for the RTC. No "Wake" events are possible, because the system does not have any power. This state occurs if the user removes the batteries, turns off a mechanical switch, or if the system power supply is at a level that is insufficient to power the "waking" logic. When system power returns, transition depends on the state just prior to the entry to G3.

Table 57 shows the transition rules among the various states. Note that transitions among the various states may appear to temporarily transition through intermediate states. These intermediate transitions and states are not listed in the table.

Table 57. ACPI PM State Transition Rules (Sheet 1 of 2)

Present State	Transition Trigger	Next State
G0/S0/C0	P_LVL2 Read	G0/S0/C2
	PM1C.SLP_EN bit set	G1/Sx or G2/S5 state (specified by PM1C.SLP_TYPE)
	Power Button Override	G2/S5
	Mechanical Off/Power Failure	G3
G0/S0/C2	C2 break events which include: MSI, Legacy Interrupt	G0/S0/C0
	Power Button Override	G2/S5
	Resume Well Power Failure	G3

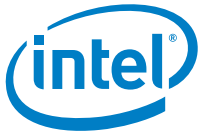


Table 57. ACPI PM State Transition Rules (Sheet 2 of 2)

Present State	Transition Trigger	Next State
G1/S3, G1/S4	Any Enabled Wake Event	G0/S0/C0
	Power button Override	G2/S5
	Resume Well Power Failure	G3
G2/S5	Any Enabled Wake Event	G0/S0/C0
	Power Failure or Removal	G3
G3	Power Returns	Option to go to S0/C0 (reboot) or G2/S5 (stay off until power button pressed or other enabled wake event) or G1/S4 (if system state was S4 prior to the power failure). Some wake events are preserved through a power failure.

8.2.3 Processor Idle States

Table 58. Processor Core/ States Support

State	Description
C0	Active mode, processor executing code
C1	AutoHALT state
C2	Stop Grant state

8.2.4 Integrated Memory Controller States

Table 59. Main Memory States

States	Description
Powerup	CKE asserted. Active mode.
Precharge Powerdown	CKE de-asserted (not self-refresh) with all banks closed.
Active Powerdown	CKE de-asserted (not self-refresh) with at least one bank active.
Self-Refresh	CKE de-asserted using device self-refresh

8.2.5 PCIe* States

Table 60. PCIe* States

States	Description
L0	Full on – Active transfer state
L0s	First Active Power Management low power state – Low exit latency
L1	Lowest Active Power Management - Longer exit latency
L3	Lowest power state (power-off) – Longest exit latency

8.2.6 Interface State Combinations

Table 61. G, S and C State Combinations

Global (G) State	Sleep (S) State	Processor Core (C) State	Processor State	System Clocks	Description
G0	S0	C0	Full On	On	Full On
G0	S0	C1	Auto-Halt	On	Auto-Halt
G0	S0	C2	Stop Grant	On	Stop Grant
G1	S3	Power Off		Off except RTC & internal ring OSC	Suspend to RAM
G1	S4	Power Off		Off except RTC & internal ring OSC	Suspend to Disk
G2	S5	Power Off		Off except RTC & internal ring OSC	Soft Off
G3	NA	Power Off		Power Off	Hard Off

8.3 Processor Core Power Management

When the processor is not executing code, it is idle. A low-power idle state is defined by ACPI as a C-state. In general, lower power C-states have longer entry and exit latencies.

8.3.1 Low-Power Idle States

When the processor core is idle, low-power idle states (C-states) are used to save power. More power savings actions are taken for numerically higher C-state. However, higher C-states have longer exit and entry latencies.

8.3.1.1 Clock Control and Low-Power States

The processor core supports low power states at core level. States for processor core include Normal (C0), Auto-Halt (C1) and Stop Grant (C2).

Transition to processor core power states higher than C1 are triggered by initiating a P_LVLx (P_LVL2) I/O read.

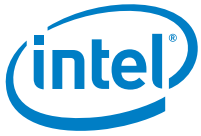
The Cx state ends due to a break event. Based on the break event, the processor returns the system to C0. The following are examples of such break events:

- Any unmasked interrupt goes active
- Any internal event that will cause an NMI or SMI_B
- CPU Pending Break Event (PBE_B)
- MSI

8.3.2 Processor Core C-States Description

8.3.2.1 Core C0 State

The normal operating state of a core where code is being executed.



8.3.2.2 Core C1 State

C1 is a low power state entered when the core executes a HLT instruction.

A System Management Interrupt (SMI) handler returns execution to either Normal state or the C1 state. See the *Intel® 64 and IA-32 Architecture Software Developer's Manual, Volume 3A/3B: System Programmer's Guide* for more information.

While the core is in C1 state, it still processes snoops.

8.3.2.3 Core C2 State

C2 is entered when the processor reads the P_LVL2 register to trigger a transition from C0 to C2. While the core is in the C2 state, it processes snoops.

An interrupt or a reset is required to exit the C2 state and return to the C0 state.

8.4 Memory Controller Power Management

The main memory is power managed during normal operation and in low-power ACPI Cx states.

8.4.1 Disabling Unused System Memory Outputs

When a given rank is not populated, the corresponding chip select and CKE signals are not driven.

At reset, all rows must be assumed to be populated, until it can be proven that they are not populated. This is due to the fact that when CKE is tri-stated the memory module is not guaranteed to maintain data integrity.

8.4.2 DRAM Power Management and Initialization

The SoC implements extensive support for power management on the SDRAM interface. There are four SDRAM operations associated with the Clock Enable (CKE) signals, SRE, SRX, PDE and PDX, which the SDRAM controller supports. The SoC drives two CKE pins to perform these operations.

8.4.2.1 Initialization Role of CKE

During power-up, CKE is the only input to the SDRAM that is recognized (other than the DDR3 reset pin) once power is applied. It must be driven LOW by the DDR controller to make sure the SDRAM components float DQ and DQS during power-up.

CKE signals remain LOW (while any reset is active) until the BIOS writes to a configuration register. Using this method, CKE is guaranteed to remain inactive for much longer than the specified 200 micro-seconds after power and clocks to SDRAM devices are stable.

8.4.2.2 Dynamic Self-Refresh

When Dynamic Self-Refresh (SR) is enabled, via DPMC0.DYNSREN, the Memory Controller places the SDRAM in SR mode when the following conditions are true:

1. No requests are pending
2. Internal Request Status is low priority
3. No SR exit requests from the DDRIO (for RCOMP updates)

If one of the above conditions change prior to the SR Entry command being sent to the DRAM the process is terminated.

When Dynamic SR is enabled the Memory Controller exits SR mode when one of the following is true:

1. Requests are pending and the Internal Request Status is normal or urgent
2. A SR exit request from the DDRIO

8.4.2.3 Dynamic Power Down Operation

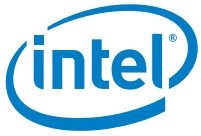
Dynamic power-down of memory is employed during normal operation. Based on idle conditions, a given memory rank may be powered down. The Memory Controller implements aggressive CKE control to dynamically put the DRAM devices in a power down state. The Memory Controller can be configured to put the devices in active power down (CKE de-assertion with open pages) or precharge power down (CKE de-assertion with all pages closed). Precharge power down provides greater power savings but has a bigger performance impact, since all pages will first be closed before putting the devices in power down mode.

If dynamic power-down is enabled, all ranks are powered up before doing a refresh cycle and all ranks are powered down at the end of refresh.

8.4.2.4 Functional Clock Gating

The Memory Controller has internal clock gating for the majority of its clocked logic. When enabled the clock gating is activated when all inputs are inactive and all commands are complete and DDR3 timing trackers are flushed. When Dynamic SR is enabled, clock gating is only applied when the SDRAM is in Self-Refresh.

§ §



9.0 Power Up and Reset Sequence

This chapter provides information on the following topics:

- Power up and down sequences, including wake events
- Reset behavior

9.1 Intel® Quark SoC X1000 System States

9.1.1 System Sleeping States Control (S-States)

The SoC supports the S0, S3, S4, and S5 sleep states. S4 and S5 states are identical from a hardware perspective.

The SoC integrates a Power Management Controller (PMC). No external power controller IC is required.

The SoC sleep states are described in Chapter 8.0, "Power Management".

9.2 Power Up and Down Sequences

9.2.1 Power Up, Wake and Reset Overview

SoC power up is dependent on two supplies:

- VCC3P3_S5, which is generated from AC power
- VCCRTC_3P3, which powers the RTC well only

VCCRTC_3P3 is derived directly from VCC3P3_S5, if present. Otherwise it can be driven by a coin-cell battery.

- In the case where the coin-cell battery is present but not AC power, only the RTC well is powered up. The SoC can move to state G3 only. The SoC can subsequently be transitioned to state S4/S5 by applying AC power.
- In the case where AC power is present but there is no coin-cell battery, power up is initiated directly by the ramping of the VCC3P3_S5 supply. The SoC transitions directly to state S4/S5.

Subsequent transition from S4/S5 to S0 is governed by activity on the power button pin, PWR_BTN:

- If PWR_BTN is strapped low (auto power button mode) when AC power is applied, the SoC transitions directly to S0 from S4/S5 via a transitional S3 state.
- If PWR_BTN is high when AC power is applied, the SoC transitions to S4/S5 only. A subsequent falling edge on PWR_BTN, with the low value being maintained for 2.5ms or more, is required to initiate a transition to S0 via the transitional S3 state.

Once in state S0 the SoC can be put to sleep, i.e., transitioned to sleep states S3 or S4/S5, through appropriate settings of the Legacy Bridge ACPI registers PM1C.SLPTYPE and PM1C.SLPEN.

A wake event is defined as a transition from state S3 to state S0. The chip can be woken up via a number of mechanisms including specific register settings, or by asserting specific SoC pins. A watchdog function in the Legacy Bridge can also trigger a wake event.

In auto power button mode, if the SoC is placed in sleep state S4/S5, the system can only be woken by the removal and reapplication of AC power. Since PWR_BTN is low it will power up and transition directly back to S0 as described in the power up sequence.

There are two classes of reset associated with Intel® Quark SoC X1000:

- A ***cold reset*** means transitioning from S0 to S4/S5 and back to S0 again, independent of the PWR_BTN value. This can only be initiated from state S0 through the register RSTC.COLD_RST. All registers except those driven by the RTC supply are effectively reset.
- A ***warm reset*** resets CPU and peripheral blocks without the removal of the power supplies. This can be initiated via a write to the register RSTC.WARM_RST or by asserting the SoC pin, RESET_BTN (active low). It can occur only in state S0 and after reset the SoC remains in state S0. RTC well and suspend well registers are left unaffected.

A **cold boot** is the sequence where AC power is applied followed by an immediate transition to S0 using the PWR_BTN signal or directly in auto power button mode.

Catastrophic shutdown can be carried out by holding PWR_BTN low for at least 3s. This results in a direct return to the S4/S5 state. It can also be initiated by software under specific error conditions.

The following sections provide more detail on these power-related functions.

9.2.2 RTC Power Well Transition: G5 to G3 State Transition

The transition to the G3 state is initiated when VCCRTC_3P3 is ramped. The sequence is as follows:

1. VCCRTC_3P3 ramps. RTCRST_B should be low.
2. The SoC starts the real time clock oscillator.
3. A minimum of t1 units after VCCRTC_3P3 ramps external circuitry deasserts RTCRST_B. The system is now in the G3 state.

Figure 19. RTC Power Well Timing Diagrams

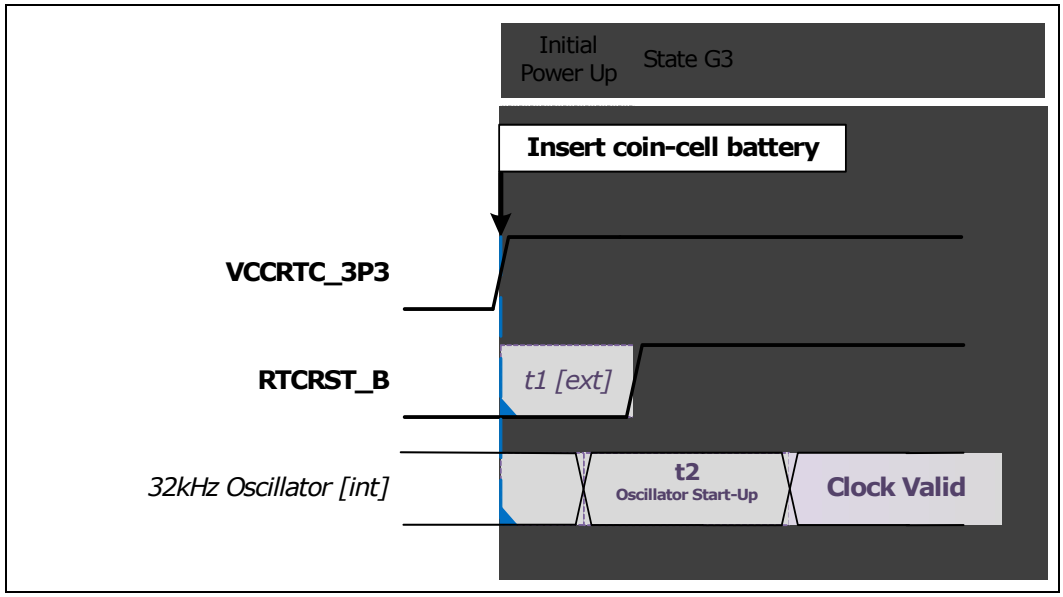


Table 62. RTC Power Well Timing Parameters

Parameter	Description	Min	Max	Units	Notes
t1	VCCRTC_3P3 to RTCRST# deassertion	9	N/A	ms	1
t2	Oscillator Startup Time	TBD	TBD	s	2,~~3~~

Notes:
1. This delay is typically created from an RC circuit.
2. The oscillator startup times are component and design specific. A crystal oscillator can take as long as 2 s to reach a large enough voltage swing. Whereas, a silicon oscillator can have startups times <10 ms.

9.2.3 Power-Up Sequence without G2/G3: No Coin-Cell Battery

This sequence must be adhered to in cases where one of the following conditions apply:

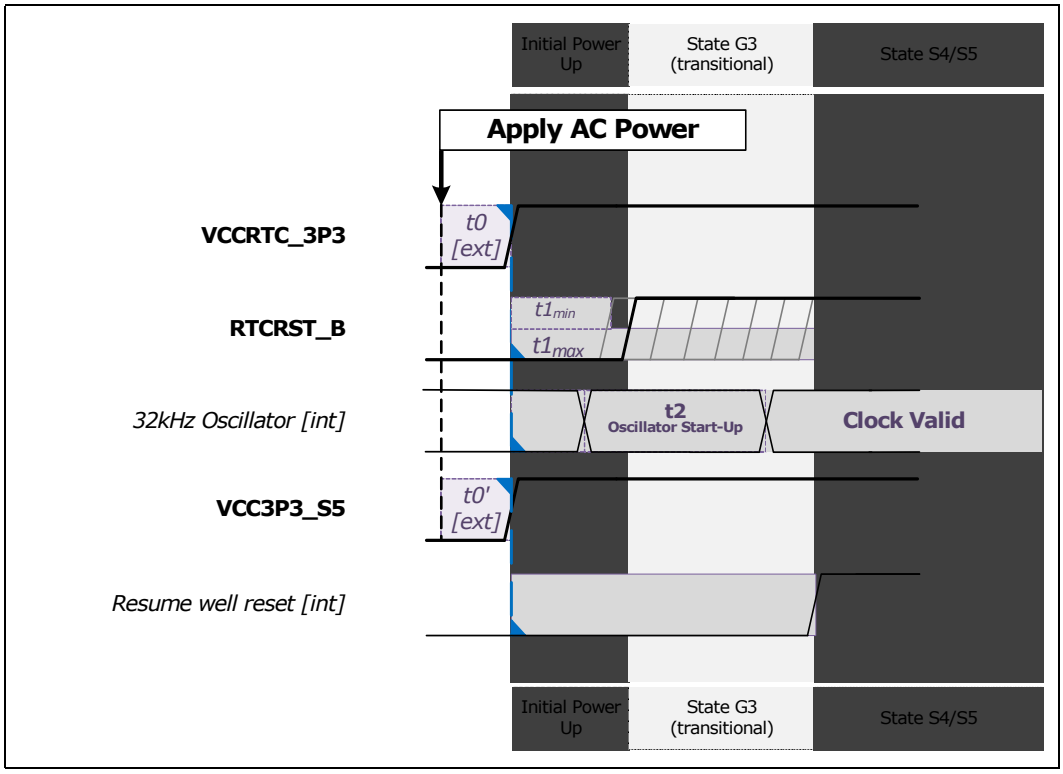
1. The system does not implement an RTC battery (coin cell) or a main battery.
2. The coin cell is drained with no main or a dead main battery.
3. No coin cell implemented or dead coin cell and main battery is being swapped.

AND one of the following conditions also applies:

1. The platform does not implement a power button to initiate a sequence to S0, and AC power becomes available.
2. The platform does use a power button, but the default first sequence when power is available is entry into S0.

In these cases, the relative timing between RTC and suspend wells becomes important. The key point is that, as well as a minimum time, there is a maximum time by which RTCRST_B must be deasserted. It must happen before an internal reset associated with the suspend well is deasserted. This is shown in Figure 20.

Figure 20. Power-Up Sequence without G2/G3



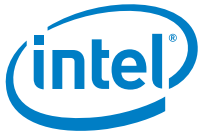
9.2.4 AC Power Applied: G3 to S4/S5 State Transition

The timings shown in Figure 21 and Table 63 occur when AC power is applied. The following occurs:

1. The supplies VCC3P3_S5 and VCC1P5_S5, generated by the platform regulator from AC power, start ramping. [t1, t2]
2. VCC3P3_S5 drives an internal S5 LDO regulator which generates internal 1.0V and 1.8V supplies. These are driven off chip via the OVOUT_1P0_S5 and OVOUT_1P8_S5 pins. [t3, t4]
3. When the internal supplies are stable an internal S5 power-good signal is generated. [t5]
4. In parallel OVOUT_1P0_S5 and OVOUT_1P8_S5 are monitored on the platform. When stable, a platform S5 power-good signal should be generated and applied to the S5_PG pin. [t6]
5. When both the internal S5 power-good and S5_PG signals are asserted the system is in state S4/S5.

9.2.5 Using PWR_BTN: Transition from S4/S5 to S0

1. The internal power management controller detects an event: either PWR_BTN is actively brought low and remains low for at least 2.5ms [t7], or it is tied low for an automatic start from the S5 state (auto power button mode). This initiates the transition from S4/S5 to S0.
2. After a PLL settling time the internal PMC generates a switch enable S3_3V3_EN which should be used to switch on of the platform S3 supply VCC3P3_S3 [t8, t8’].



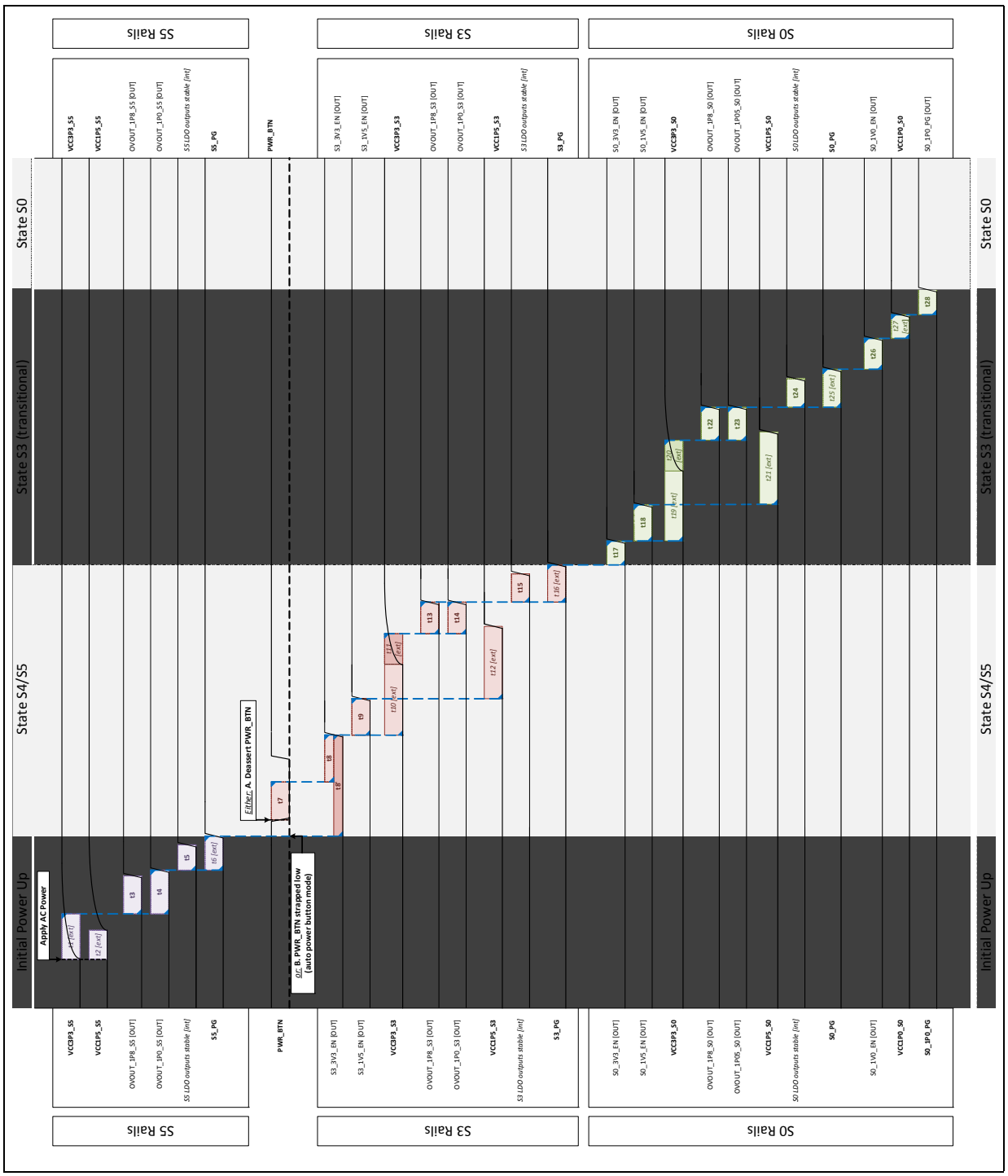
3. This is followed by the assertion of S3_1V5_EN to switch on VCC1P5_S3 [t9].

   Note: The switch enables are active high and should be used to drive PFET switches or regulator enables on the platform.

   Note: The switch enables are staggered because of SoC rail sequencing constraints.

4. After a switch-on time [t10] the S3 supply VCC3P3_S3 starts ramping [t11]. This is followed in turn by VCC1P5_S3 [t12].
5. VCC3P3_S3 drives an internal S3 LDO regulator which generates internal 1.0V and 1.8V supplies. These are driven off chip via the OVOUT_1P0_S3 and OVOUT_1P8_S3 pins [t13, t14].
6. When the internal supplies are stable an internal S3 power-good signal is generated [t15].
7. In parallel OVOUT_1P0_S3 and OVOUT_1P8_S3 are monitored on the platform. When stable, a platform S3 power-good signal should be generated and applied to the S3_PG pin [t16].
8. When both the internal S3 power-good and S3_PG signals are asserted, **the *SoC* is in a transitional S3 state.**
9. The PMC now generates the switch enable S0_3V3_EN which should be used to switch on the S0 supply VCC3P3_S0 [t17].
10. This is followed by the assertion of S0_1V5_EN to switch on VCC1P5_S0 [t18].
11. After a switch-on time [t19] the S0 supply VCC3P3_S0 starts ramping [t20]. This is followed in turn by VCC1P5_S0 [t21].
12. VCC3P3_S0 drives an internal S0 LDO regulator which generates internal 1.05V and 1.8V supplies. These are driven off chip via the OVOUT_1P05_S0 and OVOUT_1P8_S0 pins [t22, t23].
13. When the internal supplies are stable an internal S0 power-good signal is generated [t24].
14. In parallel OVOUT_1P05_S0 and OVOUT_1P8_S0 are monitored on the platform. When stable, a platform S0 power-good signal should be generated and applied to the S0_PG pin [t25].
15. When both the internal S0 power-good and S0_PG signals are asserted, the internal PMC generates the switch enable S0_1V0_EN [t26]. This is used to switch on the main S0 1.0V supply VCC1P0_S0.
16. After a switch-on time the S0 supply VCC1P0_S0 starts ramping [t27].
17. Once VCC1P0_S0 is stable the platform should generate an active high power good signal which is applied to the S0_1P0_PG pin [t28].
18. All supplies are now on and **the system is in state S0**.

Figure 21. Power Up Sequence



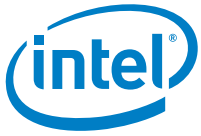


Table 63. S4/S5 to S0 Timing Parameters

Time	Internal (SoC)/ External (Platform)	Description	Min	Max	Units	Notes
t0	N/A	Apply AC power	N/A	N/A	N/A	Assume SoC is in state G3
t1	*ext*	*VCC3P3_S5 (Ramping)*				
t2	*ext*	*VCC1P5_S5 (Ramping)*				
t3	int	OVOUT_1P8_S5 (Delay)				
t4	int	OVOUT_1P0_S5 (Delay)				
t5	int	Internal S5 power good asserted				
t6	*ext*	*Assertion of S5_PG*				*1*
t7	int	PWR_BTN debounce time	2.5		ms	Must hold PWR_BTN low for at least this time for falling edge to take effect
t8	int	S3_3V3_EN (Delay)				With respect to PWR_BTN event
t8’	int	S3_3V3_EN (Delay)				auto power button mode: With respect to S5_PG
t9	int	S3_1V5_EN (Delay)	90	1800	µs	Offset from S3_3V3_EN due to SoC rail sequencing requirements
t10	*ext*	*VCC3P3_S3 (Switch Delay)*				
t11	*ext*	*VCC3P3_S3 (Ramping)*				
t12	*ext*	*VCC1P5_S3 (Switch Delay)*				
t13	int	OVOUT_1P8_S3 (Delay)				
t14	int	OVOUT_1P0_S3 (Delay)				
t15	int	Internal S3 power good asserted				
t16	*ext*	*Assertion of S3_PG*				*1*
t17	int	S0_3V3_EN (Delay)				
t18	int	S0_1V5_EN (Delay)				Offset from S3_3V3_EN due to SoC rail sequencing requirements
t19	*ext*	*VCC3P3_S0 (Switch Delay)*				
t20	*ext*	*VCC3P3_S0 (Ramping)*				
t21	*ext*	*VCC1P5_S0 (Switch Delay)*				
t22	int	OVOUT_1P8_S0 (Delay)				
t23	int	OVOUT_1P05_S0 (Delay)				
t24	int	Internal S0 power good asserted				
t25	*ext*	*Assertion of S0_PG*				*1*

Note:
1. Must be asserted *after* internal power good assertions from respective [S5/S3/S0] LDO regulators

Table 63. S4/S5 to S0 Timing Parameters

Time	Internal (SoC)/ External (Platform)	Description	Min	Max	Units	Notes
t26	int	S0_1V0_EN (Delay)				
t27	*ext*	*VCC1P0_S0 (Delay)*				
t28	*ext*	*Assertion of S0_1P0_PG*				

Note:
1. Must be asserted *after* internal power good assertions from respective [S5/S3/S0] LDO regulators

9.2.6 Going to Sleep: Transitions from S0 to S3 or S4/S5

Entry to sleep states (S3, S4/S5) is initiated by any of the following methods:

- Setting the desired type in PM1C.SLPTYPE and setting PM1C.SLPEN in the Legacy Bridge ACPI registers.
- Detection of a catastrophic event causes a direct transition to S4/S5. This can occur when the main power well is on, i.e. in the S0 state. Such an event can be initiated by pressing PWR_BTN low for more than 3s. It can also be initiated by software after detection of a temperature or other error event.

9.2.7 Wake Events: Transition from S3 to S0

Wake events are used to return the system to S0 and can only be initiated in state S3. They are controlled completely by the PMC. Upon exit from sleep states, the PM1S.WAKE bit in the Legacy Bridge ACPI registers will be set.

Table 64 describes these events:

Table 64. Intel® Quark SoC X1000 S3 Wake Events

Event	How Enabled	Description
RTC Alarm	Set both PM1S.RTC and PM1E.RTC in Legacy Bridge	Triggered by RTC asserting IRQ#8 in Legacy Block
Resume GPIOSUS	Set both GPE0STS.GPIO and GPE0EN.GPIO in Legacy Bridge	Generates SCI/SMI via ACPI GPE0 registers in Legacy Bridge
External GPE_B (pin)	Set both GPE0S.EGPE and GPE0E.EGPE in Legacy Bridge	Generates SCI/SMI via General Purpose Event Register in Legacy Bridge
WAKE_B (pin)	N/A	Input to SoC that indicates a PCI Express port wants to wake the system.
Power button	Press PWR_BTN	PWR_BTN press triggers transition to S0
Reset button	Press RESET_BTN	RESET_BTN press triggers transition to S0

9.2.8 System Reset Sequences

There are two types of reset:

- Cold Reset: Results in power cycling of all rails except the RTC well. The entire chip is reset except for the RTC well.
- Warm Reset: Results in a reset without the removal of power. All core logic gets reset. The suspend and RTC wells are not reset. The system remains in state S0.

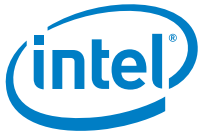


Table 65. SoC Reset Events

Sequence Type	How Initiated
Cold Boot	Switch AC power off and then on again When not in auto power button mode PWR_BTN is asserted (low) for at least 2.5ms
Cold Reset (Internal)	RSTC.COLD_RST bit set
Warm Reset (External)	RESET_BTN pin asserted (low) for at least 2.5ms
Warm Reset (Internal)	RSTC.WARM_RST bit set

9.2.8.1 Cold Boot Sequence

Cold boot happens when AC power is turned off and on again. The power button is used to wake up the system: assert PWR_BTN low for at least 2.5ms. In auto power button mode (PWR_BTN strapped low) the system proceeds straight to state S0 once the AC power is applied, as documented in the power up sequence.

9.2.8.2 Cold Reset Sequence

Cold reset is initiated by the CPU writing to Reset Control Register bit RSTC.COLD_RST in the Legacy Bridge. Cold reset causes a full cycling of power. The chip is transitioned to state S4/S5 and then back to S0, independent of the setting of PWR_BTN. All functions are reset except for those powered from the RTC well.

The Watchdog Timer in the Legacy Bridge can be enabled to generate a cold reset in the event of a timeout event. This is indistinguishable from a cold reset due to RSTC.COLD_RST being set.

9.2.8.3 Warm Reset Sequence (Internal)

Warm reset is initiated by the CPU writing to Reset Control Register bit RSTC.WARM_RST in the Legacy Bridge. Warm reset causes reset of the CPU and peripherals without switching off their power supplies.

The Watchdog Timer in the Legacy Bridge can be enabled to generate a warm reset in the event of a timeout event. This is indistinguishable from a warm reset due to RSTC.WARM_RST being set.

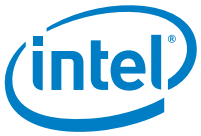
9.2.8.4 Externally Initiated Warm Reset Sequence

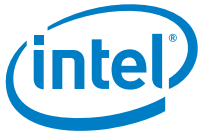
Warm Reset can also be externally initiated by asserting the reset button RESET_BTN. It results in reset without removal of power on any of the supplies.

9.2.9 Handling Power Failures

When the power supply to SoC is removed in a disorderly fashion, either through removal of the coin-cell battery or a failure on the AC supply, a normal cold boot sequence should be initiated.

§ §





10.0 Thermal Management

10.1 Overview

The Intel® Quark SoC X1000 thermal management feature helps in managing the overall thermal profile of the system to prevent overheating and system breakdown. The architecture implements various proven methods of maintaining maximum performance while remaining within the thermal specification.

The thermal management features are:

- On-die thermal sensor
- Supports a hardware trip point and programmable trip points based on the temperature indicated by thermal sensor.

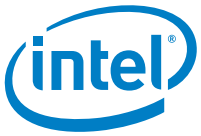
10.2 Thermal Sensor

The SoC provides an on-die Thermal Sensor that can be read via Message Bus registers. The Thermal Sensor provides an 8-bit temperature reading with a resolution of 1 degree Celsius.

To use the Thermal Sensor:

1. It first must be taken out of reset by setting bit 0 of Msg Port 31:R34h to 0 and subsequently enabled via bit 15 of Msg Port 04:RB0h.
2. Once enabled, registers within the Remote Management Unit can be used to configure trip points and read the current temperature value.

§ §



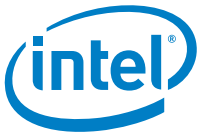
11.0 Processor Core

Processor core features on the Intel® Quark SoC X1000 include:

- Single processor core
- Single Instruction 5-stage pipeline
- 32-bit processor with 32-bit data bus
- Integrated Floating Point Unit
- 16 Kbyte, 4-way shared instruction and data L1 write-back cache
- Integrated local APIC
- Support for IA 32-bit Pentium x86 ISA compatibility
- Supports C0, C1, and C2 states
- Supports Supervisor Mode Execution Protection (SMEP)
- Supports Execute-Disable Page Protection (PAE.XD)

Note: The processor core provides an integrated Local APIC but does not support the IA32_APIC_BASE MSR. As a result, the Local APIC is always globally enabled and the Local APIC base address is fixed at FEE00000h. Attempting to access the IA32_APIC_BASE MSR causes a general protection fault.

§ §



October 2013
Document Number: 329676-001US

12.0 Host Bridge

The Host Bridge is a central hub that routes transactions to and from the Intel® Quark SoC X1000's CPU core, DRAM controller, and other functional blocks. In general, it handles:

- CPU Core Interface: Requests for CPU Core-initiated memory and I/O read and write operations and processor-initiated message-signaled interrupt transactions
- Device MMIO and PCI configuration routing
- Buffering and memory arbitration
- PCI Config and MMIO accesses to host device (0/0/0)

12.1 Embedded SRAM (eSRAM)

The Host Bridge contains an interface to 512KB of on-chip, low latency, embedded SRAM (eSRAM). The eSRAM memory may be used as either 128 x 4KB pages, or in block mode as a single contiguous 512KB block page. The eSRAM pages may be mapped anywhere in the physical address space as a DRAM overlay.

The eSRAM is a volatile memory and functionality is provided to flush eSRAM pages to DRAM as part of entry to an S3 system state. Sections of DRAM overlaid by eSRAM are inaccessible to all system agents.

12.1.1 Initialization

Immediately on coming out of a warm or cold reset, the Host Bridge initializes eSRAM data to 0. While this is taking place the register fields ESRAMPGCTRL*x*.INIT_IN_PROG and ESRAMPGCTRL_BLOCK.BLOCK_INIT_IN_PROG are 1. Software may map and enable eSRAM pages during this time, but accesses to an eSRAM 4KB page will not complete until the page has completed initialization.

12.1.2 Configuration

Once an eSRAM page (4KB page or 512KB block page) is enabled, it may only be flushed or disabled as part of an entry to an S3 system state. In order to re-configure an eSRAM page, the Host Bridge must be warm or cold reset.

12.1.2.1 4KB Page Mode

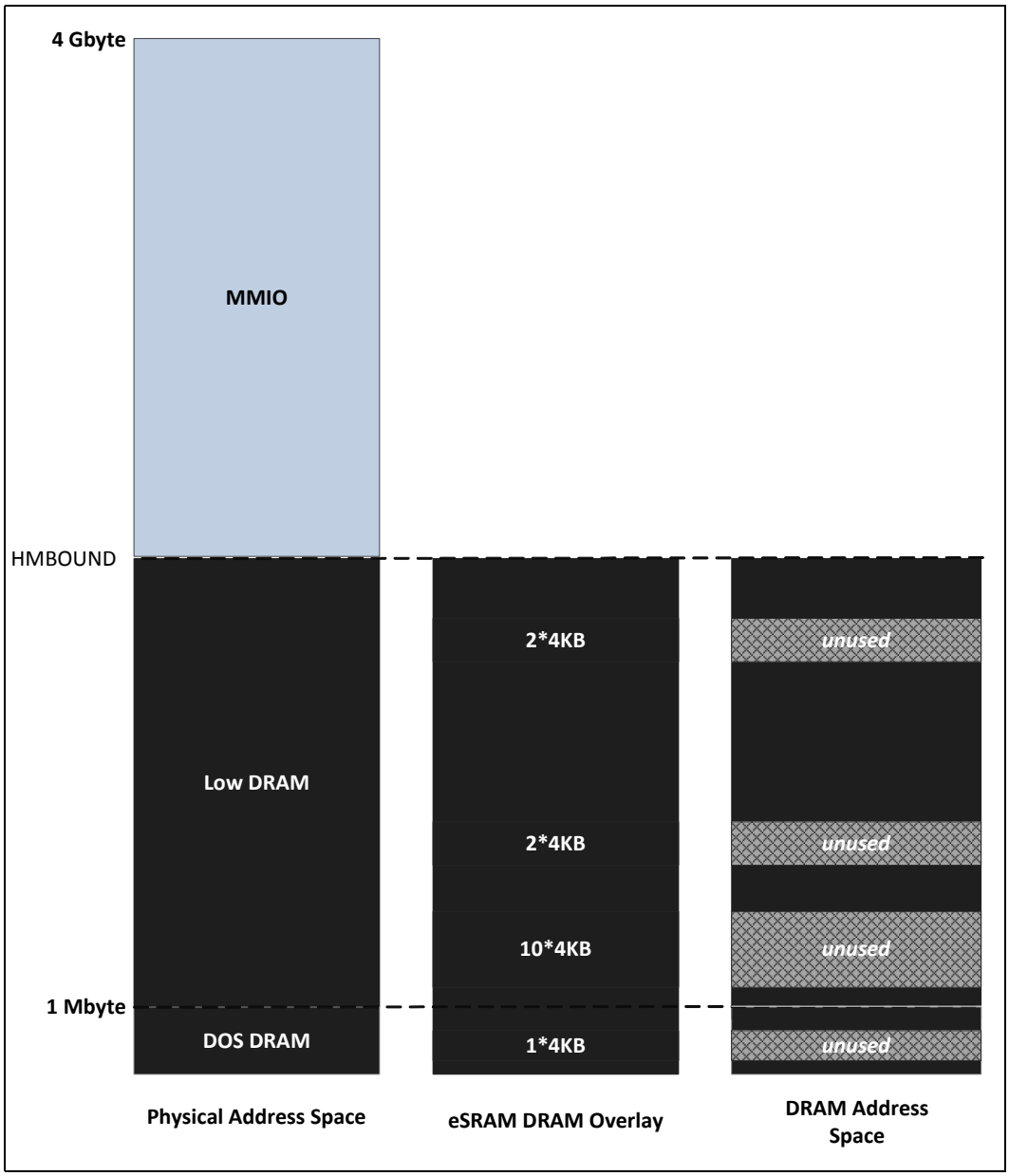
The Host Bridge provides 128 registers (ESRAMPGCTRL*x*) that allow individual configuration of the 128*4KB eSRAM pages. If the block page is already enabled, it is not possible to individually map 4KB pages.

To map and enable 4KB page *x*, the following steps should be followed:

- Set ESRAMPGCTRL*x*.PG_SYSTEM_ADDRESS_4K to the required address value
- Set ESRAMPGCTRL*x*.ENABLE_PG to 1

Software must be careful not to map different eSRAM pages to the same system address. There is no hardware protection against this.

Figure 22. eSRAM 4KB Page Mapping



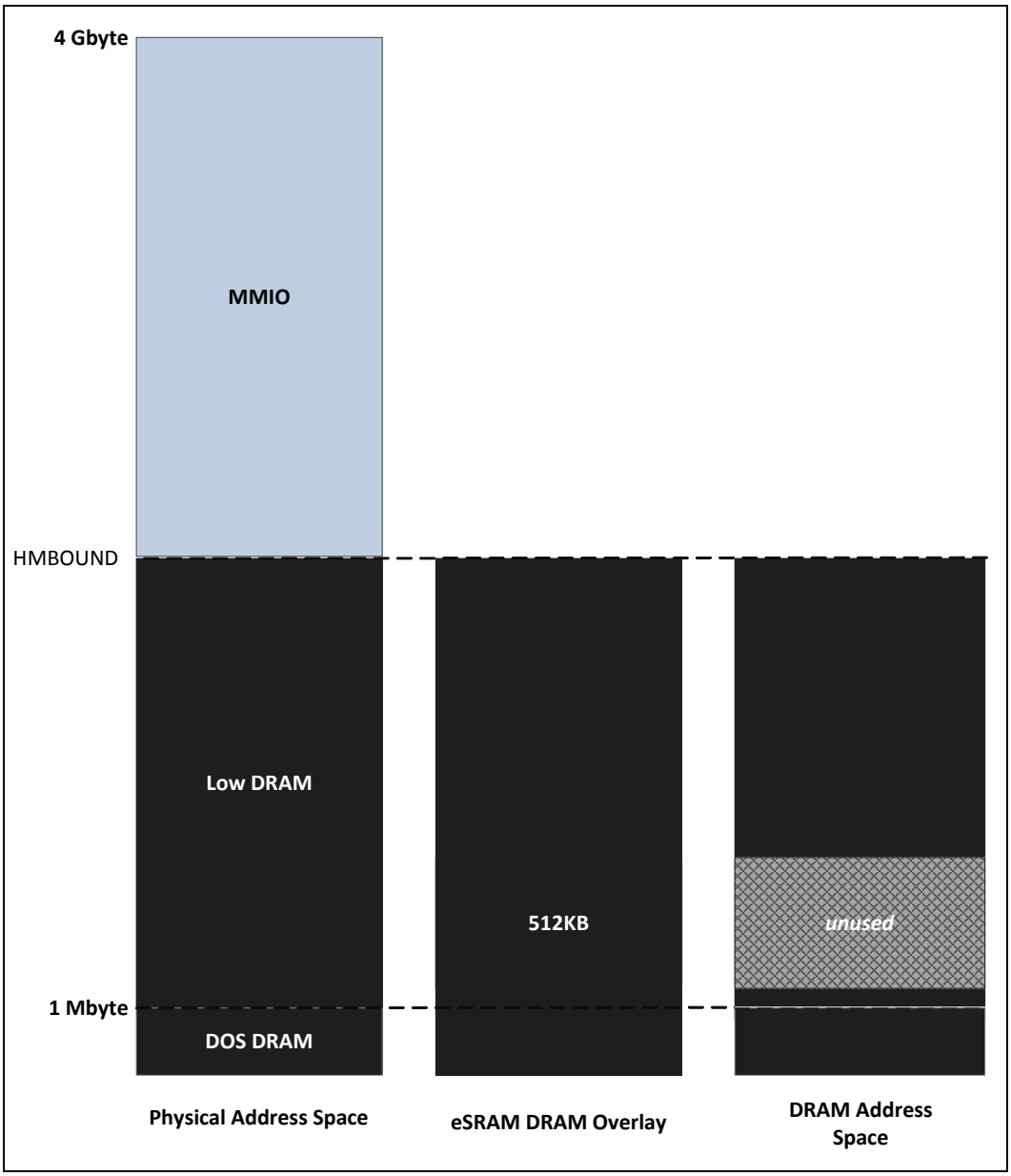
12.1.2.2 512KB Block Page Mode

To map the eSRAM as a single 512KB block page, the register ESRAMPGCTRL_BLOCK is used. If any of the 4KB pages are already enabled, it is not possible to enable the block page.

To map and enable the 512KB block page, the following steps should be followed

- Set ESRAMPGCTRL_BLOCK.BLOCK_PG_SYSTEM_ADDRESS_16MB to the required address value
- Set ESRAMPGCTRL_BLOCK.BLOCK_ENABLE_PG to 1

Figure 23. eSRAM 512KB Page Mapping



12.1.3 Configuration Locking

Once an eSRAM page is enabled, it is implicitly locked and any further configuration change attempts will fail.

eSRAM page configuration may be explicitly locked on a per page basis with the ESRAMPGCTRL*x*. PAGE_CSR_LOCK and ESRAMPGCTRL_BLOCK. BLOCK_PAGE_CSR_LOCK fields. Locked eSRAM pages may still be flushed to DRAM.

All eSRAM configuration registers may be locked with the ESRAMCTRL.eSRAM_GLOBAL_CSR_LOCK field.

12.1.4 ECC Protection

The Host Bridge implements ECC protection for the eSRAM. The eSRAM ECC provides single bit error correction and double bit error detection (SECDED). It is enabled by default, but may be disabled/enabled by setting the register field ESRAMCTRL.SECDED_ENABLE to 0/1.

The ESRAMCERR register provides debug information on the most recent single bit ECC error. Software may configure a threshold number of correctable ECC errors with the ESRAMCTRL.ECC_THRESH field. If the ECC_THRESH_SB_MSG_EN field is set to 1, and the threshold number of correctable ECC errors is reached, the Memory Manager will send an interrupt to the Remote Management Unit with the opcode 0xD8.

The ESRAMUERR register provides debug information on un-correctable ECC errors. If an un-correctable eSRAM ECC error occurs the Memory Manager will send a message to inform the Remote Management Unit. A warm reset of the Host Bridge is required in this case.

The ESRAMSDROME register can be used to decode where in the eSRAM data word the most recent ECC error occurred.

12.1.5 Flush to DRAM

In order to flush a page to DRAM, software must set the ESRAMPGCTRL*x*.FLUSH_PG_ENABLE field to 1 for 4KB pages or the ESRAMPGCTRL_BLOCK.BLOCK_FLUSH_PG_ENABLE to 1 for the 512KB block page. On an S3 entry or warm reset, firmware will flush pages configured in this way to DRAM.

12.2 Isolated Memory Regions (IMR)

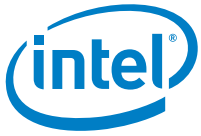
The Host Bridge provides support for Isolated Memory Regions (IMRs). An IMR is an area of system memory that is accessible only to certain system agents. The range and access rights of an IMR are software configurable. There are 3 types of IMR

- General IMR
- Host Memory I/O Boundary (HMBOUND) IMR
- System Management Mode (SMM) IMR

There are 8 general IMRs available. The general IMRs allow any location of system memory- with a 1KB granularity, to have software controlled access rights. The upper and lower boundaries of a general IMR are set via the IMRxL.IMRL and IMRxH.IMRH register fields. Read access rights are controlled via the IMRxRM registers, write access rights are controlled via the IMRx.WM register fields.

General IMRs may be overlapping. In this case, in order be allowed access a particular region in memory, an agent will need to have access rights to all the IMRs which contain that region.

The HMBOUND IMR prevents access by non host agents to any region of memory above HMBOUND. HMBOUND is software configured in the HMBOUND register.



The SMM IMR prevents access by non host agents to any region of memory contained within the SMM region. The SMM region access rights are configured in the HSMMCTL register.

12.2.1 IMR Violation

If an agent is blocked attempting to write to a region covered by an IMR, its write data is dropped, and system memory is not updated.

If an agent is blocked attempting to read to a region covered by an IMR, all 0s are returned for read data. The region of memory is still read (with ECC checking/ correction performed if enabled), but false (all 0) data is returned to the agent instead of the real data from system memory.

If the register field BIMRVCTL.EnableIMRInt is set to 1, and an IMR violation occurs, an interrupt will be sent to the Remote Management Unit with opcode 0xC0. In this case, software may use the BIMRVCTL register to debug the cause of the violation.

12.2.2 IMR Locking

The following settings lock the relevant IMR

- General IMR: set IMRxL.IMR_LOCK to 1
- HMBOUND IMR: set HMBOUND.HMBOUND_LOCK to 1
- SMM IMR: set HSMMCTL.SMM_LOCK to 1

Until HMBOUND is configured and locked, any General IMR region that is programmed will only be applied if the General IMR's register set is not locked. This allows software to configure a General IMR region and test it without locking it's register set. Once a General IMR register set is locked, however, HMBOUND is required to be configured and locked or the security mechanism will deny all accesses to that General IMR region.

Until HMBOUND is configured and locked, the SMM IMR region that is programmed will only be applied if the SMM IMR's register set is not locked. This allows software to configure the SMM IMR region and test it without locking it's register set. Once the SMM IMR register set is locked, however, HMBOUND is required to be configured and locked or the security mechanism will deny all accesses to the SMM IMR region.

12.3 Remote Management Unit DMA

The Remote Management Unit supports DMA transfers between System Memory and Legacy SPI Flash. The DMA engine is used on boot-up to perform the initial firmware fetch from SPI Flash. In addition, this can be used for shadowing firmware to DRAM or eSRAM.

SPI DMA Block registers are used to control DMA transfers. These registers are managed by the Remote Management Unit firmware. Firmware will also shadow the SPI_DMA_CNT, SPI_DMA_DST and SPI_DMA_SRC IO-mapped registers to Remote Management Unit message bus registers 0x60, 0x61 and 0x62. The base address for SPI_DMA_BLK IO registers is given my Remote Management Unit message bus register 0x7A. The SPI_DMA_CNT register should be programmed after SPI_DMA_SRC and SPI_DMA_DST as writing to the SPI_DMA_CNT will trigger the start of the DMA transfer.

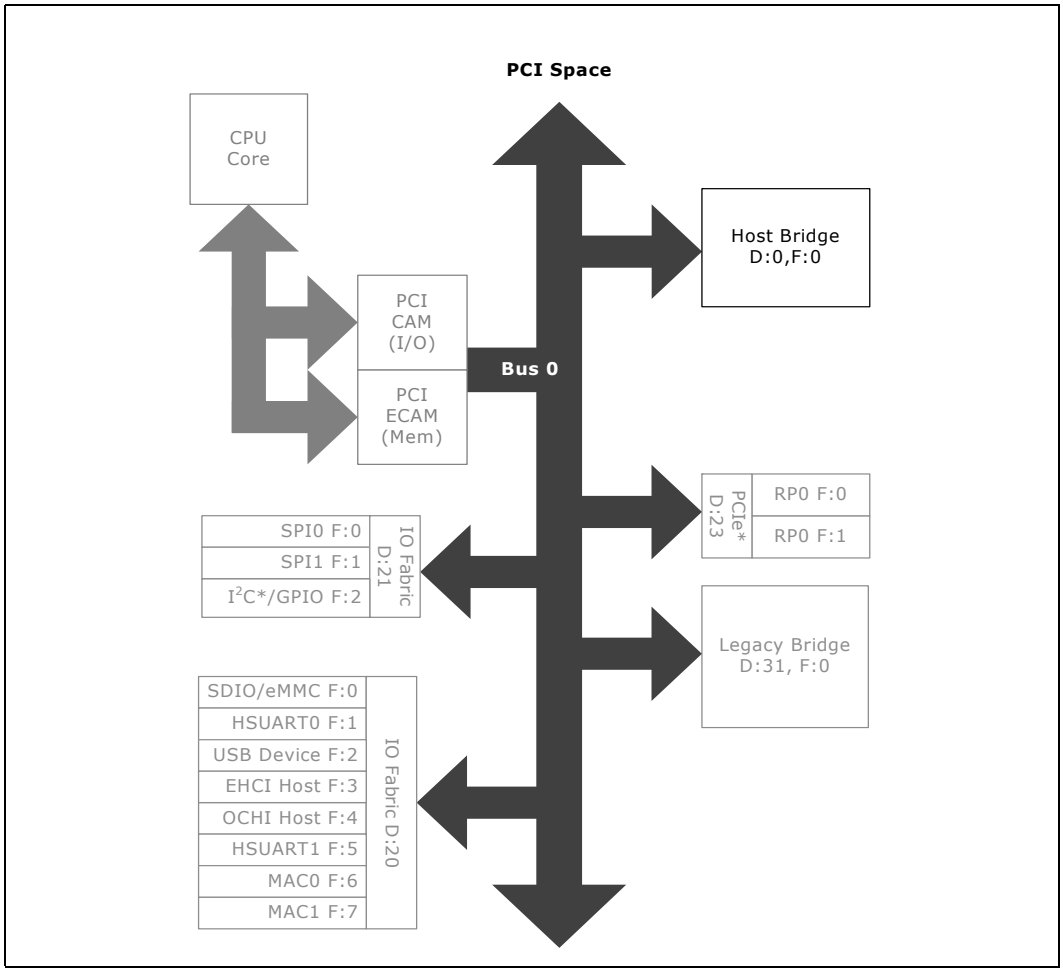
12.3.1 ECC Scrubbing

The Remote Management Unit offers an ECC scrubbing function. The ECC scrubber initiates memory reads that triggers the logic to correct single bit ECC errors and detect double bit ECC errors. If single bit ECC errors are not corrected then they may result in double bit ECC errors over time. Double bit ECC errors are fatal errors that cannot be corrected. To avoid double bit ECC errors, the ECC scrubber may be initialized to periodically scrub all physical memory and correct single bit ECC errors. Note that the ECC scrubber will have an effect on the performance and real-time characteristics of the system.

12.4 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 24. Intel® Quark SoC X1000 Host Bridge Register Map



12.5 PCI Configuration Registers

Table 66. Summary of PCI Configuration Registers—0/0/0

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h" on page 129	09588086h
4h	7h	"PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h" on page 129	00000007h
8h	Bh	"PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h" on page 130	06000000h
Ch	Fh	"PCI Miscellaneous Fields (PCI_MISC)—Offset Ch" on page 130	00000000h
2Ch	2Fh	"PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch" on page 131	00000000h
D0h	D3h	"Message Bus Control Register (MCR) (SB_PACKET_REG)—Offset D0h" on page 132	00000000h
D4h	D7h	"Message Data Register (MDR) (SB_DATA_REG)—Offset D4h" on page 132	00000000h
D8h	DBh	"Message Control Register eXtension (MCRX) (SB_ADDR_EXTN_REG)—Offset D8h" on page 132	00000000h
F8h	FBh	"Manufacturer ID (PCI_MANUFACTURER)—Offset F8h" on page 133	00000FB1h

12.5.1 PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h

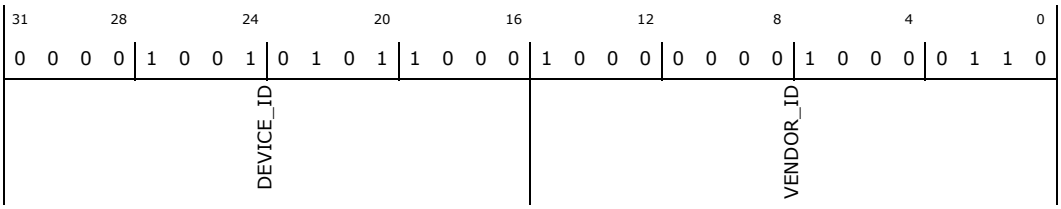
PCI Device ID

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_DEVICE_VENDOR: [B:0, D:0, F:0] + 0h

Default: 09588086h



Bit Range	Default & Access	Description
31: 16	0958h RO	**Device ID (DEVICE_ID):** PCI Device ID
15: 0	8086h RO	**Vendor ID (VENDOR_ID):** PCI Vendor ID for Intel

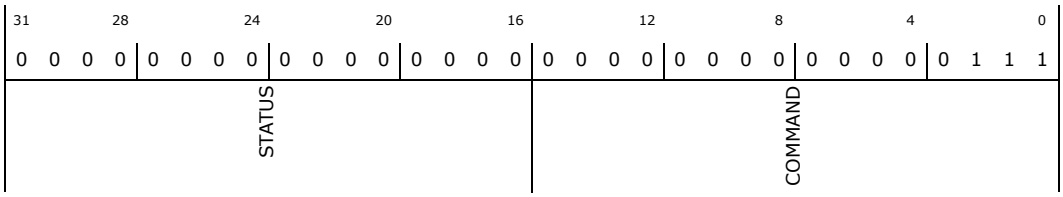
12.5.2 PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_STATUS_COMMAND: [B:0, D:0, F:0] + 4h

Default: 00000007h



Bit Range	Default & Access	Description
31: 16	0000h RO	**Status (STATUS):** Hardwired to 0.
15: 0	0007h RO	**Command (COMMAND):** Hardwired to 0.

12.5.3 PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h

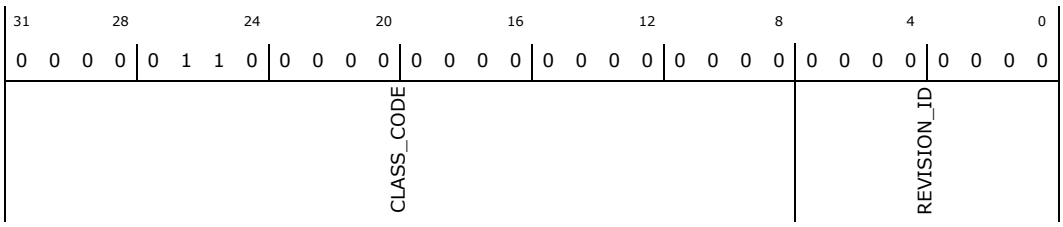
PCI Revision ID

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_CLASS_REVISION: [B:0, D:0, F:0] + 8h

Default: 06000000h



Bit Range	Default & Access	Description
31: 8	060000h RO	**Class Code (CLASS_CODE):** PCI Class Code for Chipset.
7: 0	00h RO	**Revision ID (REVISION_ID):** PCI Revision ID

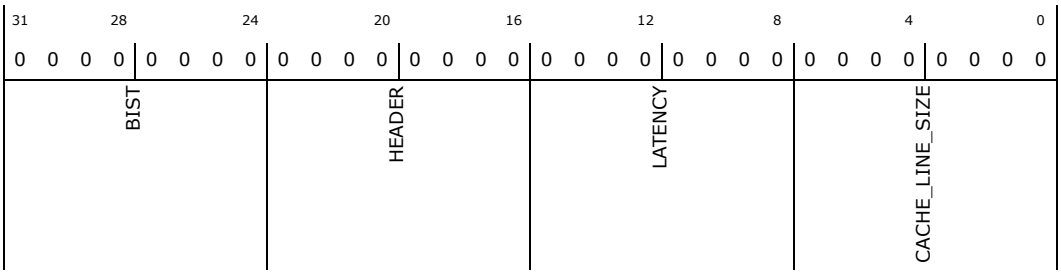
12.5.4 PCI Miscellaneous Fields (PCI_MISC)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_MISC: [B:0, D:0, F:0] + Ch

Default: 00000000h



Bit Range	Default & Access	Description
31: 24	00h RO	**BIST (BIST):** PCI BIST Field
23: 16	00h RO	**Header Type (HEADER):** PCI Header Type Field
15: 8	00h RO	**Latency Timer (LATENCY):** PCI Latency Timer Field
7: 0	00h RO	**Cache Line Size (CACHE_LINE_SIZE):** PCI Cache Line Size Field

12.5.5 PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch

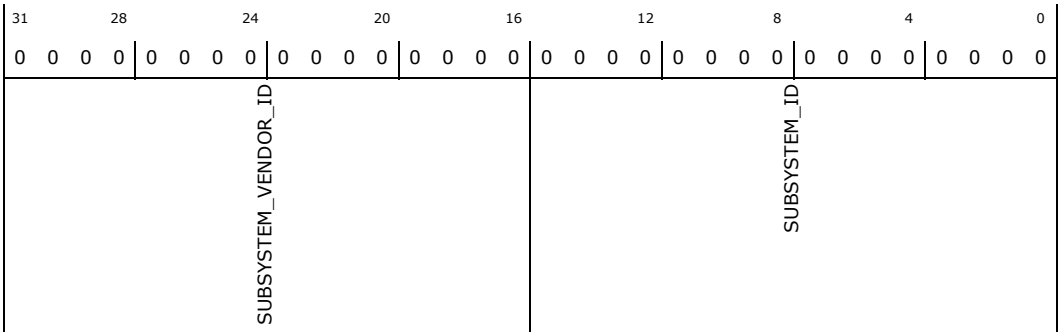
PCI Subsystem ID

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_SUBSYSTEM: [B:0, D:0, F:0] + 2Ch

Default: 00000000h



Bit Range	Default & Access	Description
31: 16	0000h RO	**Subsystem Vendor ID (SUBSYSTEM_VENDOR_ID):** PCI Subsystem Vendor ID
15: 0	0000h RO	**Subsystem ID (SUBSYSTEM_ID):** PCI Subsystem ID

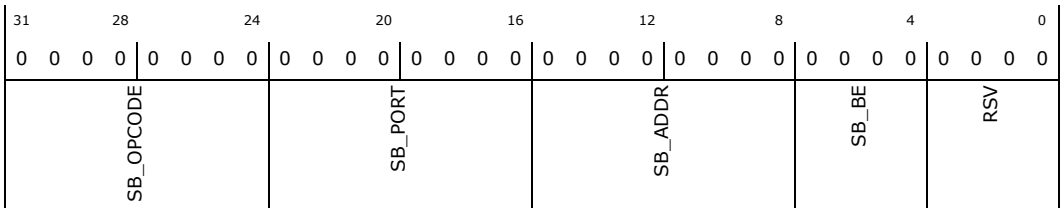
12.5.6 Message Bus Control Register (MCR) (SB_PACKET_REG)—Offset D0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

SB_PACKET_REG: [B:0, D:0, F:0] + D0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 24	00h WO	**OpCode (SB_OPCODE):** The operation to be performed on the target port.
23: 16	00h WO	**Port (SB_PORT):** The device or unit to be targeted by the message bus transaction.
15: 8	00h WO	**Offset/Register (SB_ADDR):** Bits 7:0 of the private register offset to be targeted by the message bus transaction. This field applies only to register read and write operations.
7: 4	0h WO	**Byte Enable (SB_BE):** The byte enables to be used by the triggered transaction. This field applies only to register read and write operations.
3: 0	0h WO	**Reserved (RSV):** Reserved.

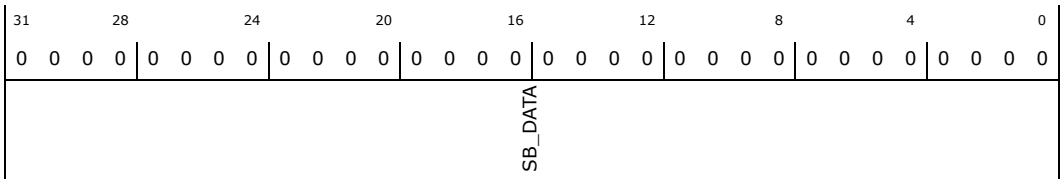
12.5.7 Message Data Register (MDR) (SB_DATA_REG)—Offset D4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

SB_DATA_REG: [B:0, D:0, F:0] + D4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**Data (SB_DATA):** Used as the place to store the data when the operation triggered is a read semantic, or the place to get the data if the triggered operation is a data write semantic.

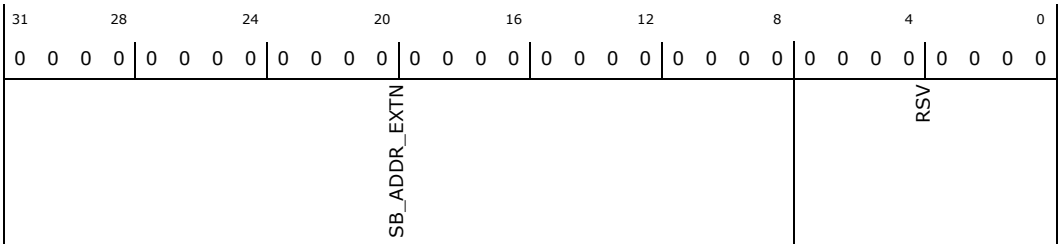
12.5.8 Message Control Register eXtension (MCRX) (SB_ADDR_EXTN_REG)—Offset D8h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

SB_ADDR_EXTN_REG: [B:0, D:0, F:0] + D8h

Default: 00000000h



Bit Range	Default & Access	Description
31: 8	000000h RW/S	**Offset/Register Extension (SB_ADDR_EXTN):** This is used for messages sent to end points that require more than 8 bits for the offset/register. These bits are a direct extension of MCR[15:8].
7: 0	00h RO	**Reserved (RSV):** Reserved.

12.5.9 Manufacturer ID (PCI_MANUFACTURER)—Offset F8h

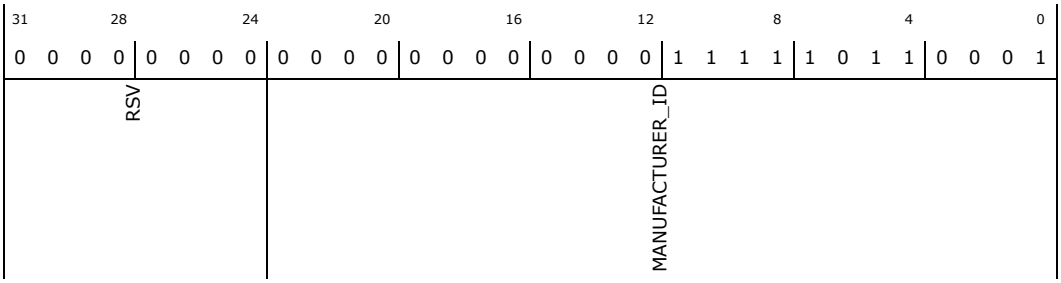
Manufacturer ID

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

PCI_MANUFACTURER: [B:0, D:0, F:0] + F8h

Default: 00000FB1h



Bit Range	Default & Access	Description
31: 24	00h RO	**Reserved (RSV):** Reserved.
23: 0	000FB1h RO	**Manufacturer ID (MANUFACTURER_ID):** Manufacturer ID

12.6 IO Mapped Register

12.6.1 ACPI Processor Block

Table 67. Summary of I/O Registers—PMBA

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"Processor Control (P_CNT)—Offset 0h" on page 134	00000000h
4h	7h	"Level 2 Register (P_LVL2)—Offset 4h" on page 135	00000000h
Ch	Fh	"C6 Control Register (P_C6C)—Offset Ch" on page 135	00000000h

12.6.1.1 Processor Control (P_CNT)—Offset 0h

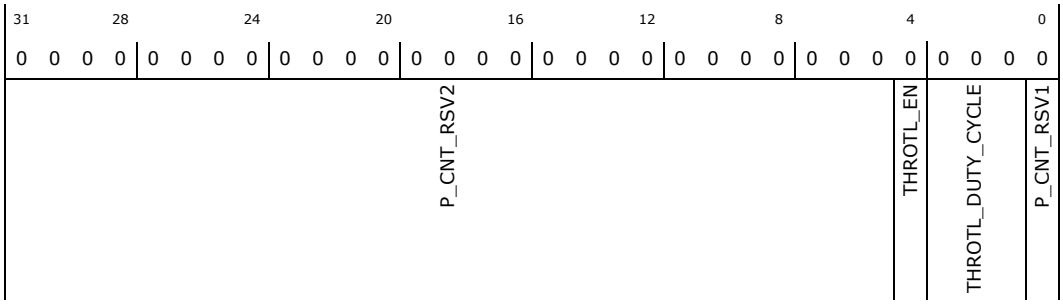
Access Method

Type: I/O Register (Size: 32 bits)

P_CNT: [PMBA] + 0h

PMBA Type: Message Bus Register (Size: 32 bits)
PMBA Reference: [Port: 0x04] + 70h

Default: 00000000h



Bit Range	Default & Access	Description
31:5	0b RO	**Reserved (P_CNT_RSV2):** Reserved.
4	0b RW	**Throttle Enable (THROTL_EN):** When set and the processor is in C0, it enables software-controlled STPCLK# throttling. The duty cycle is selected via TDTY. It remains in effect on each re-entry to C0 as long as enabled.
3:1	000b RW	**Throttle Duty Cycle (THROTL_DUTY_CYCLE):** This field determines the duty cycle of throttling (percentage of time STPCLK# is asserted) when throttling is enabled. The throttle period is approximately 8 s. 000b : 50% (Default) 001b : 87.5% 010b : 75% 011b : 62.5% 100b : 50% 101b : 37.5% 110b : 35% 111b : 12.5%
0	0b RO	**Reserved (P_CNT_RSV1):** Reserved.

12.6.1.2 Level 2 Register (P_LVL2)—Offset 4h

Level 2 Register is an 8-bit register that is used to generate requests to enter C2.

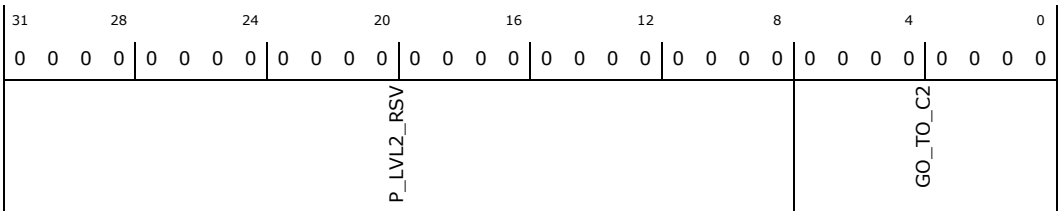
Access Method

Type: I/O Register (Size: 32 bits)

P_LVL2: [PMBA] + 4h

PMBA Type: Message Bus Register (Size: 32 bits)
PMBA Reference: [Port: 0x04] + 70h

Default: 00000000h



Bit Range	Default & Access	Description
31:8	0b RW	**Reserved (P_LVL2_RSV):** Reserved.
7:0	00h RO	**Go to C2 (GO_TO_C2):** Reads to this register return all zeroes, writes have no effect. Reads to this register generate a C2 request.

12.6.1.3 C6 Control Register (P_C6C)—Offset Ch

This is a read only register. It provides information on the last C-state entered and residency in the last entered C-state

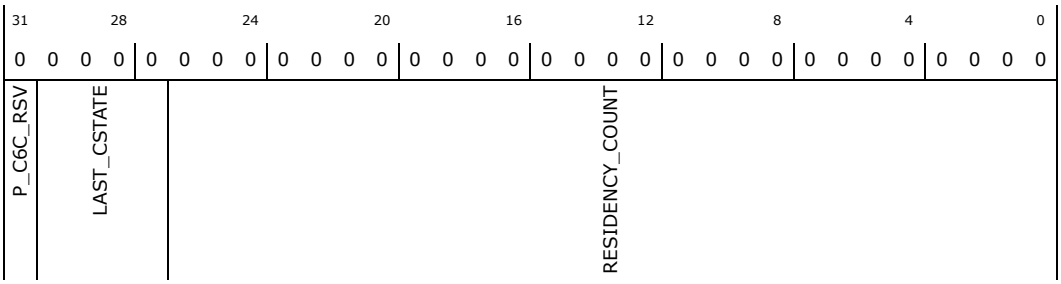
Access Method

Type: I/O Register (Size: 32 bits)

P_C6C: [PMBA] + Ch

PMBA Type: Message Bus Register (Size: 32 bits)
PMBA Reference: [Port: 0x04] + 70h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RO	**Reserved (P_C6C_RSV):** Reserved.

Bit Range	Default & Access	Description
30:27	0b RO	**Last Entered C-State (LAST_CSTATE):** Once CPU transitions from C0 to C2, it updates this register. 0000b : C0 0010b : C2 All other values are reserved.
26:0	0b RO	**Residency Count (RESIDENCY_COUNT):** This register reports the residency in the last entered C-state. The granularity used is microseconds

12.6.2 SPI DMA Block

Table 68. Summary of I/O Registers—SPI_DMA_BAR

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"SPI DMA Count Register (SPI_DMA_CNT_IOSF)—Offset 0h" on page 136	00000000h
4h	7h	"SPI DMA Destination Register (SPI_DMA_DST_IOSF)—Offset 4h" on page 136	00000000h
8h	Bh	"SPI DMA Source Register (SPI_DMA_SRC_IOSF)—Offset 8h" on page 137	00000000h

12.6.2.1 SPI DMA Count Register (SPI_DMA_CNT_IOSF)—Offset 0h

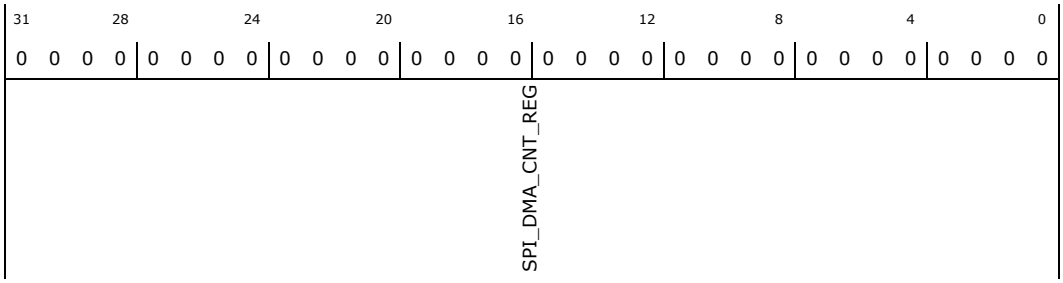
Access Method

Type: I/O Register (Size: 32 bits)

Offset: [SPI_DMA_BAR] + 0h

SPI_DMA_BAR Type: Message Bus Register (Size: 32 bits)
SPI_DMA_BAR Reference: [Port: 0x04] + 7Ah

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**SPI DMA Count (SPI_DMA_CNT_REG):** Count of 512 byte block transfers. Writing this register triggers the start of the transfer of the indicated number of blocks. Reading this register returns the number of blocks that are remaining to be transferred. A value of 0 indicates the transfer is complete.

12.6.2.2 SPI DMA Destination Register (SPI_DMA_DST_IOSF)—Offset 4h

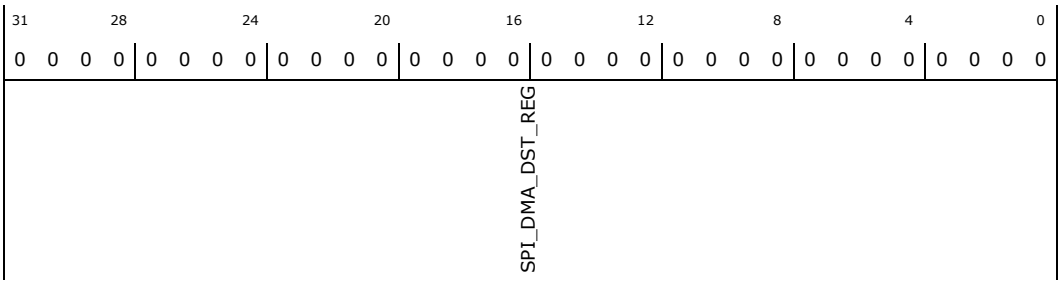
Access Method

Type: I/O Register
(Size: 32 bits)

Offset: [SPI_DMA_BAR] + 4h

SPI_DMA_BAR Type: Message Bus Register (Size: 32 bits)
SPI_DMA_BAR Reference: [Port: 0x04] + 7Ah

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h WO	**SPI DMA Destination (SPI_DMA_DST_REG):** 32-bit Destination Address of data in System Memory (eSRAM/DRAM).

12.6.2.3 SPI DMA Source Register (SPI_DMA_SRC_IOSF)—Offset 8h

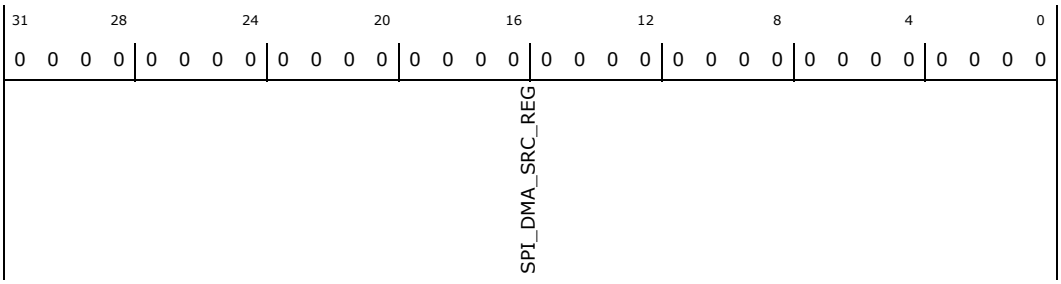
Access Method

Type: I/O Register
(Size: 32 bits)

Offset: [SPI_DMA_BAR] + 8h

SPI_DMA_BAR Type: Message Bus Register (Size: 32 bits)
SPI_DMA_BAR Reference: [Port: 0x04] + 7Ah

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h WO	**SPI DMA Source (SPI_DMA_SRC_REG):** 32-bit Source Address of data in Legacy SPI.

12.7 Message Bus Register

12.7.1 Host Bridge Arbiter (Port 0x00)

Table 69. Summary of Message Bus Registers—0x00

Offset	Register ID—Description	Default Value
0h	"Enhanced Configuration Space (AEC_CTRL)—Offset 0h" on page 138	00000000h
21h	"STATUS—Offset 21h" on page 138	00000000h
50h	"Requester ID Match Control (ASUBCHAN_CTRL)—Offset 50h" on page 139	00000000h
51h	"Requester ID Match Sub-Channel 1 (ASUBCHAN1_MATCH)—Offset 51h" on page 140	00000000h
52h	"Requester ID Match Sub-Channel 2 (ASUBCHAN2_MATCH)—Offset 52h" on page 141	00000000h
53h	"Requester ID Match Sub-Channel 3 (ASUBCHAN3_MATCH)—Offset 53h" on page 141	00000000h

12.7.1.1 Enhanced Configuration Space (AEC_CTRL)—Offset 0h

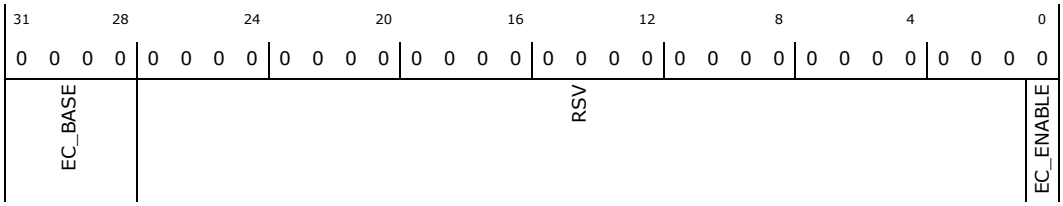
Access Method

Type: Message Bus Register (Size: 32 bits)

AEC_CTRL: [Port: 0x00] + 0h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:28	0b RW	**Enhanced Configuration Space Base Address (EC_BASE):** When EC_BASE matches bits [31:28] of a system memory address that has been forwarded to the Host Bridge Arbiter and Enhanced Configuration operation is enabled, the corresponding operation is treated as an Enhanced Configuration Space access.
27:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RW	**Enable (EC_ENABLE):** Enables Enhanced Configuration operation

12.7.1.2 STATUS—Offset 21h

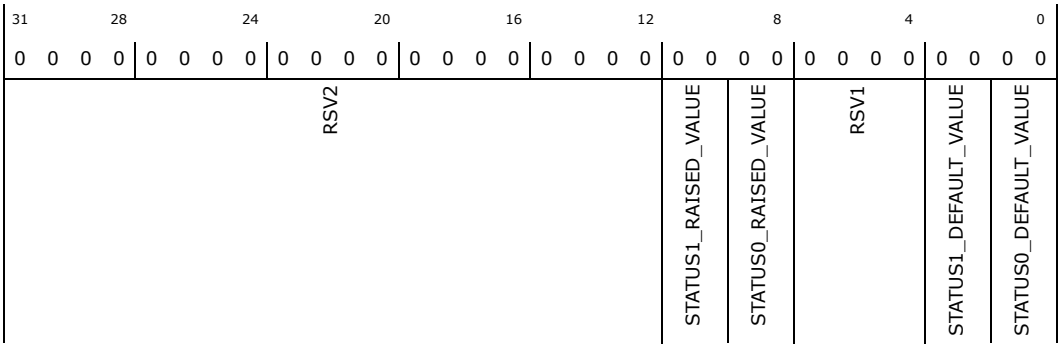
Access Method

Type: Message Bus Register (Size: 32 bits)

ASTATUS: [Port: 0x00] + 21h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	0b RO	**RSV2:** Reserved
11:10	0b RW	**STATUS1_RAISED_VALUE:** Elevated Status value presented to the Memory Manager for Virtual Channel 1 (VC1) accesses from the Host Bridge Arbiter. Encodings as follows: 2'b00 = Casual - Channel is requesting low priority, will be serviced only if convenient. 2'b01 = Reserved. 2'b10 = Normal - Channel is requesting normal priority for servicing. 2'b11 = Urgent - Channel is requesting high priority for servicing.
9:8	0b RW	**STATUS0_RAISED_VALUE:** Elevated Status value presented to the Memory Manager for Virtual Channel 0 (VC0) accesses from the Host Bridge Arbiter. Encodings as follows: 2'b00 = Casual - Channel is requesting low priority, will be serviced only if convenient. 2'b01 = Reserved. 2'b10 = Normal - Channel is requesting normal priority for servicing. 2'b11 = Urgent - Channel is requesting high priority for servicing.
7:4	0b RO	**RSV1:** Reserved
3:2	0b RW	**STATUS1_DEFAULT_VALUE:** Default Status value presented to the Memory Manager for Virtual Channel 1 (VC1) accesses from the Host Bridge Arbiter. Encodings as follows: 2'b00 = Casual - Channel is requesting low priority, will be serviced only if convenient. 2'b01 = Reserved. 2'b10 = Normal - Channel is requesting normal priority for servicing. 2'b11 = Urgent - Channel is requesting high priority for servicing.
1:0	0b RW	**STATUS0_DEFAULT_VALUE:** Default Status value presented to the Memory Manager for Virtual Channel 0 (VC0) accesses from the Host Bridge Arbiter. Encodings as follows: 2'b00 = Casual - Channel is requesting low priority, will be serviced only if convenient. 2'b01 = Reserved. 2'b10 = Normal - Channel is requesting normal priority for servicing. 2'b11 = Urgent - Channel is requesting high priority for servicing.

12.7.1.3 Requester ID Match Control (ASUBCHAN_CTRL)—Offset 50h

Access Method

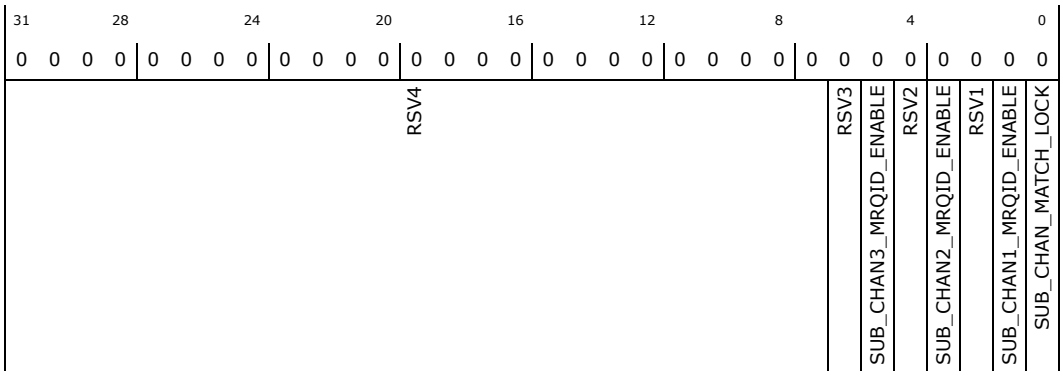
Type: Message Bus Register (Size: 32 bits)

ASUBCHAN_CTRL: [Port: 0x00] + 50h

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:7	0b RO	**Reserved (RSV4):** Reserved.
6	0b RO	**Reserved (RSV3):** Reserved.
5	0b RW/L	**Enable Sub-Channel 3 Matching (SUB_CHAN3_MRQID_ENABLE):** When set, register 53h will be enabled for Sub-Channel 3 generation on a Requester ID match.
4	0b RO	**Reserved (RSV2):** Reserved.
3	0b RW/L	**Enable Sub-Channel 2 Matching (SUB_CHAN2_MRQID_ENABLE):** When set, register 52h will be enabled for Sub-Channel 2 generation on a Requester ID match.
2	0b RO	**Reserved (RSV1):** Reserved.
1	0b RW/L	**Enable Sub-Channel 1 Matching (SUB_CHAN1_MRQID_ENABLE):** When set, register 51h will be enabled for Sub-Channel 1 generation on a Requester ID match.
0	0b RW/O	**Lock Requester ID Matching Registers (SUB_CHAN_MATCH_LOCK):** When set, registers 50h-53h will be set to read-only, in order to preserve the integrity of the IMR Sub-Channel mechanism.

12.7.1.4 Requester ID Match Sub-Channel 1 (ASUBCHAN1_MATCH)—Offset 51h

Access Method

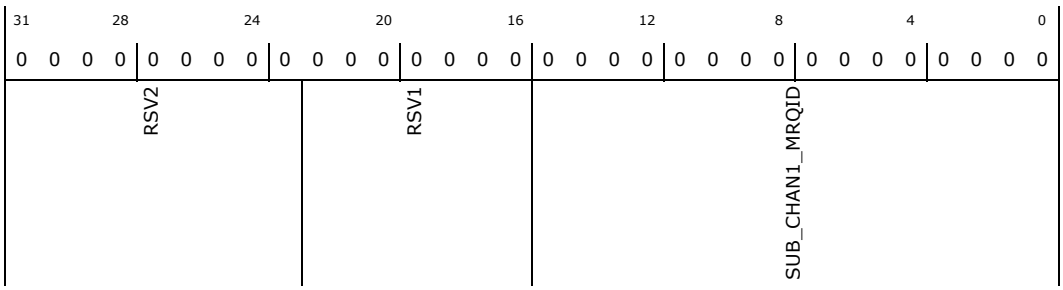
Type: Message Bus Register (Size: 32 bits)

ASUBCHAN1_MATCH: [Port: 0x00] + 51h

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:23	0b RO	**Reserved (RSV2):** Reserved.
22:16	0b RO	**Reserved (RSV1):** Reserved.
15:0	0b RW/L	**Requester ID Match Value (SUB_CHAN1_MRQID):** Value compared to an incoming Requester ID value to determine its Sub-Channel.

12.7.1.5 Requester ID Match Sub-Channel 2 (ASUBCHAN2_MATCH)—Offset 52h

Access Method

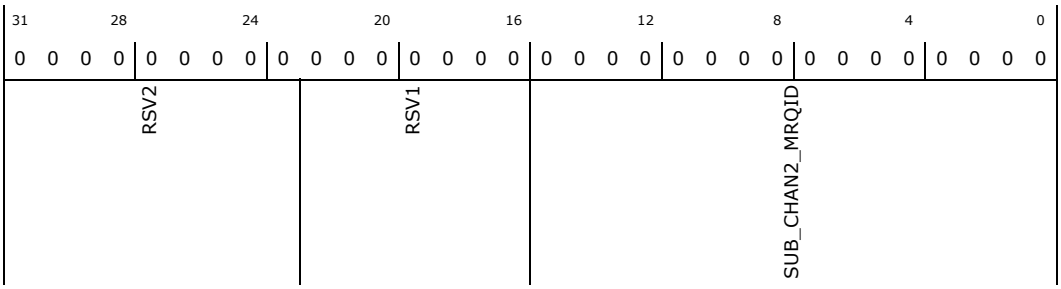
Type: Message Bus Register (Size: 32 bits)

ASUBCHAN2_MATCH: [Port: 0x00] + 52h

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:23	0b RO	**Reserved (RSV2):** Reserved.
22:16	0b RO	**Reserved (RSV1):** Reserved.
15:0	0b RW/L	**Requester ID Match Value (SUB_CHAN2_MRQID):** Value compared to an incoming Requester ID value to determine its Sub-Channel.

12.7.1.6 Requester ID Match Sub-Channel 3 (ASUBCHAN3_MATCH)—Offset 53h

Access Method

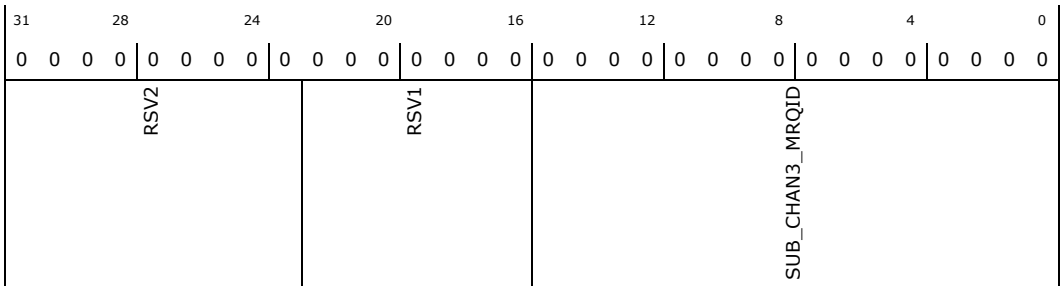
Type: Message Bus Register (Size: 32 bits)

ASUBCHAN3_MATCH: [Port: 0x00] + 53h

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:23	0b RO	**Reserved (RSV2):** Reserved.
22:16	0b RO	**Reserved (RSV1):** Reserved.
15:0	0b RW/L	**Requester ID Match Value (SUB_CHAN3_MRQID):** Value compared to an incoming Requester ID value to determine its Sub-Channel.

12.7.2 Host Bridge (Port 0x03)

Table 70. Summary of Message Bus Registers—0x03

Offset	Register ID—Description	Default Value
3h	"Host Miscellaneous Controls 2 (HMISC2)—Offset 3h" on page 143	00170001h
4h	"Host System Management Mode Controls (HSMMCTL)—Offset 4h" on page 144	00060006h
8h	"Host Memory I/O Boundary (HMBOUND)—Offset 8h" on page 145	40000000h
9h	"Extended Configuration Space (HECREG)—Offset 9h" on page 146	00000000h
Ch	"Host Bridge Write Flush Control (HWFLUSH)—Offset Ch" on page 147	00010000h
40h	"MTRR Capabilities (MTRR_CAP)—Offset 40h" on page 147	00000908h
41h	"MTRR Default Type (MTRR_DEF_TYPE)—Offset 41h" on page 148	00000000h
42h	"MTRR Fixed 64KB Range 0x00000 (MTRR_FIX64K_00000)—Offset 42h" on page 148	00000000h
43h	"MTRR Fixed 64KB Range 0x40000 (MTRR_FIX64K_40000)—Offset 43h" on page 149	00000000h
44h	"MTRR Fixed 16KB Range 0x80000 (MTRR_FIX16K_80000)—Offset 44h" on page 149	00000000h
45h	"MTRR Fixed 16KB Range 0x90000 (MTRR_FIX16K_90000)—Offset 45h" on page 150	00000000h
46h	"MTRR Fixed 16KB Range 0xA0000 (MTRR_FIX16K_A0000)—Offset 46h" on page 151	00000000h
47h	"MTRR Fixed 16KB Range 0xB0000 (MTRR_FIX16K_B0000)—Offset 47h" on page 151	00000000h
48h	"MTRR Fixed 4KB Range 0xC0000 (MTRR_FIX4K_C0000)—Offset 48h" on page 152	00000000h
49h	"MTRR Fixed 4KB Range 0xC4000 (MTRR_FIX4K_C4000)—Offset 49h" on page 152	00000000h
4Ah	"MTRR Fixed 4KB Range 0xC8000 (MTRR_FIX4K_C8000)—Offset 4Ah" on page 153	00000000h
4Bh	"MTRR Fixed 4KB Range 0xCC000 (MTRR_FIX4K_CC000)—Offset 4Bh" on page 153	00000000h
4Ch	"MTRR Fixed 4KB Range 0xD0000 (MTRR_FIX4K_D0000)—Offset 4Ch" on page 154	00000000h
4Dh	"MTRR Fixed 4KB Range 0xD40000 (MTRR_FIX4K_D4000)—Offset 4Dh" on page 155	00000000h
4Eh	"MTRR Fixed 4KB Range 0xD8000 (MTRR_FIX4K_D8000)—Offset 4Eh" on page 155	00000000h

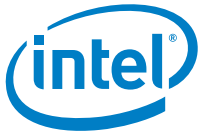


Table 70. Summary of Message Bus Registers—0x03 (Continued)

Offset	Register ID—Description	Default Value
4Fh	"MTRR Fixed 4KB Range 0xDC000 (MTRR_FIX4K_DC000)—Offset 4Fh" on page 156	00000000h
50h	"MTRR Fixed 4KB Range 0xE0000 (MTRR_FIX4K_E0000)—Offset 50h" on page 156	00000000h
51h	"MTRR Fixed 4KB Range 0xE4000 (MTRR_FIX4K_E4000)—Offset 51h" on page 157	00000000h
52h	"MTRR Fixed 4KB Range 0xE8000 (MTRR_FIX4K_E8000)—Offset 52h" on page 157	00000000h
53h	"MTRR Fixed 4KB Range 0xEC000 (MTRR_FIX4K_EC000)—Offset 53h" on page 158	00000000h
54h	"MTRR Fixed 4KB Range 0xF0000 (MTRR_FIX4K_F0000)—Offset 54h" on page 158	00000000h
55h	"MTRR Fixed 4KB Range 0xF4000 (MTRR_FIX4K_F4000)—Offset 55h" on page 159	00000000h
56h	"MTRR Fixed 4KB Range 0xF8000 (MTRR_FIX4K_F8000)—Offset 56h" on page 160	00000000h
57h	"MTRR Fixed 4KB Range 0xFC000 (MTRR_FIX4K_FC000)—Offset 57h" on page 160	00000000h
58h	"System Management Range Physical Base (MTRR_SMRR_PHYSBASE)—Offset 58h" on page 161	00000000h
59h	"System Management Range Physical Mask (MTRR_SMRR_PHYSMASK)—Offset 59h" on page 161	00000000h
5Ah	"MTRR Variable Range Physical Base 0 (MTRR_VAR_PHYSBASE0)—Offset 5Ah" on page 162	00000000h
5Bh	"MTRR Variable Range Physical Mask 0 (MTRR_VAR_PHYSMASK0)—Offset 5Bh" on page 162	00000000h
5Ch	"MTRR Variable Range Physical Base 1 (MTRR_VAR_PHYSBASE1)—Offset 5Ch" on page 163	00000000h
5Dh	"MTRR Variable Range Physical Mask 1 (MTRR_VAR_PHYSMASK1)—Offset 5Dh" on page 164	00000000h
5Eh	"MTRR Variable Range Physical Base 2 (MTRR_VAR_PHYSBASE2)—Offset 5Eh" on page 164	00000000h
5Fh	"MTRR Variable Range Physical Mask 2 (MTRR_VAR_PHYSMASK2)—Offset 5Fh" on page 165	00000000h
60h	"MTRR Variable Range Physical Base 3 (MTRR_VAR_PHYSBASE3)—Offset 60h" on page 165	00000000h
61h	"MTRR Variable Range Physical Mask 3 (MTRR_VAR_PHYSMASK3)—Offset 61h" on page 166	00000000h
62h	"MTRR Variable Range Physical Base 4 (MTRR_VAR_PHYSBASE4)—Offset 62h" on page 166	00000000h
63h	"MTRR Variable Range Physical Mask 4 (MTRR_VAR_PHYSMASK4)—Offset 63h" on page 167	00000000h
64h	"MTRR Variable Range Physical Base 5 (MTRR_VAR_PHYSBASE5)—Offset 64h" on page 168	00000000h
65h	"MTRR Variable Range Physical Mask 5 (MTRR_VAR_PHYSMASK5)—Offset 65h" on page 168	00000000h
66h	"MTRR Variable Range Physical Base 6 (MTRR_VAR_PHYSBASE6)—Offset 66h" on page 169	00000000h
67h	"MTRR Variable Range Physical Mask 6 (MTRR_VAR_PHYSMASK6)—Offset 67h" on page 169	00000000h
68h	"MTRR Variable Range Physical Base 7 (MTRR_VAR_PHYSBASE7)—Offset 68h" on page 170	00000000h
69h	"MTRR Variable Range Physical Mask 7 (MTRR_VAR_PHYSMASK7)—Offset 69h" on page 170	00000000h

12.7.2.1 Host Miscellaneous Controls 2 (HMISC2)—Offset 3h

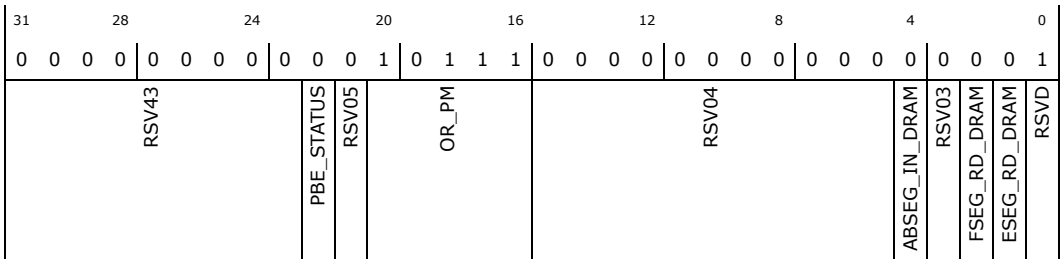
Access Method

Type: Message Bus Register (Size: 32 bits)

HMISC2: [Port: 0x03] + 3h

Op Codes:
10h - Read, 11h - Write

Default: 00170001h



Bit Range	Default & Access	Description
31:23	0h RO	**Reserved (RSV43):** Reserved.
22	0b RO	**PBE Status (PBE_STATUS):** Reflects the value of the Pending Break Event pin from the processor
21	0b RO	**Reserved (RSV05):** Reserved.
20:16	10111b RW	**OR PM Signals from Legacy Bridge (OR_PM):** When set, the Host Bridge will OR the power management signals driven by the Legacy Bridge with the internal values generated by the Remote Management Unit. This field specifies, on a signal-by-signal basis, whether a given bit should be driven via a message from the Remote Management Unit or via a direct pin from the Legacy Bridge. [20] Reserved [19] SMI [18] NMI [17] INIT [16] INTR
15:5	0000h RO	**Reserved (RSV04):** Reserved.
4	0b RW	**A and B Segment in DRAM (ABSEG_IN_DRAM):** When this bit is set, memory reads and writes targeting A-segment or B-segment are routed to DRAM
3	0b RO	**Reserved (RSV03):** Reserved.
2	0b RW	**Read F Segment from DRAM (FSEG_RD_DRAM):** When this bit is set, memory reads targeting F-segment are routed to DRAM
1	0b RW	**Read E Segment from DRAM (ESEG_RD_DRAM):** When this bit is set, memory reads targeting E-segment are routed to DRAM
0	1b RO	**Reserved (RSVD):** Reserved.

12.7.2.2 Host System Management Mode Controls (HSMMCTL)—Offset 4h

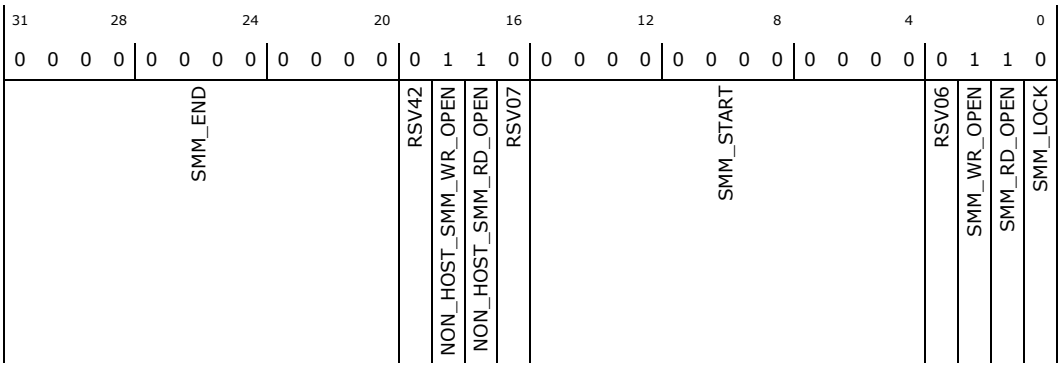
Access Method

Type: Message Bus Register (Size: 32 bits)

HSMMCTL: [Port: 0x03] + 4h

Op Codes:
10h - Read, 11h - Write

Default: 00060006h



Bit Range	Default & Access	Description
31:20	000h RW/L	**SMM Upper Bound (SMM_END):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the protected SMM range.
19	0b RO	**Reserved (RSV42):** Reserved.
18	1b RW/L	**Non-Host SMM Writes Open (NON_HOST_SMM_WR_OPEN):** Allow writes to SMM space from non-host devices. The Memory Manager uses this bit to allow non-host writes to the SMM space defined by the SMM Start and SMM End fields
17	1b RW/L	**Non-Host SMM Reads Open (NON_HOST_SMM_RD_OPEN):** Allow reads to SMM space from non-host devices. The Memory Manager uses this bit to allow non-host reads to the SMM space defined by the SMM Start and SMM End fields
16	0b RO	**Reserved (RSV07):** Reserved.
15:4	000h RW/L	**SMM Lower Bound (SMM_START):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the protected SMM range.
3	0b RO	**Reserved (RSV06):** Reserved.
2	1b RW/L	**SMM Writes Open (SMM_WR_OPEN):** Allow non-SMM writes to SMM space. This bit allows processor writes to the SMM space defined by the SMM Start and SMM End fields even when the processor is not in SMM mode
1	1b RW/L	**SMM Reads Open (SMM_RD_OPEN):** Allow non-SMM reads to SMM space. This bit allows processor reads to the SMM space defined by the SMM Start and SMM End fields even when the processor is not in SMM mode
0	0b RW/O	**SMM Locked (SMM_LOCK):** When set, this bit locks this register and prevents write access until the system is reset

12.7.2.3 Host Memory I/O Boundary (HMBOUND)—Offset 8h

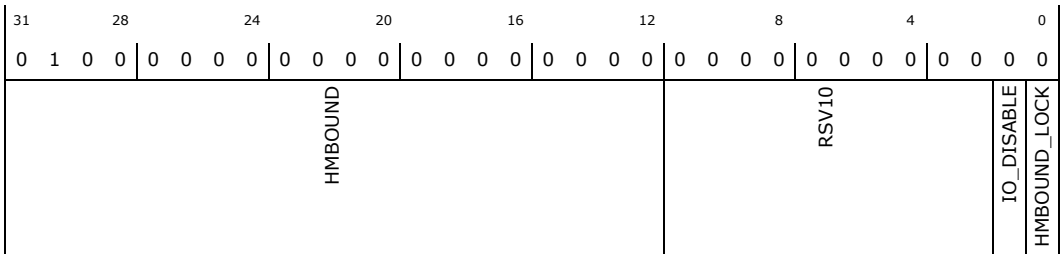
Access Method

Type: Message Bus Register (Size: 32 bits)

HMBOUND: [Port: 0x03] + 8h

Op Codes:
10h - Read, 11h - Write

Default: 40000000h



Bit Range	Default & Access	Description
31:12	40000h RW/L	**Host IO Boundary (HMBOUND):** This register field is compared with the bits [31:12] of incoming memory accesses to determine if the transaction should be routed to Memory space or MMIO space. If address bits[31:12] are greater than or equal to the Host IO Boundary then the transaction is routed to MMIO space. This allows the Host IO Boundary to be set to a 4KB aligned boundary. By default, the Host IO Boundary is set at 1GB.
11:2	000h RO	**Reserved (RSV10):** Reserved.
1	0b RW/L	**Host IO Disable (IO_DISABLE):** When this bit is set, all accesses will be sent to memory regardless of the address with the exception of accesses to the A, B, E and F Segments. Access to the segments is controlled by HMISC2
0	0b RW/O	**HMBOUND Lock (HMBOUND_LOCK):** When this bit is set, the HMBOUND register is locked and can no longer be modified until Host Bridge is reset

12.7.2.4 Extended Configuration Space (HECREG)—Offset 9h

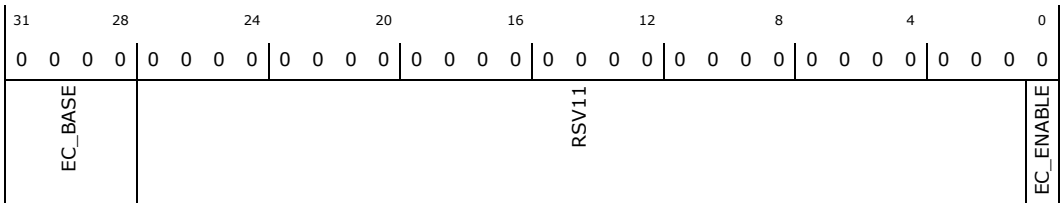
Access Method

Type: Message Bus Register (Size: 32 bits)

HECREG: [Port: 0x03] + 9h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:28	0000b RW	**Extended Configuration Space Base Address (EC_BASE):** This field describes the upper 4-bits of the 32-bit address range used to access the memory-mapped configuration space. This field must not be set to 0xF
27:1	000000h RO	**Reserved (RSV11):** Reserved.
0	0b RW	**Extended Configuration Space Enable (EC_ENABLE):** When set, causes the EC_Base range to be compared to incoming memory accesses. If bits [31:28] of the memory access match the EC_Base value then a posted memory access is treated as a non-posted configuration access.

12.7.2.5 Host Bridge Write Flush Control (HWFLUSH)—Offset Ch

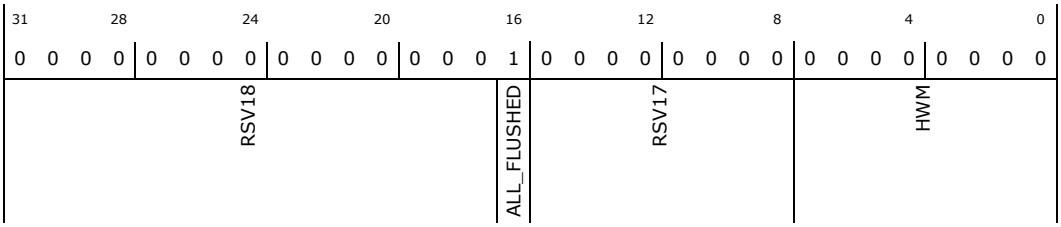
Access Method

Type: Message Bus Register (Size: 32 bits)

HWFLUSH: [Port: 0x03] + Ch

Op Codes:
10h - Read, 11h - Write

Default: 00010000h



Bit Range	Default & Access	Description
31:17	0000h RO	**Reserved (RSV18):** Reserved.
16	1b RO	**All Entries Flushed (ALL_FLUSHED):** Indicates all dirty entries have been flushed from the Host Bridge to the Memory Manager
15:8	00h RO	**Reserved (RSV17):** Reserved.
7:0	00h RW	**High Water Mark (HWM):** High Water Mark for Dirty Entries within the Host Bridge. When this threshold is exceeded, entries are flushed from the Host Bridge to the Memory Manager. Valid values are 0x00, 0x01 and 0x02

12.7.2.6 MTRR Capabilities (MTRR_CAP)—Offset 40h

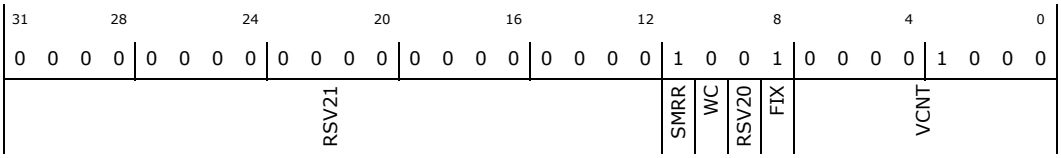
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_CAP: [Port: 0x03] + 40h

Op Codes:
10h - Read, 11h - Write

Default: 00000908h



Bit Range	Default & Access	Description
31:12	00000h RO	**Reserved (RSV21):** Reserved.
11	1b RO	**System Management Register Range Supported (SMRR):** System Management Register Range supported if set
10	0b RO	**Write Combining Memory Type Supported (WC):** Write Combining memory supported if set

Bit Range	Default & Access	Description
9	0b RO	**Reserved (RSV20):** Reserved.
8	1b RO	**Fixed Range Registers Supported (FIX):** Indicates fixed range registers are supported if set
7:0	08h RO	**Variable Range Registers Count (VCNT):** Indicates the number of variable range registers implemented in the Host Bridge

12.7.2.7 MTRR Default Type (MTRR_DEF_TYPE)—Offset 41h

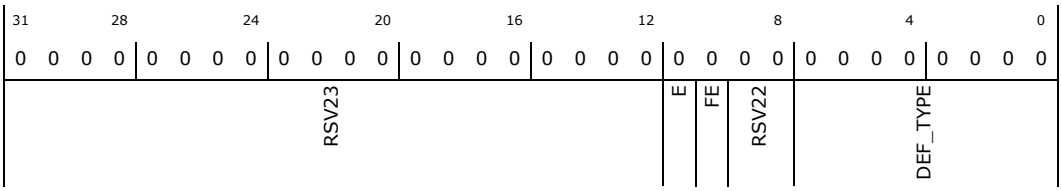
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_DEF_TYPE: [Port: 0x03] + 41h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RO	**Reserved (RSV23):** Reserved.
11	0b RW	**MTRR Enable (E):** MTRRs are enabled when set and are disabled when clear and the UC memory type is applied to all of physical memory. When this flag is set, the FE flag can disable the fixed range MTRRs. When the flag is clear, the FE flag has no affect. When the E flag is set, the type specified in the default memory type field is used for areas of memory not already mapped by either a fixed or variable MTRR.
10	0b RW	**Fixed MTRR Enable (FE):** Fixed range MTRRs are enabled when set and are disabled when clear. When the fixed range MTRRs are enabled, they take priority over the variable range MTRRs when overlaps in ranges occur. If the fixed range MTRRs are disabled, the variable range MTRRs can still be used and can map the range ordinarily covered by the fixed range MTRRs.
9:8	00b RO	**Reserved (RSV22):** Reserved.
7:0	00h RW	**Default Memory Type (DEF_TYPE):** Indicates the default memory type used for memory address ranges that do not have a memory type specified for them by an MTRR. Default value is UC (Uncached)

12.7.2.8 MTRR Fixed 64KB Range 0x00000 (MTRR_FIX64K_00000)—Offset 42h

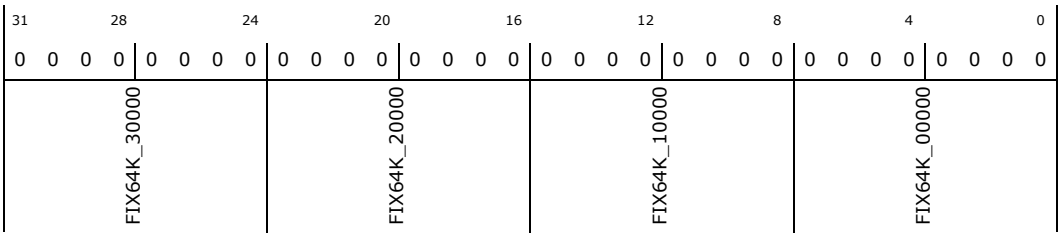
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX64K_00000: [Port: 0x03] + 42h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 64KB Range 0x30000 (FIX64K_30000):** Maps the 64KB range from 0x30000 to 0x3FFFF
23:16	00h RW	**Fixed 64KB Range 0x20000 (FIX64K_20000):** Maps the 64KB range from 0x20000 to 0x2FFFF
15:8	00h RW	**Fixed 64KB Range 0x10000 (FIX64K_10000):** Maps the 64KB range from 0x10000 to 0x1FFFF
7:0	00h RW	**Fixed 64KB Range 0x00000 (FIX64K_00000):** Maps the 64KB range from 0x00000 to 0x0FFFF

12.7.2.9 MTRR Fixed 64KB Range 0x40000 (MTRR_FIX64K_40000)—Offset 43h

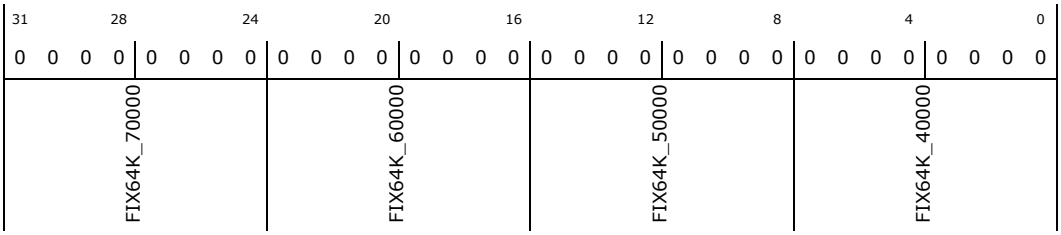
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX64K_40000: [Port: 0x03] + 43h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 64KB Range 0x70000 (FIX64K_70000):** Maps the 64KB range from 0x70000 to 0x7FFFF
23:16	00h RW	**Fixed 64KB Range 0x60000 (FIX64K_60000):** Maps the 64KB range from 0x60000 to 0x6FFFF
15:8	00h RW	**Fixed 64KB Range 0x50000 (FIX64K_50000):** Maps the 64KB range from 0x50000 to 0x5FFFF
7:0	00h RW	**Fixed 64KB Range 0x40000 (FIX64K_40000):** Maps the 64KB range from 0x40000 to 0x4FFFF

12.7.2.10 MTRR Fixed 16KB Range 0x80000 (MTRR_FIX16K_80000)—Offset 44h

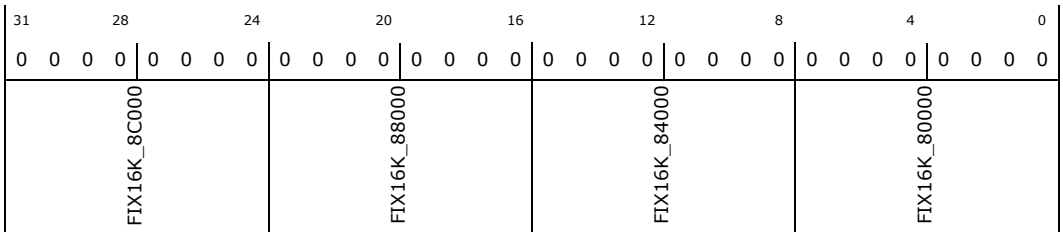
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX16K_80000: [Port: 0x03] + 44h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 16KB Range 0x8C000 (FIX16K_8C000):** Maps the 16KB range from 0x8C000 to 0x8FFFF
23:16	00h RW	**Fixed 16KB Range 0x88000 (FIX16K_88000):** Maps the 16KB range from 0x88000 to 0x8BFFF
15:8	00h RW	**Fixed 16KB Range 0x84000 (FIX16K_84000):** Maps the 16KB range from 0x84000 to 0x87FFF
7:0	00h RW	**Fixed 16KB Range 0x80000 (FIX16K_80000):** Maps the 16KB range from 0x80000 to 0x83FFF

12.7.2.11 MTRR Fixed 16KB Range 0x90000 (MTRR_FIX16K_90000)—Offset 45h

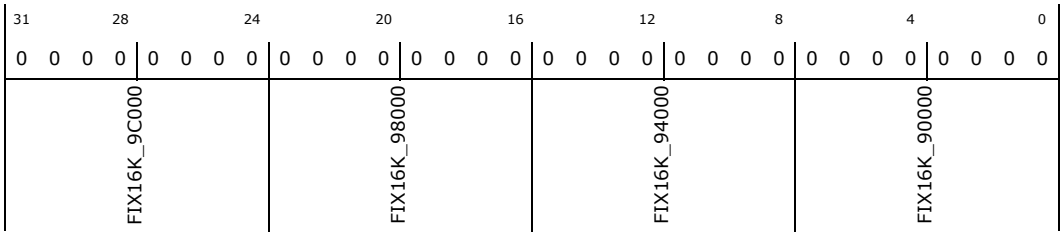
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX16K_90000: [Port: 0x03] + 45h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 16KB Range 0x9C000 (FIX16K_9C000):** Maps the 16KB range from 0x9C000 to 0x9FFFF
23:16	00h RW	**Fixed 16KB Range 0x98000 (FIX16K_98000):** Maps the 16KB range from 0x98000 to 0x9BFFF
15:8	00h RW	**Fixed 16KB Range 0x94000 (FIX16K_94000):** Maps the 16KB range from 0x94000 to 0x97FFF

Bit Range	Default & Access	Description
7:0	00h RW	**Fixed 16KB Range 0x90000 (FIX16K_90000):** Maps the 16KB range from 0x90000 to 0x93FFF

12.7.2.12 MTRR Fixed 16KB Range 0xA0000 (MTRR_FIX16K_A0000)—Offset 46h

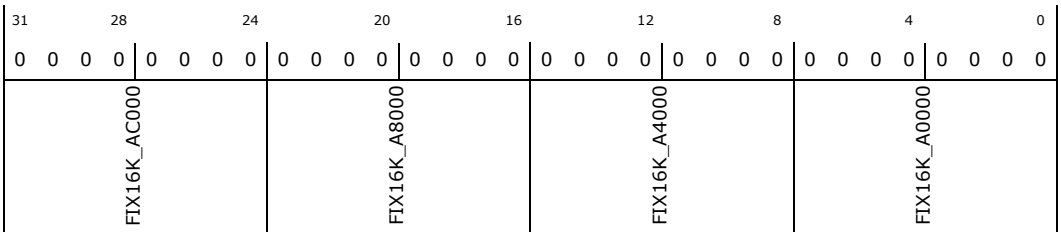
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX16K_A0000: [Port: 0x03] + 46h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 16KB Range 0xAC000 (FIX16K_AC000):** Maps the 16KB range from 0xAC000 to 0xAFFFF
23:16	00h RW	**Fixed 16KB Range 0xA8000 (FIX16K_A8000):** Maps the 16KB range from 0xA8000 to 0xABFFF
15:8	00h RW	**Fixed 16KB Range 0xA4000 (FIX16K_A4000):** Maps the 16KB range from 0xA4000 to 0xA7FFF
7:0	00h RW	**Fixed 16KB Range 0xA0000 (FIX16K_A0000):** Maps the 16KB range from 0xA0000 to 0xA3FFF

12.7.2.13 MTRR Fixed 16KB Range 0xB0000 (MTRR_FIX16K_B0000)—Offset 47h

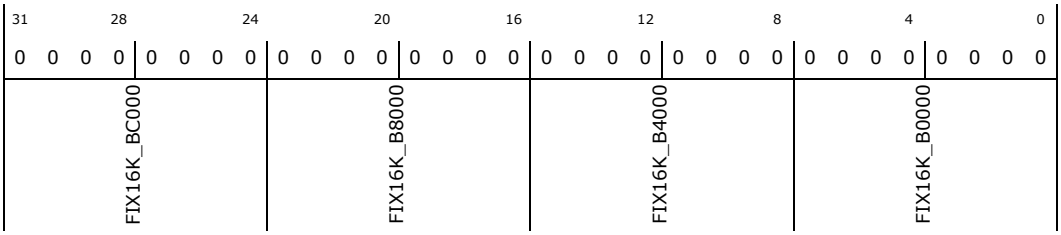
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX16K_B0000: [Port: 0x03] + 47h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 16KB Range 0xBC000 (FIX16K_BC000):** Maps the 16KB range from 0xBC000 to 0xBFFFF
23:16	00h RW	**Fixed 16KB Range 0xB8000 (FIX16K_B8000):** Maps the 16KB range from 0xB8000 to 0xBBFFF
15:8	00h RW	**Fixed 16KB Range 0xB4000 (FIX16K_B4000):** Maps the 16KB range from 0xB4000 to 0xB7FFF
7:0	00h RW	**Fixed 16KB Range 0xB0000 (FIX16K_B0000):** Maps the 16KB range from 0xB0000 to 0xB3FFF

12.7.2.14 MTRR Fixed 4KB Range 0xC0000 (MTRR_FIX4K_C0000)—Offset 48h

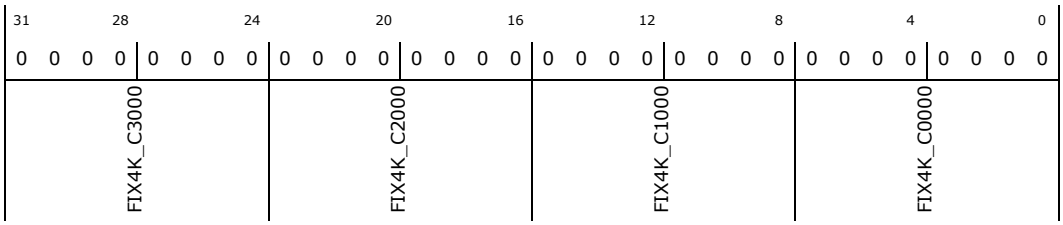
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_C0000: [Port: 0x03] + 48h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xC3000 (FIX4K_C3000):** Maps the 4KB range from 0xC3000 to 0xC3FFF
23:16	00h RW	**Fixed 4KB Range 0xC2000 (FIX4K_C2000):** Maps the 4KB range from 0xC2000 to 0xC2FFF
15:8	00h RW	**Fixed 4KB Range 0xC1000 (FIX4K_C1000):** Maps the 4KB range from 0xC1000 to 0xC1FFF
7:0	00h RW	**Fixed 4KB Range 0xC0000 (FIX4K_C0000):** Maps the 4KB range from 0xC0000 to 0xC0FFF

12.7.2.15 MTRR Fixed 4KB Range 0xC4000 (MTRR_FIX4K_C4000)—Offset 49h

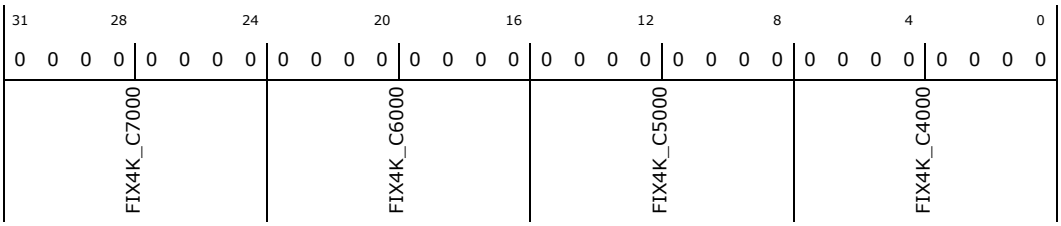
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_C4000: [Port: 0x03] + 49h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xC7000 (FIX4K_C7000):** Maps the 4KB range from 0xC7000 to 0xC7FFF
23:16	00h RW	**Fixed 4KB Range 0xC6000 (FIX4K_C6000):** Maps the 4KB range from 0xC6000 to 0xC6FFF
15:8	00h RW	**Fixed 4KB Range 0xC5000 (FIX4K_C5000):** Maps the 4KB range from 0xC5000 to 0xC5FFF
7:0	00h RW	**Fixed 4KB Range 0xC4000 (FIX4K_C4000):** Maps the 4KB range from 0xC4000 to 0xC4FFF

12.7.2.16 MTRR Fixed 4KB Range 0xC8000 (MTRR_FIX4K_C8000)—Offset 4Ah

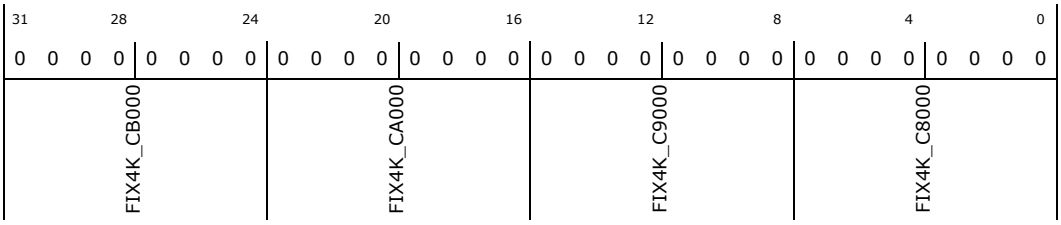
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_C8000: [Port: 0x03] + 4Ah

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xCB000 (FIX4K_CB000):** Maps the 4KB range from 0xCB000 to 0xCBFFF
23:16	00h RW	**Fixed 4KB Range 0xCA000 (FIX4K_CA000):** Maps the 4KB range from 0xCA000 to 0xCAFFF
15:8	00h RW	**Fixed 4KB Range 0xC9000 (FIX4K_C9000):** Maps the 4KB range from 0xC9000 to 0xC9FFF
7:0	00h RW	**Fixed 4KB Range 0xC8000 (FIX4K_C8000):** Maps the 4KB range from 0xC8000 to 0xC8FFF

12.7.2.17 MTRR Fixed 4KB Range 0xCC000 (MTRR_FIX4K_CC000)—Offset 4Bh

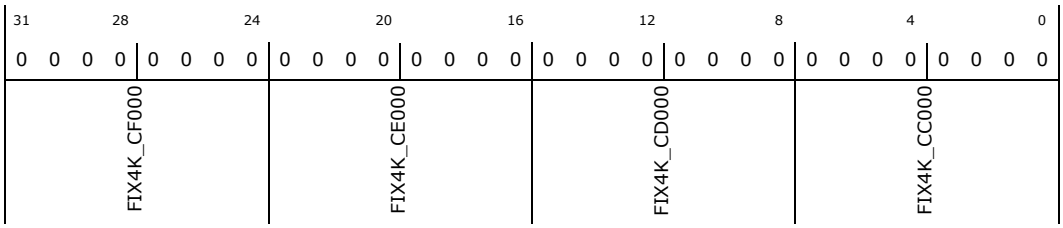
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_CC000: [Port: 0x03] + 4Bh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xCF000 (FIX4K_CF000):** Maps the 4KB range from 0xCF000 to 0xCFFFF
23:16	00h RW	**Fixed 4KB Range 0xCE000 (FIX4K_CE000):** Maps the 4KB range from 0xCE000 to 0xCEFFF
15:8	00h RW	**Fixed 4KB Range 0xCD000 (FIX4K_CD000):** Maps the 4KB range from 0xCD000 to 0xCDFFF
7:0	00h RW	**Fixed 4KB Range 0xCC000 (FIX4K_CC000):** Maps the 4KB range from 0xCC000 to 0xCCFFF

12.7.2.18 MTRR Fixed 4KB Range 0xD0000 (MTRR_FIX4K_D0000)—Offset 4Ch

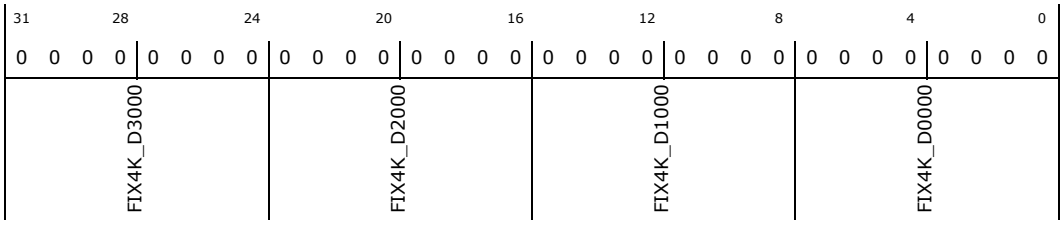
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_D0000: [Port: 0x03] + 4Ch

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xD3000 (FIX4K_D3000):** Maps the 4KB range from 0xD3000 to 0xD3FFF
23:16	00h RW	**Fixed 4KB Range 0xD2000 (FIX4K_D2000):** Maps the 4KB range from 0xD2000 to 0xD2FFF
15:8	00h RW	**Fixed 4KB Range 0xD1000 (FIX4K_D1000):** Maps the 4KB range from 0xD1000 to 0xD1FFF
7:0	00h RW	**Fixed 4KB Range 0xD0000 (FIX4K_D0000):** Maps the 4KB range from 0xD0000 to 0xD0FFF

12.7.2.19 MTRR Fixed 4KB Range 0xD40000 (MTRR_FIX4K_D4000)—Offset 4Dh

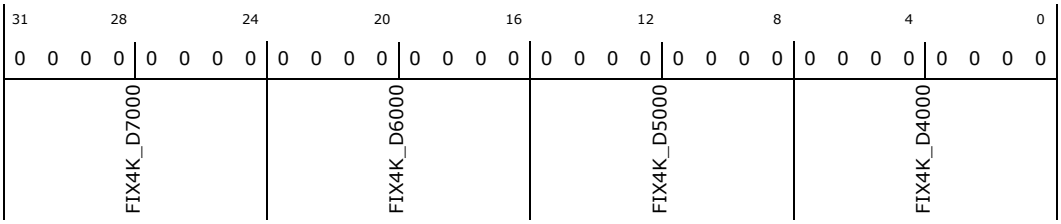
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_D4000: [Port: 0x03] + 4Dh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xD7000 (FIX4K_D7000):** Maps the 4KB range from 0xD7000 to 0xD7FFF
23:16	00h RW	**Fixed 4KB Range 0xD6000 (FIX4K_D6000):** Maps the 4KB range from 0xD6000 to 0xD6FFF
15:8	00h RW	**Fixed 4KB Range 0xD5000 (FIX4K_D5000):** Maps the 4KB range from 0xD5000 to 0xD5FFF
7:0	00h RW	**Fixed 4KB Range 0xD4000 (FIX4K_D4000):** Maps the 4KB range from 0xD4000 to 0xD4FFF

12.7.2.20 MTRR Fixed 4KB Range 0xD8000 (MTRR_FIX4K_D8000)—Offset 4Eh

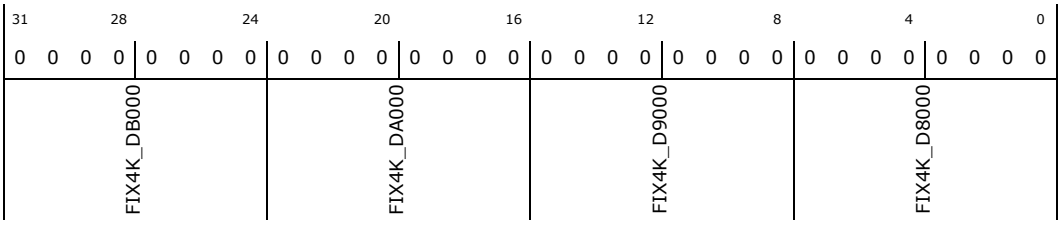
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_D8000: [Port: 0x03] + 4Eh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xDB000 (FIX4K_DB000):** Maps the 4KB range from 0xDB000 to 0xDBFFF
23:16	00h RW	**Fixed 4KB Range 0xDA000 (FIX4K_DA000):** Maps the 4KB range from 0xDA000 to 0xDAFFF

Bit Range	Default & Access	Description
15:8	00h RW	**Fixed 4KB Range 0xD9000 (FIX4K_D9000):** Maps the 4KB range from 0xD9000 to 0xD9FFF
7:0	00h RW	**Fixed 4KB Range 0xD8000 (FIX4K_D8000):** Maps the 4KB range from 0xD8000 to 0xD8FFF

12.7.2.21 MTRR Fixed 4KB Range 0xDC000 (MTRR_FIX4K_DC000)—Offset 4Fh

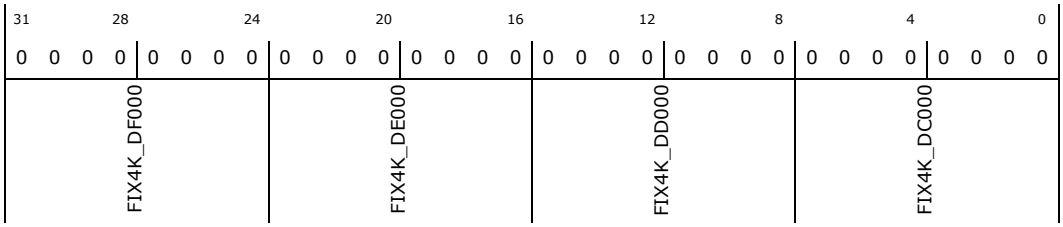
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_DC000: [Port: 0x03] + 4Fh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xDF000 (FIX4K_DF000):** Maps the 4KB range from 0xDF000 to 0xDFFFF
23:16	00h RW	**Fixed 4KB Range 0xDE000 (FIX4K_DE000):** Maps the 4KB range from 0xDE000 to 0xDEFFF
15:8	00h RW	**Fixed 4KB Range 0xDD000 (FIX4K_DD000):** Maps the 4KB range from 0xDD000 to 0xDDFFF
7:0	00h RW	**Fixed 4KB Range 0xDC000 (FIX4K_DC000):** Maps the 4KB range from 0xDC000 to 0xDCFFF

12.7.2.22 MTRR Fixed 4KB Range 0xE0000 (MTRR_FIX4K_E0000)—Offset 50h

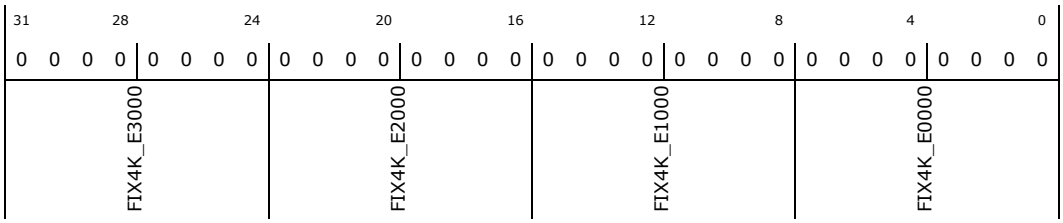
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_E0000: [Port: 0x03] + 50h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xE3000 (FIX4K_E3000):** Maps the 4KB range from 0xE3000 to 0xE3FFF
23:16	00h RW	**Fixed 4KB Range 0xE2000 (FIX4K_E2000):** Maps the 4KB range from 0xE2000 to 0xE2FFF
15:8	00h RW	**Fixed 4KB Range 0xE1000 (FIX4K_E1000):** Maps the 4KB range from 0xE1000 to 0xE1FFF
7:0	00h RW	**Fixed 4KB Range 0xE0000 (FIX4K_E0000):** Maps the 4KB range from 0xE0000 to 0xE0FFF

12.7.2.23 MTRR Fixed 4KB Range 0xE4000 (MTRR_FIX4K_E4000)—Offset 51h

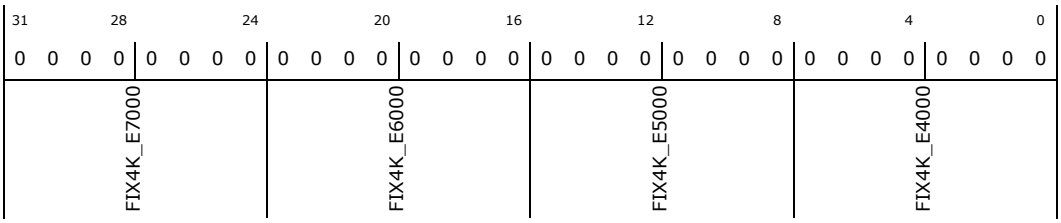
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_E4000: [Port: 0x03] + 51h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xE7000 (FIX4K_E7000):** Maps the 4KB range from 0xE7000 to 0xE7FFF
23:16	00h RW	**Fixed 4KB Range 0xE6000 (FIX4K_E6000):** Maps the 4KB range from 0xE6000 to 0xE6FFF
15:8	00h RW	**Fixed 4KB Range 0xE5000 (FIX4K_E5000):** Maps the 4KB range from 0xE5000 to 0xE5FFF
7:0	00h RW	**Fixed 4KB Range 0xE4000 (FIX4K_E4000):** Maps the 4KB range from 0xE4000 to 0xE4FFF

12.7.2.24 MTRR Fixed 4KB Range 0xE8000 (MTRR_FIX4K_E8000)—Offset 52h

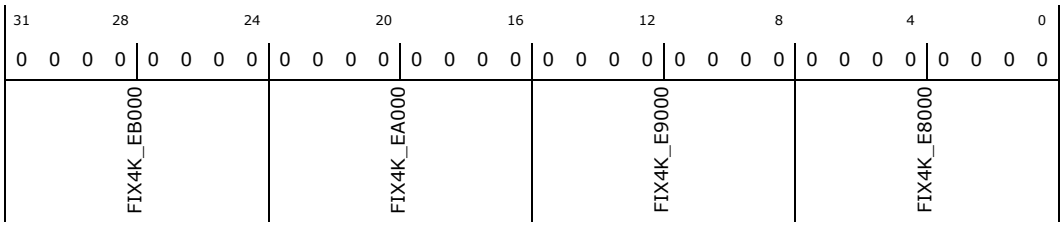
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_E8000: [Port: 0x03] + 52h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xEB000 (FIX4K_EB000):** Maps the 4KB range from 0xEB000 to 0xEBFFF
23:16	00h RW	**Fixed 4KB Range 0xEA000 (FIX4K_EA000):** Maps the 4KB range from 0xEA000 to 0xEAFFF
15:8	00h RW	**Fixed 4KB Range 0xE9000 (FIX4K_E9000):** Maps the 4KB range from 0xE9000 to 0xE9FFF
7:0	00h RW	**Fixed 4KB Range 0xE8000 (FIX4K_E8000):** Maps the 4KB range from 0xE8000 to 0xE8FFF

12.7.2.25 MTRR Fixed 4KB Range 0xEC000 (MTRR_FIX4K_EC000)—Offset 53h

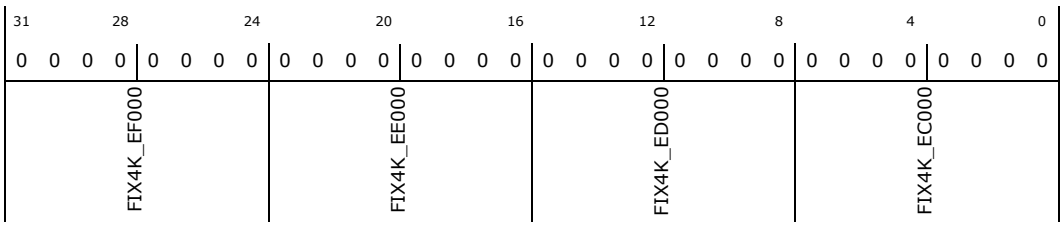
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_EC000: [Port: 0x03] + 53h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xEF000 (FIX4K_EF000):** Maps the 4KB range from 0xEF000 to 0xEFFFF
23:16	00h RW	**Fixed 4KB Range 0xEE000 (FIX4K_EE000):** Maps the 4KB range from 0xEE000 to 0xEEFFF
15:8	00h RW	**Fixed 4KB Range 0xED000 (FIX4K_ED000):** Maps the 4KB range from 0xED000 to 0xEDFFF
7:0	00h RW	**Fixed 4KB Range 0xEC000 (FIX4K_EC000):** Maps the 4KB range from 0xEC000 to 0xECFFF

12.7.2.26 MTRR Fixed 4KB Range 0xF0000 (MTRR_FIX4K_F0000)—Offset 54h

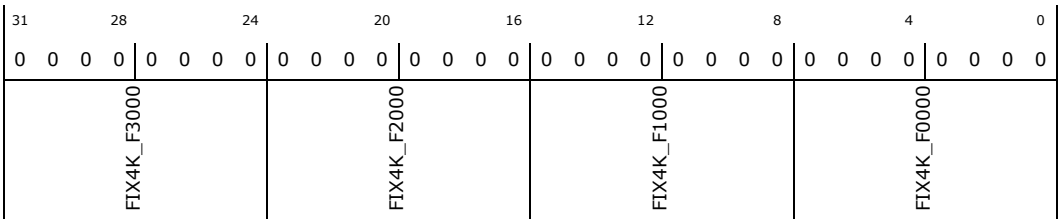
Access Method

Type: Message Bus Register
(Size: 32 bits)

MTRR_FIX4K_F0000: [Port: 0x03] + 54h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xF3000 (FIX4K_F3000):** Maps the 4KB range from 0xF3000 to 0xF3FFF
23:16	00h RW	**Fixed 4KB Range 0xF2000 (FIX4K_F2000):** Maps the 4KB range from 0xF2000 to 0xF2FFF
15:8	00h RW	**Fixed 4KB Range 0xF1000 (FIX4K_F1000):** Maps the 4KB range from 0xF1000 to 0xF1FFF
7:0	00h RW	**Fixed 4KB Range 0xF0000 (FIX4K_F0000):** Maps the 4KB range from 0xF0000 to 0xF0FFF

12.7.2.27 MTRR Fixed 4KB Range 0xF4000 (MTRR_FIX4K_F4000)—Offset 55h

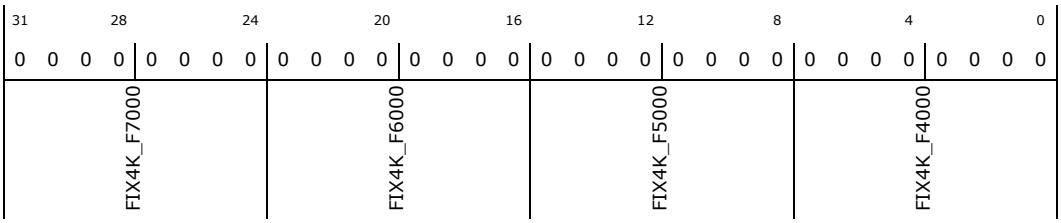
Access Method

Type: Message Bus Register
(Size: 32 bits)

MTRR_FIX4K_F4000: [Port: 0x03] + 55h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xF7000 (FIX4K_F7000):** Maps the 4KB range from 0xF7000 to 0xF7FFF
23:16	00h RW	**Fixed 4KB Range 0xF6000 (FIX4K_F6000):** Maps the 4KB range from 0xF6000 to 0xF6FFF
15:8	00h RW	**Fixed 4KB Range 0xF5000 (FIX4K_F5000):** Maps the 4KB range from 0xF5000 to 0xF5FFF
7:0	00h RW	**Fixed 4KB Range 0xF4000 (FIX4K_F4000):** Maps the 4KB range from 0xF4000 to 0xF4FFF

12.7.2.28 MTRR Fixed 4KB Range 0xF8000 (MTRR_FIX4K_F8000)—Offset 56h

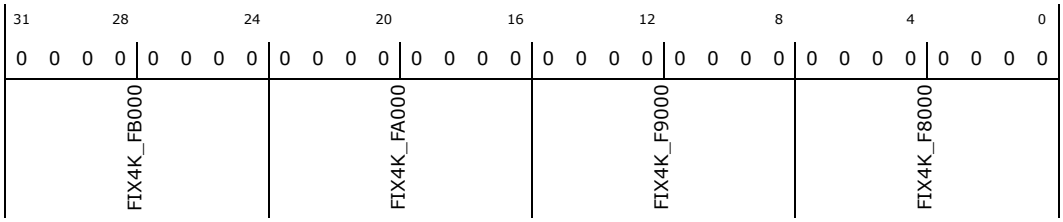
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_F8000: [Port: 0x03] + 56h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xFB000 (FIX4K_FB000):** Maps the 4KB range from 0xFB000 to 0xFBFFF
23:16	00h RW	**Fixed 4KB Range 0xFA000 (FIX4K_FA000):** Maps the 4KB range from 0xFA000 to 0xFAFFF
15:8	00h RW	**Fixed 4KB Range 0xF9000 (FIX4K_F9000):** Maps the 4KB range from 0xF9000 to 0xF9FFF
7:0	00h RW	**Fixed 4KB Range 0xF8000 (FIX4K_F8000):** Maps the 4KB range from 0xF8000 to 0xF8FFF

12.7.2.29 MTRR Fixed 4KB Range 0xFC000 (MTRR_FIX4K_FC000)—Offset 57h

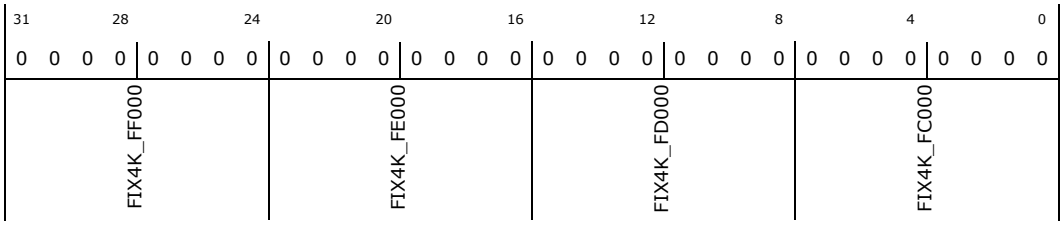
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_FIX4K_FC000: [Port: 0x03] + 57h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RW	**Fixed 4KB Range 0xFF000 (FIX4K_FF000):** Maps the 4KB range from 0xFF000 to 0xFFFFF
23:16	00h RW	**Fixed 4KB Range 0xFE000 (FIX4K_FE000):** Maps the 4KB range from 0xFE000 to 0xFEFFF

Bit Range	Default & Access	Description
15:8	00h RW	**Fixed 4KB Range 0xFD000 (FIX4K_FD000):** Maps the 4KB range from 0xFD000 to 0xFDFFF
7:0	00h RW	**Fixed 4KB Range 0xFC000 (FIX4K_FC000):** Maps the 4KB range from 0xFC000 to 0xFCFFF

12.7.2.30 System Management Range Physical Base (MTRR_SMRR_PHYSBASE)—Offset 58h

This register may only be written while in SMM. Attempts to write this register outside of SMM will be ignored.

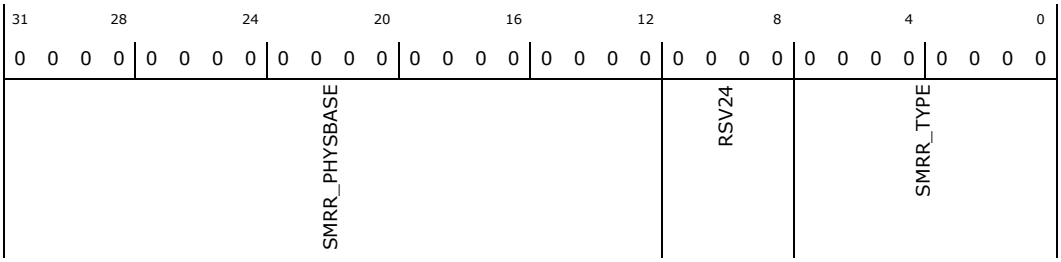
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_SMRR_PHYSBASE: [Port: 0x03] + 58h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**SMRR Physical Base (SMRR_PHYSBASE):** Specifies the base address for the System Management Range. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV24):** Reserved.
7:0	00h RW	**SMRR Type (SMRR_TYPE):** Specifies the memory type for System Management Range

12.7.2.31 System Management Range Physical Mask (MTRR_SMRR_PHYSMASK)—Offset 59h

This register may only be written while in SMM. Attempts to write this register outside of SMM will be ignored.

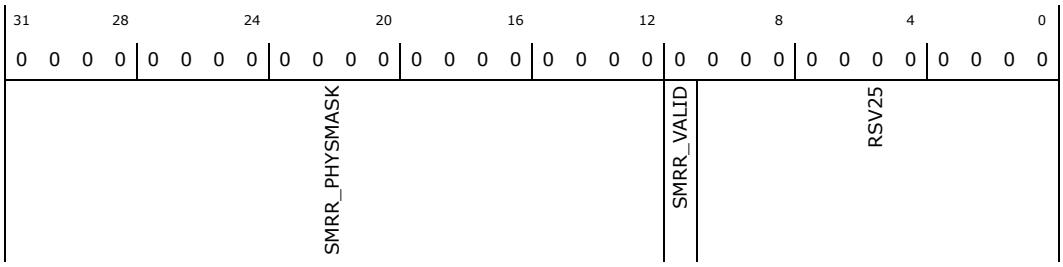
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_SMRR_PHYSMASK: [Port: 0x03] + 59h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**SMRR Physical Mask (SMRR_PHYSMASK):** Specifies a mask value for the System Management Range. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**SMRR Valid (SMRR_VALID):** Enables the register pair for the System Management Range when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV25):** Reserved.

12.7.2.32 MTRR Variable Range Physical Base 0 (MTRR_VAR_PHYSBASE0)—Offset 5Ah

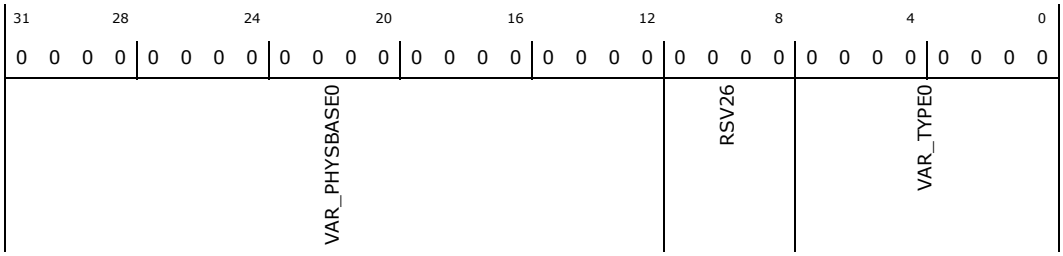
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE0: [Port: 0x03] + 5Ah

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE0):** Specifies the base address for Variable Range 0. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV26):** Reserved.
7:0	00h RW	**Type (VAR_TYPE0):** Specifies the memory type for Variable Range 0

12.7.2.33 MTRR Variable Range Physical Mask 0 (MTRR_VAR_PHYSMASK0)—Offset 5Bh

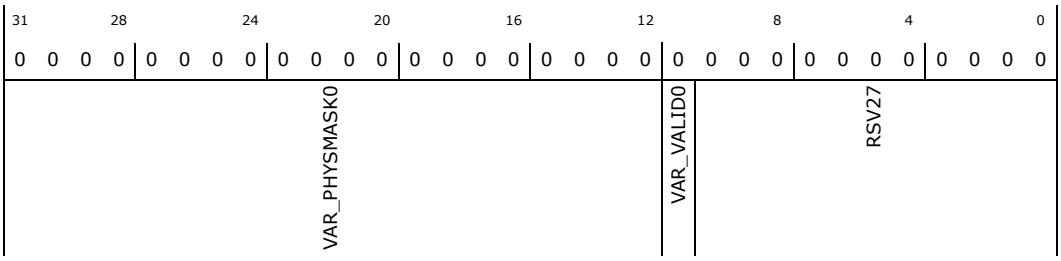
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK0: [Port: 0x03] + 5Bh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK0):** Specifies a mask value for Variable Range 0. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID0):** Enables the register pair for Variable Range 0 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV27):** Reserved.

12.7.2.34 MTRR Variable Range Physical Base 1 (MTRR_VAR_PHYSBASE1)—Offset 5Ch

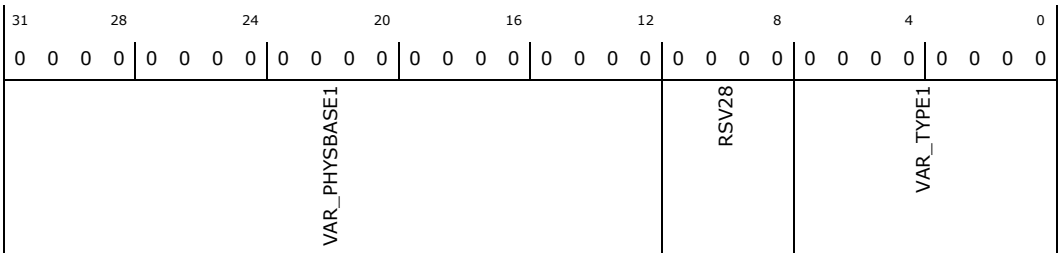
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE1: [Port: 0x03] + 5Ch

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE1):** Specifies the base address for Variable Range 1. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV28):** Reserved.
7:0	00h RW	**Type (VAR_TYPE1):** Specifies the memory type for Variable Range 1

12.7.2.35 MTRR Variable Range Physical Mask 1 (MTRR_VAR_PHYSMASK1)—Offset 5Dh

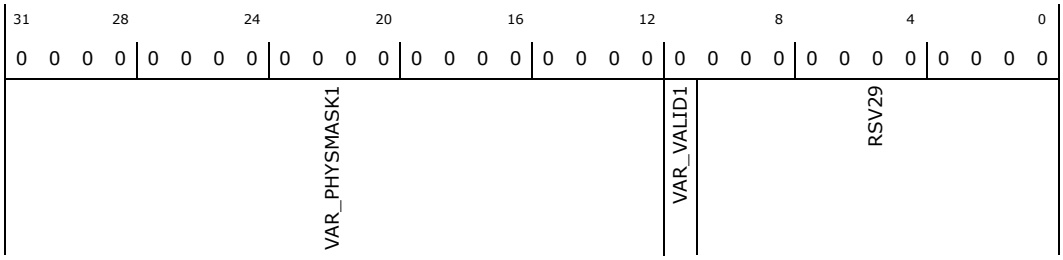
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK1: [Port: 0x03] + 5Dh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK1):** Specifies a mask value for Variable Range 1. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID1):** Enables the register pair for Variable Range 1 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV29):** Reserved.

12.7.2.36 MTRR Variable Range Physical Base 2 (MTRR_VAR_PHYSBASE2)—Offset 5Eh

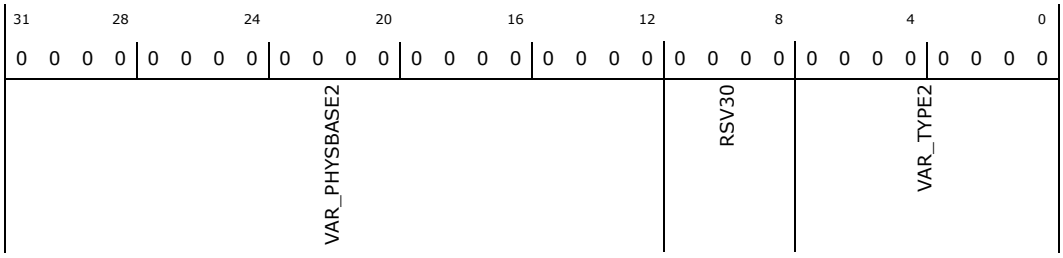
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE2: [Port: 0x03] + 5Eh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE2):** Specifies the base address for Variable Range 2. This 20 bit value is extended by 12 bits at the low end to form the base address

Bit Range	Default & Access	Description
11:8	0000b RO	**Reserved (RSV30):** Reserved.
7:0	00h RW	**Type (VAR_TYPE2):** Specifies the memory type for Variable Range 2

12.7.2.37 MTRR Variable Range Physical Mask 2 (MTRR_VAR_PHYSMASK2)—Offset 5Fh

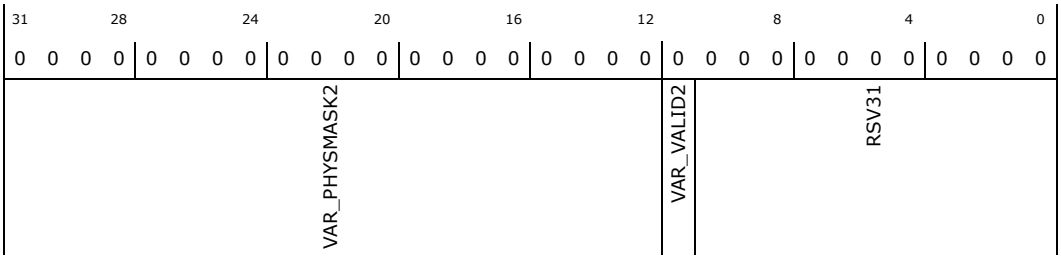
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK2: [Port: 0x03] + 5Fh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK2):** Specifies a mask value for Variable Range 2. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID2):** Enables the register pair for Variable Range 2 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV31):** Reserved.

12.7.2.38 MTRR Variable Range Physical Base 3 (MTRR_VAR_PHYSBASE3)—Offset 60h

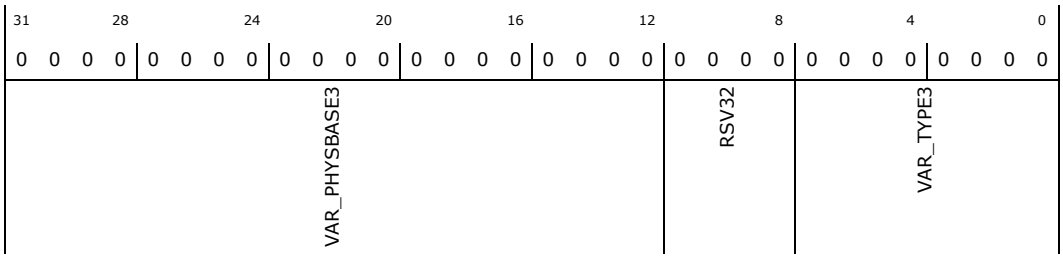
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE3: [Port: 0x03] + 60h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE3):** Specifies the base address for Variable Range 3. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV32):** Reserved.
7:0	00h RW	**Type (VAR_TYPE3):** Specifies the memory type for Variable Range 3

12.7.2.39 MTRR Variable Range Physical Mask 3 (MTRR_VAR_PHYSMASK3)—Offset 61h

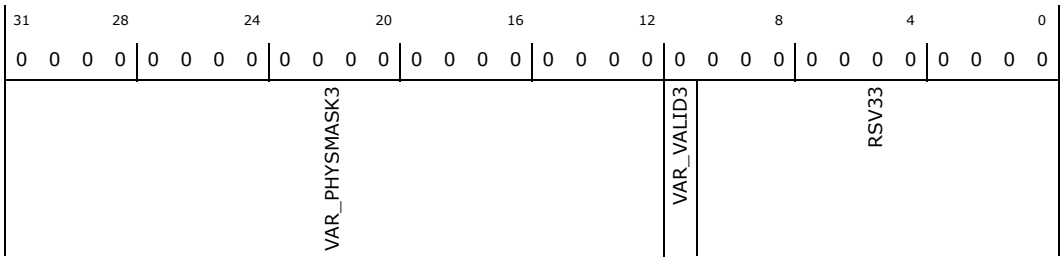
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK3: [Port: 0x03] + 61h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK3):** Specifies a mask value for Variable Range 3. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID3):** Enables the register pair for Variable Range 3 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV33):** Reserved.

12.7.2.40 MTRR Variable Range Physical Base 4 (MTRR_VAR_PHYSBASE4)—Offset 62h

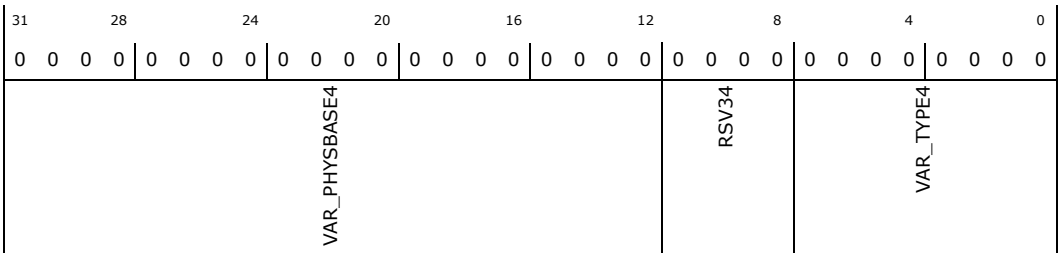
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE4: [Port: 0x03] + 62h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE4):** Specifies the base address for Variable Range 4. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV34):** Reserved.
7:0	00h RW	**Type (VAR_TYPE4):** Specifies the memory type for Variable Range 4

12.7.2.41 MTRR Variable Range Physical Mask 4 (MTRR_VAR_PHYSMASK4)—Offset 63h

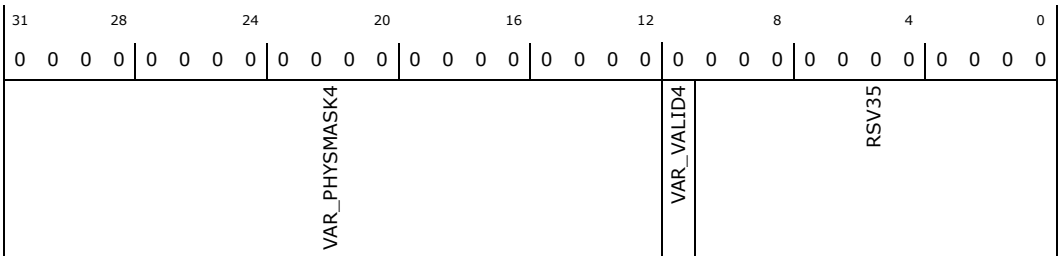
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK4: [Port: 0x03] + 63h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK4):** Specifies a mask value for Variable Range 4. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID4):** Enables the register pair for Variable Range 4 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV35):** Reserved.

12.7.2.42 MTRR Variable Range Physical Base 5 (MTRR_VAR_PHYSBASE5)—Offset 64h

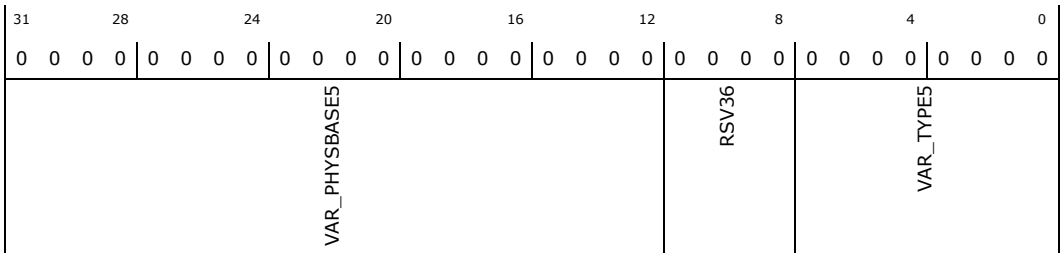
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE5: [Port: 0x03] + 64h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE5):** Specifies the base address for Variable Range 5. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV36):** Reserved.
7:0	00h RW	**Type (VAR_TYPE5):** Specifies the memory type for Variable Range 5

12.7.2.43 MTRR Variable Range Physical Mask 5 (MTRR_VAR_PHYSMASK5)—Offset 65h

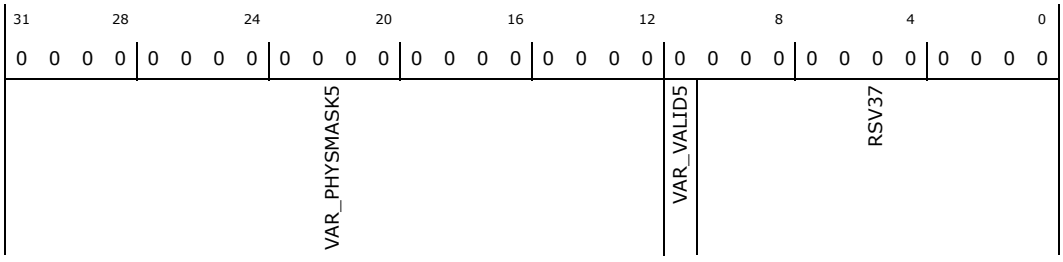
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK5: [Port: 0x03] + 65h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK5):** Specifies a mask value for Variable Range 5. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.

Bit Range	Default & Access	Description
11	0b RW	**Valid (VAR_VALID5):** Enables the register pair for Variable Range 5 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV37):** Reserved.

12.7.2.44 MTRR Variable Range Physical Base 6 (MTRR_VAR_PHYSBASE6)—Offset 66h

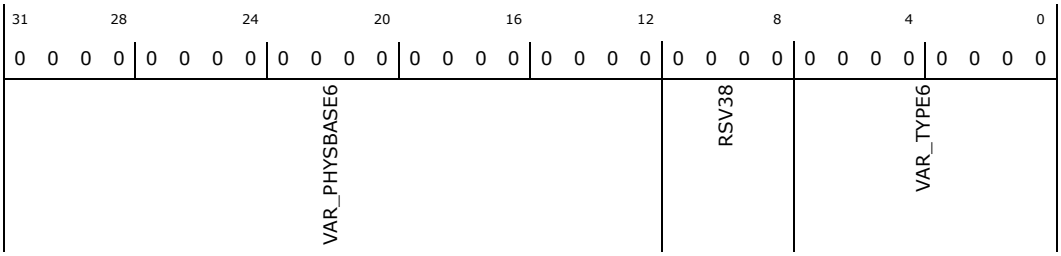
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE6: [Port: 0x03] + 66h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE6):** Specifies the base address for Variable Range 6. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV38):** Reserved.
7:0	00h RW	**Type (VAR_TYPE6):** Specifies the memory type for Variable Range 6

12.7.2.45 MTRR Variable Range Physical Mask 6 (MTRR_VAR_PHYSMASK6)—Offset 67h

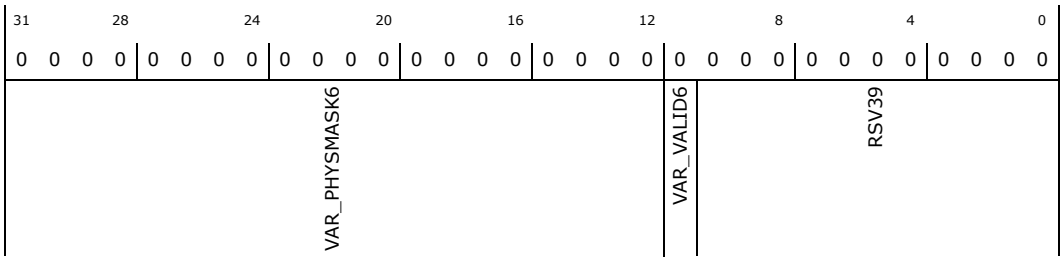
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK6: [Port: 0x03] + 67h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK6):** Specifies a mask value for Variable Range 6. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID6):** Enables the register pair for Variable Range 6 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV39):** Reserved.

12.7.2.46 MTRR Variable Range Physical Base 7 (MTRR_VAR_PHYSBASE7)—Offset 68h

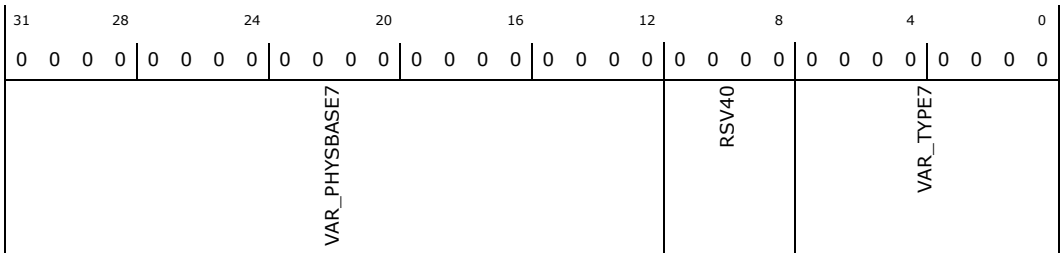
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSBASE7: [Port: 0x03] + 68h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Base (VAR_PHYSBASE7):** Specifies the base address for Variable Range 7. This 20 bit value is extended by 12 bits at the low end to form the base address
11:8	0000b RO	**Reserved (RSV40):** Reserved.
7:0	00h RW	**Type (VAR_TYPE7):** Specifies the memory type for Variable Range 7

12.7.2.47 MTRR Variable Range Physical Mask 7 (MTRR_VAR_PHYSMASK7)—Offset 69h

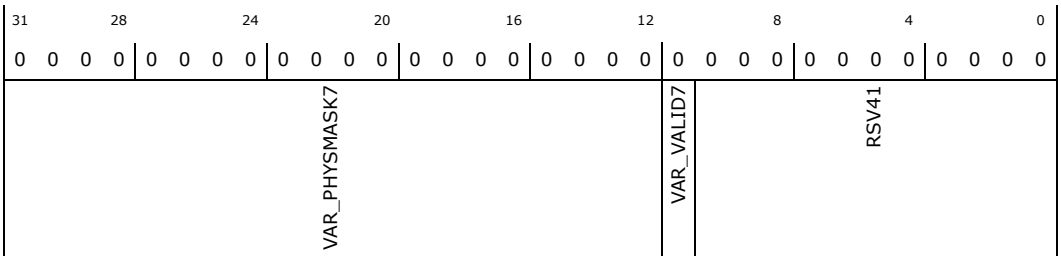
Access Method

Type: Message Bus Register (Size: 32 bits)

MTRR_VAR_PHYSMASK7: [Port: 0x03] + 69h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:12	00000h RW	**Physical Mask (VAR_PHYSMASK7):** Specifies a mask value for Variable Range 7. The mask determines the range of the region begin mapped. The mask value is extended by 12 bits at the low end to form the mask value.
11	0b RW	**Valid (VAR_VALID7):** Enables the register pair for Variable Range 7 when set and disables the register pair when clear.
10:0	0000h RO	**Reserved (RSV41):** Reserved.

12.7.3 Remote Management Unit (Port 0x04)

Table 71. Summary of Message Bus Registers—0x04

Offset	Register Name (Register Symbol)	Default Value
50h	"ECC Scrubber Configuration Register (P_CFG_50)—Offset 50h" on page 171	00530864h
60h	"SPI DMA Count Register (P_CFG_60)—Offset 60h" on page 172	00000000h
61h	"SPI DMA Destination Register (P_CFG_61)—Offset 61h" on page 173	00000000h
62h	"SPI DMA Source Register (P_CFG_62)—Offset 62h" on page 173	00000000h
70h	"Processor Register Block (P_BLK) Base Address (P_CFG_70)—Offset 70h" on page 174	00000000h
71h	"Control Register (P_CFG_71)—Offset 71h" on page 174	00000009h
74h	"Watchdog Control Register (P_CFG_74)—Offset 74h" on page 175	00040000h
76h	"ECC Scrubber Start Address Register (P_CFG_76)—Offset 76h" on page 176	00000000h
77h	"ECC Scrubber End Address Register (P_CFG_77)—Offset 77h" on page 176	00000000h
7Ch	"ECC Scrubber Next Address Register (P_CFG_7C)—Offset 7Ch" on page 177	00000000h
B0h	"Thermal Sensor Mode Register (P_CFG_B0)—Offset B0h" on page 178	00000000h
B1h	"Thermal Sensor Temperature Register (P_CFG_B1)—Offset B1h" on page 178	00000000h
B2h	"Thermal Sensor Programmable Trip Point Register (P_CFG_B2)—Offset B2h" on page 179	FFFFFFFFh

12.7.3.1 ECC Scrubber Configuration Register (P_CFG_50)—Offset 50h

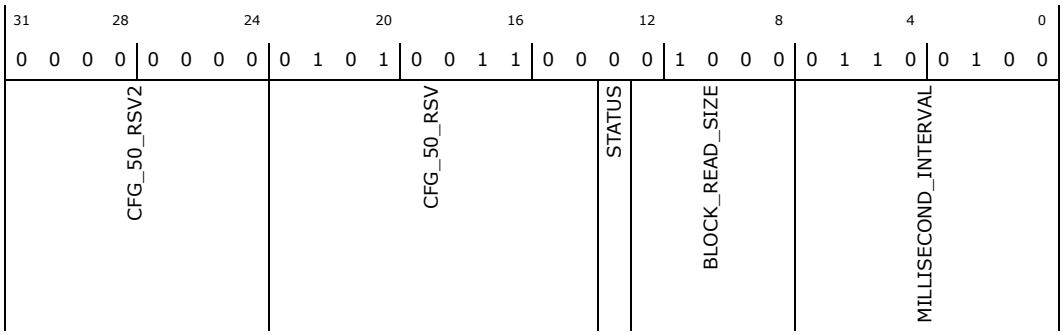
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 50h

Op Codes:
10h - Read, 11h - Write

Default: 00530864h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RO	**Reserved (CFG_50_RSV2):** Reserved.
23:14	14Ch RW	**Reserved (CFG_50_RSV):** Reserved.
13	0b RW	**Status (STATUS):** Indicates if ECC scrubbing is active or inactive. 0b : Scrubbing Inactive 1b : Scrubbing Active
12:8	08h RW	**Block Read Size (BLOCK_READ_SIZE):** Block read size in DWORDs (4 bytes). Range 1 to 128 (4-512 bytes).
7:0	64h RW	**Millisecond Interval (MILLISECOND_INTERVAL):** Millisecond interval between block reads. Range 1-255 msec.

12.7.3.2 SPI DMA Count Register (P_CFG_60)—Offset 60h

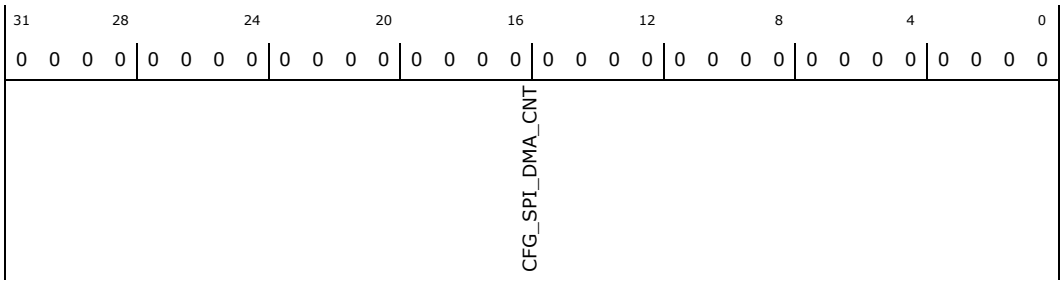
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 60h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**SPI DMA Count (CFG_SPI_DMA_CNT):** Count of 512 byte block transfers. Writing this register triggers the start of the transfer of the indicated number of blocks. Reading this register returns the number of blocks that are remaining to be transferred. A value of 0 indicates the transfer is complete.

12.7.3.3 SPI DMA Destination Register (P_CFG_61)—Offset 61h

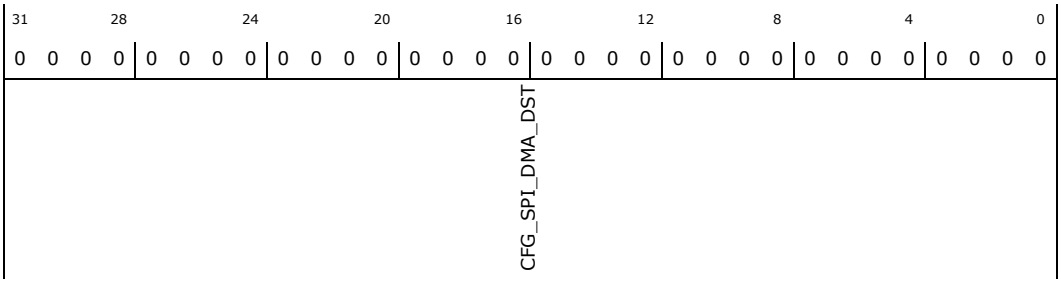
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 61h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**SPI DMA Destination (CFG_SPI_DMA_DST):** 32-bit Destination Address of data in System Memory (eSRAM/DRAM).

12.7.3.4 SPI DMA Source Register (P_CFG_62)—Offset 62h

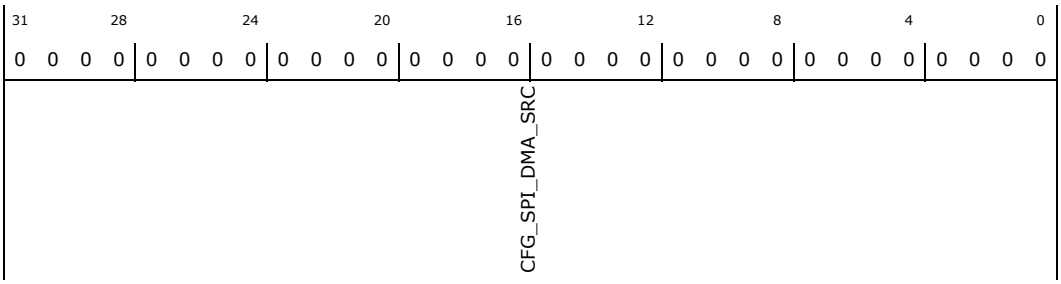
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 62h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**SPI DMA Source (CFG_SPI_DMA_SRC):** 32-bit Source Address of data in Legacy SPI.

12.7.3.5 Processor Register Block (P_BLK) Base Address (P_CFG_70)—Offset 70h

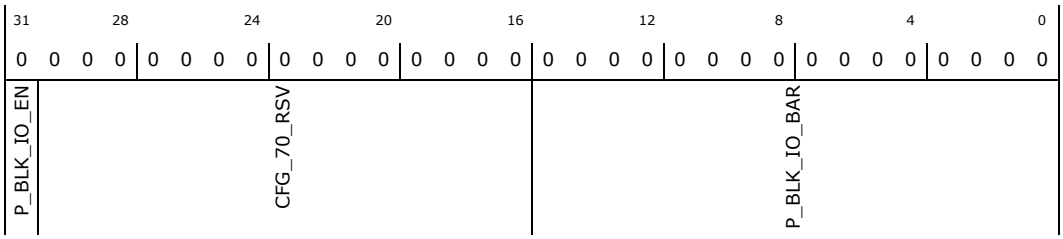
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 70h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Enable (P_BLK_IO_EN):** When set, decode of the IO range pointed to be the Base Address is enabled.
30:16	0b RO	**Reserved (CFG_70_RSV):** Reserved.
15:0	0b RW	**Base Address (P_BLK_IO_BAR):** IO Base Address for the Processor Register Block (P_BLK) decode range.

12.7.3.6 Control Register (P_CFG_71)—Offset 71h

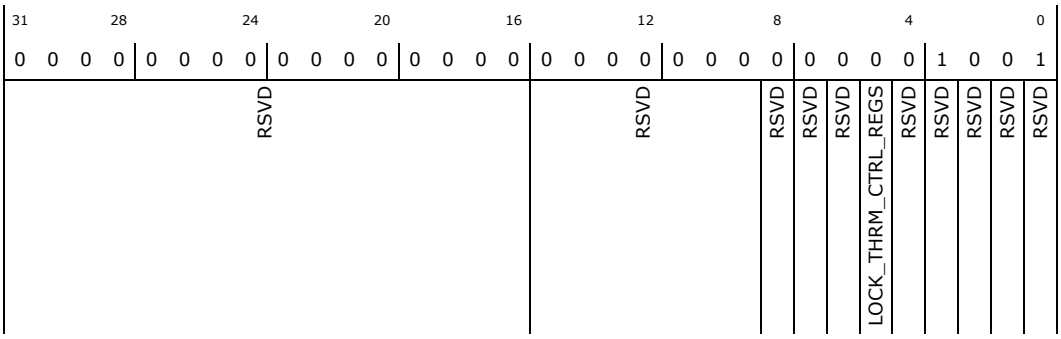
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 71h

Op Codes:
10h - Read, 11h - Write

Default: 00000009h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSVD):** Reserved.
15:9	0b RO	**Reserved (RSVD):** Reserved.
8	0b RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Reserved (RSVD):** Reserved.
6	0b RO	**Reserved (RSVD):** Reserved.
5	0b RW/O	**Lock Thermal Control Registers (LOCK_THRM_CTRL_REGS):** Lock the thermal control registers (registers 0xB0 and 0xB2).
4	0b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	0b RO	**Reserved (RSVD):** Reserved.
1	0b RO	**Reserved (RSVD):** Reserved.
0	1b RO	**Reserved (RSVD):** Reserved.

12.7.3.7 Watchdog Control Register (P_CFG_74)—Offset 74h

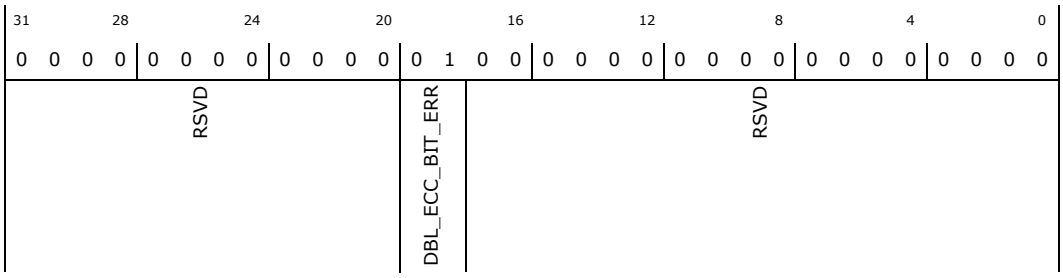
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 74h

Op Codes:
10h - Read, 11h - Write

Default: 00040000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0b RO	**Reserved (RSVD):** Reserved.
19:18	01b RW	**Double ECC Bit Error (DBL_ECC_BIT_ERR):** Double ECC bit error handling selection: 00b: Do nothing 01b: Catastrophic Shutdown 10b: Warm Reset 11b: Send SERR
17:0	0b RO	**Reserved (RSVD):** Reserved.

12.7.3.8 ECC Scrubber Start Address Register (P_CFG_76)—Offset 76h

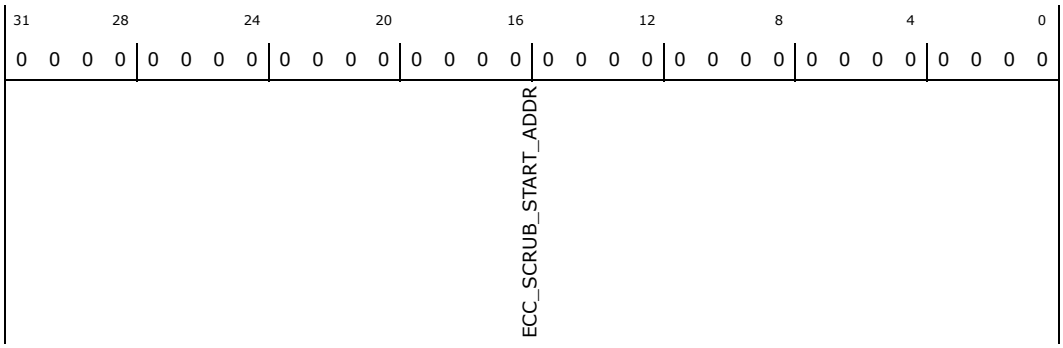
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 76h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Start Address (ECC_SCRUB_START_ADDR):** Start of memory address.

12.7.3.9 ECC Scrubber End Address Register (P_CFG_77)—Offset 77h

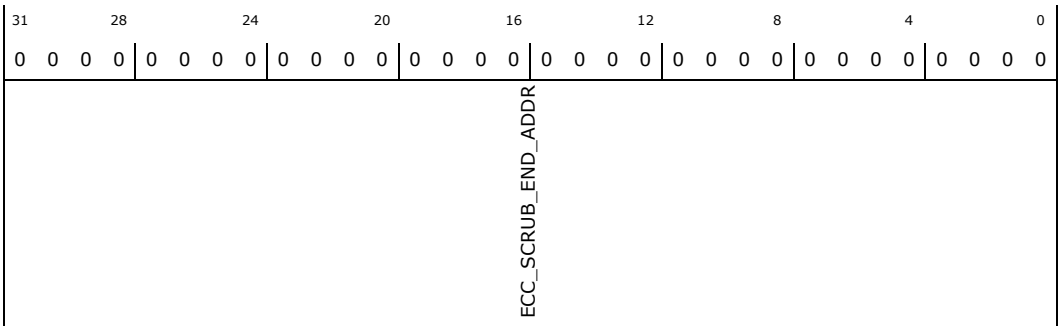
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 77h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**End Address (ECC_SCRUB_END_ADDR):** End of memory address.

12.7.3.10 ECC Scrubber Next Address Register (P_CFG_7C)—Offset 7Ch

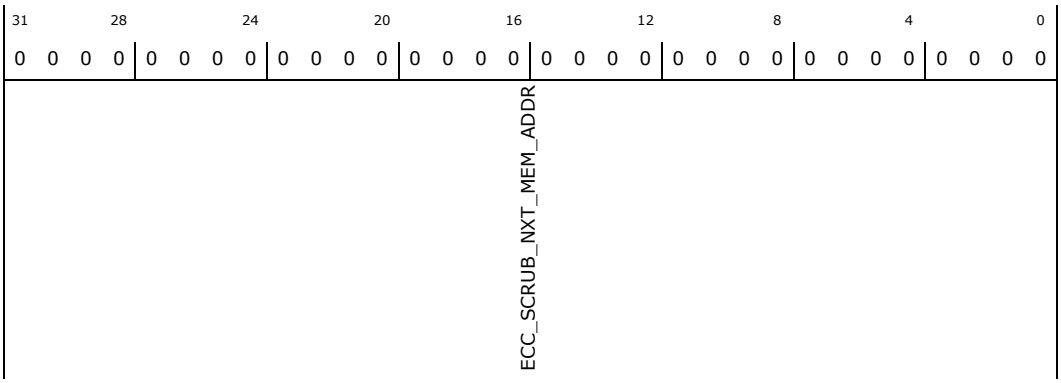
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + 7Ch

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Next Address (ECC_SCRUB_NXT_MEM_ADDR):** Next memory address to be scrubbed.

12.7.3.11 Thermal Sensor Mode Register (P_CFG_B0)—Offset B0h

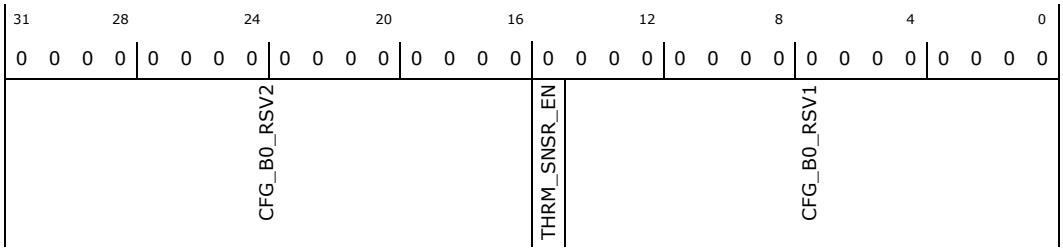
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + B0h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RW	**Reserved (CFG_B0_RSV2):** Reserved.
15	0b RW/L	**Thermal Sensor Enable (THRM_SNSR_EN):** Enables Thermal Sensor
14:0	0b RW	**Reserved (CFG_B0_RSV1):** Reserved.

12.7.3.12 Thermal Sensor Temperature Register (P_CFG_B1)—Offset B1h

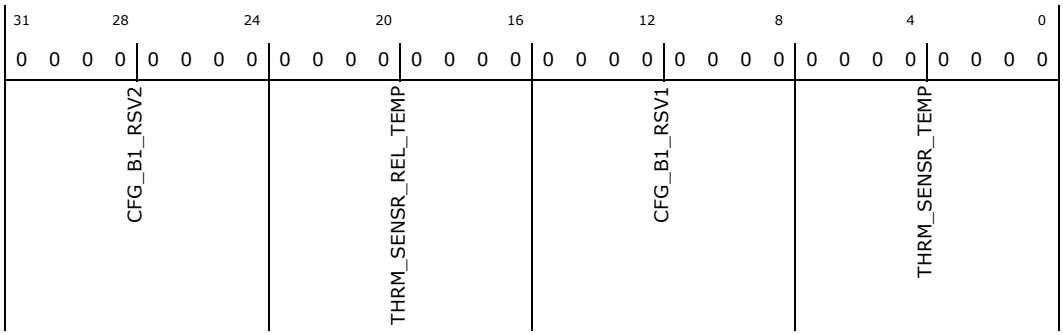
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + B1h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0b RO	**Reserved (CFG_B1_RSV2):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
23:16	0b RO	**Thermal Sensor Relative Temperature (THRM_SENSR_REL_TEMP):** The thermal sensor relative temperature value is an 8-bit signed value relative to the Hot Trip point. If the Hot Trip point minus the current temperature is greater than +127, this value is clipped at +127. If the Hot Trip point minus the current temperature is less than -127, this value is clipped at -127. Otherwise this value is a signed sum magnitude where bit[23] is the sign and bits[22:16] are the magnitude.
15:8	0b RO	**Reserved (CFG_B1_RSV1):** Reserved.
7:0	0b RO	**Thermal Sensor Temperature (THRM_SENSR_TEMP):** 8-bit Thermal Sensor Temperature. The temperature in degrees Celsius is calculated by subtracting an offset of 50 from the 8-bit register value. The temperature in degrees Celsius corresponds to: 00h: -50 01h: -49 ... FEh: 204 FFh: 205

12.7.3.13 Thermal Sensor Programmable Trip Point Register (P_CFG_B2)—Offset B2h

Access Method

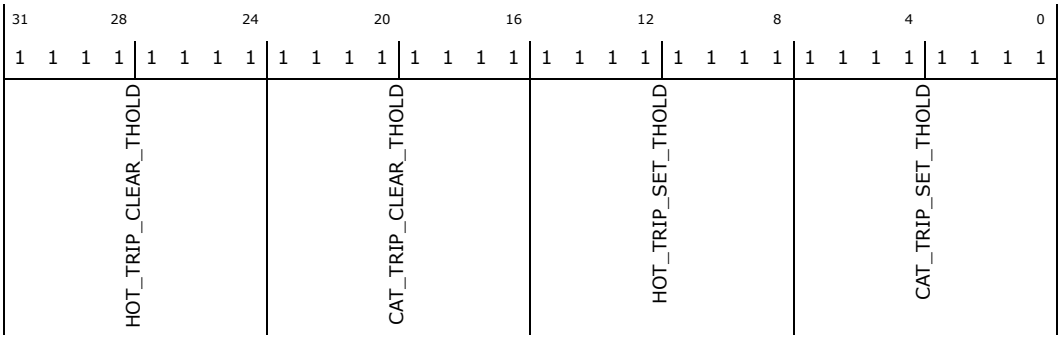
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x04] + B2h

Op Codes:

10h - Read, 11h - Write

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:24	FFh RW/L	**Hot Clear Trip Point Threshold (HOT_TRIP_CLEAR_THOLD):** Sets the target value for the hot trip clear point
23:16	FFh RW/L	**Catastrophic Clear Trip Point Threshold (CAT_TRIP_CLEAR_THOLD):** Sets the target value for the catastrophic trip clear point
15:8	FFh RW/L	**Hot Set Trip Point Threshold (HOT_TRIP_SET_THOLD):** Sets the target value for the hot trip set point
7:0	FFh RW/L	**Catastrophic Set Trip Point Threshold (CAT_TRIP_SET_THOLD):** Sets the target value for the catastrophic trip set point

12.7.4 Memory Manager (Port 0x05)

Table 72. Summary of Message Bus Registers—0x05

Offset	Register ID—Description	Default Value
1h	"Control (BCTRL)—Offset 1h" on page 181	00000800h
2h	"Write Flush Policy (BWFLUSH)—Offset 2h" on page 182	0C070408h
19h	"Isolated Memory Region Violation Control (BIMRVCTL)—Offset 19h" on page 183	00000000h
31h	"Debug 1 (DEBUG1)—Offset 31h" on page 184	4F08C20Ch
40h	"Isolated Memory Region 0 Low Address (IMR0L)—Offset 40h" on page 185	00000000h
41h	"Isolated Memory Region 0 High Address (IMR0H)—Offset 41h" on page 186	00000000h
42h	"Isolated Memory Region 0 Read Mask (IMR0RM)—Offset 42h" on page 186	BFFFFFFFh
43h	"Isolated Memory Region 0 Write Mask (IMR0WM)—Offset 43h" on page 188	FFFFFFFFh
44h	"Isolated Memory Region 1 Low Address (IMR1L)—Offset 44h" on page 190	00000000h
45h	"Isolated Memory Region 1 High Address (IMR1H)—Offset 45h" on page 191	00000000h
46h	"Isolated Memory Region 1 Read Mask (IMR1RM)—Offset 46h" on page 191	BFFFFFFFh
47h	"Isolated Memory Region 1 Write Mask (IMR1WM)—Offset 47h" on page 193	FFFFFFFFh
48h	"Isolated Memory Region 2 Low Address (IMR2L)—Offset 48h" on page 195	00000000h
49h	"Isolated Memory Region 2 High Address (IMR2H)—Offset 49h" on page 195	00000000h
4Ah	"Isolated Memory Region 2 Read Mask (IMR2RM)—Offset 4Ah" on page 196	BFFFFFFFh
4Bh	"Isolated Memory Region 2 Write Mask (IMR2WM)—Offset 4Bh" on page 198	FFFFFFFFh
4Ch	"Isolated Memory Region 3 Low Address (IMR3L)—Offset 4Ch" on page 199	00000000h
4Dh	"Isolated Memory Region 3 High Address (IMR3H)—Offset 4Dh" on page 200	00000000h
4Eh	"Isolated Memory Region 3 Read Mask (IMR3RM)—Offset 4Eh" on page 200	BFFFFFFFh
4Fh	"Isolated Memory Region 3 Write Mask (IMR3WM)—Offset 4Fh" on page 202	FFFFFFFFh
50h	"Isolated Memory Region 4 Low Address (IMR4L)—Offset 50h" on page 204	00000000h
51h	"Isolated Memory Region 4 High Address (IMR4H)—Offset 51h" on page 205	00000000h
52h	"Isolated Memory Region 4 Read Mask (IMR4RM)—Offset 52h" on page 205	BFFFFFFFh
53h	"Isolated Memory Region 4 Write Mask (IMR4WM)—Offset 53h" on page 207	FFFFFFFFh
54h	"Isolated Memory Region 5 Low Address (IMR5L)—Offset 54h" on page 209	00000000h
55h	"Isolated Memory Region 5 High Address (IMR5H)—Offset 55h" on page 209	00000000h
56h	"Isolated Memory Region 5 Read Mask (IMR5RM)—Offset 56h" on page 210	BFFFFFFFh
57h	"Isolated Memory Region 5 Write Mask (IMR5WM)—Offset 57h" on page 212	FFFFFFFFh
58h	"Isolated Memory Region 6 Low Address (IMR6L)—Offset 58h" on page 213	00000000h
59h	"Isolated Memory Region 6 High Address (IMR6H)—Offset 59h" on page 214	00000000h
5Ah	"Isolated Memory Region 6 Read Mask (IMR6RM)—Offset 5Ah" on page 214	BFFFFFFFh
5Bh	"Isolated Memory Region 6 Write Mask (IMR6WM)—Offset 5Bh" on page 216	FFFFFFFFh
5Ch	"Isolated Memory Region 7 Low Address (IMR7L)—Offset 5Ch" on page 218	00000000h
5Dh	"Isolated Memory Region 7 High Address (IMR7H)—Offset 5Dh" on page 219	00000000h
5Eh	"Isolated Memory Region 7 Read Mask (IMR7RM)—Offset 5Eh" on page 219	BFFFFFFFh
5Fh	"Isolated Memory Region 7 Write Mask (IMR7WM)—Offset 5Fh" on page 221	FFFFFFFFh
81h	"eSRAM Control (ESRAMCTRL)—Offset 81h" on page 223	047F3F91h
82h	"eSRAM Block Page Control (ESRAMPGCTRL_BLOCK)—Offset 82h" on page 224	850000FFh

Table 72. Summary of Message Bus Registers—0x05 (Continued)

Offset	Register ID—Description	Default Value
83h	"eSRAM Correctable Error (ESRAMCERR)—Offset 83h" on page 225	00000000h
84h	"eSRAM Uncorrectable Error (ESRAMUERR)—Offset 84h" on page 226	00000000h
88h	"eSRAM ECC Error Syndrome (ESRAMSDROME)—Offset 88h" on page 227	00000000h

12.7.4.1 Control (BCTRL)—Offset 1h

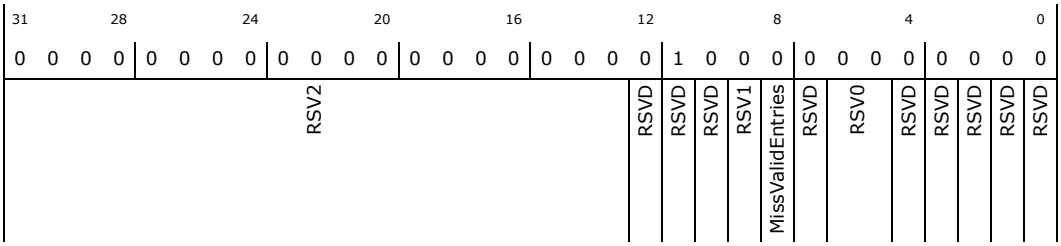
Access Method

Type: Message Bus Register (Size: 32 bits)

BCTRL: [Port: 0x05] + 1h

Op Codes:
10h - Read, 11h - Write

Default: 00000800h



Bit Range	Default & Access	Description
31:13	0000h RO	**Reserved (RSV2):** Reserved.
12	0b RO	**Reserved (RSVD):** Reserved.
11	1b RO	**Reserved (RSVD):** Reserved.
10	0b RO	**Reserved (RSVD):** Reserved.
9	0b RO	**Reserved (RSV1):** Reserved.
8	0b RW	**Miss Valid Entries (MissValidEntries):** This mode, mostly present for test purposes, causes reads to clean valid Memory Manager buffer entries with zero reference counts to look like misses instead of hits
7	0b RO	**Reserved (RSVD):** Reserved.
6:5	0h RO	**Reserved (RSV0):** Reserved.
4	0b RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Reserved (RSVD):** Reserved.
2	0b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
1	0b RO	**Reserved (RSVD):** Reserved.
0	0b RO	**Reserved (RSVD):** Reserved.

12.7.4.2 Write Flush Policy (BWFLUSH)—Offset 2h

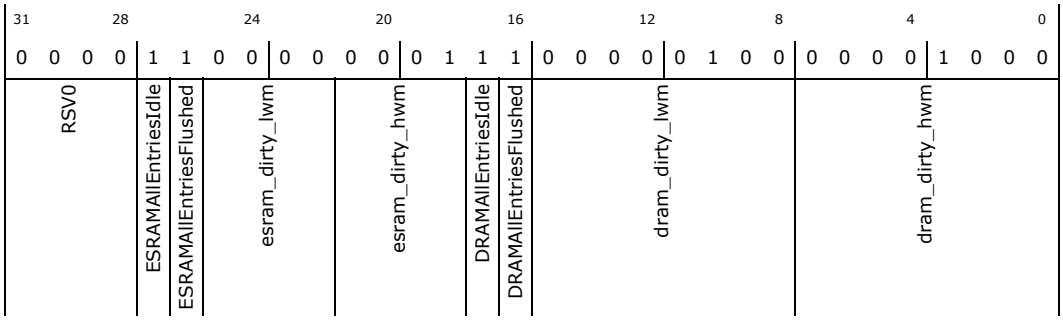
Access Method

Type: Message Bus Register (Size: 32 bits)

BWFLUSH: [Port: 0x05] + 2h

Op Codes:
10h - Read, 11h - Write

Default: 0C070408h



Bit Range	Default & Access	Description
31:28	0h RO	**Reserved (RSV0):** Reserved.
27	1b RO	**eSRAM All Entries Idle (ESRAMAllEntriesIdle):** All entries in the Memory Manager eCACHE tag store have 0 reference counts and are unlocked, indicating that there are no transactions in progress in this cache
26	1b RO	**eSRAM All Entries Flushed (ESRAMAllEntriesFlushed):** All entries in the Memory Manager eCACHE have been flushed to the eSRAM
25:22	00h RW	**eSRAM Low Water Mark (esram_dirty_lwm):** Low Water Mark for Dirty Entries retained by eCACHE in the Memory Manager
21:18	01h RW	**eSRAM High Water Mark (esram_dirty_hwm):** High Water Mark for Dirty Entries retained by eCACHE in the Memory Manager
17	1b RO	**DRAM All Entries Idle (DRAMAllEntriesIdle):** All entries in the Memory Manager DCACHE entries have 0 reference counts and are unlocked, indicating that the Memory Manager has no DRAM transactions in progress
16	1b RO	**DRAM All Entries Flushed (DRAMAllEntriesFlushed):** All entries in the Memory Manager DCACHE have been flushed
15:8	04h RW	**DRAM Low Water Mark (dram_dirty_lwm):** Low water mark for dirty entries retained by the Memory Manager
7:0	08h RW	**DRAM High Water Mark (dram_dirty_hwm):** High water mark for dirty entries retained by the Memory Manager

12.7.4.3 Isolated Memory Region Violation Control (BIMRVCTL)—Offset 19h

This register is used to configure the interrupt and to capture status when an IMR is violated. Note that the Enable Interrupt on IMR Violation field is reset on an IMR violation event and this register captures the first violation only. It can be set back to 1, and if a new violation occurs any status previously captured is overridden with the new violation details. This register is not secured or locked because the violation interrupt is not part of securing the region and is only intended to help in debugging IMR configuration.

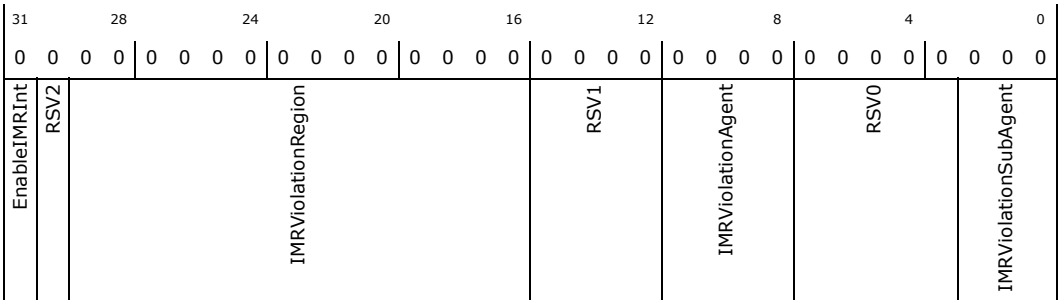
Access Method

Type: Message Bus Register (Size: 32 bits)

BIMRVCTL: [Port: 0x05] + 19h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Enable Interrupt on IMR Violation (EnableIMRInt):** When set, the Memory Manager will latch violation information into this register and send an interrupt request to the Remote Management Unit. This bit is cleared upon triggering and must be reset by software in order to trigger again. Memory protection is maintained even while the interrupt/capture mechanism is disabled
30	0b RO	**Reserved (RSV2):** Reserved.
29:16	0000h RO	**IMR Violation Region (IMRViolationRegion):** This 14-bit value indicates which region the last IMR violation occurred on if the IMR interrupt is enabled. A bit of this field will be asserted for every IMR region, HMBOUND or SMM region violated by the transaction that caused EnableIMRInt to deassert. [13] : HMBOUND Violation [12] : SMM Bound Violation [11:8] : Reserved [7] : IMR7 Violation [6] : IMR6 Violation [5] : IMR5 Violation [4] : IMR4 Violation [3] : IMR3 Violation [2] : IMR2 Violation [1] : IMR1 Violation [0] : IMR0 Violation
15:12	0h RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Description
11:8	0h RO	**IMR Violation Agent (IMRViolationAgent):** This 4-bit value indicates which agent caused the last IMR violation if the IMR interrupt is enabled: 0000b : CPU 0001b : Host Bridge Arbiter VC0 0010b : Host Bridge Arbiter VC1 0011b : Reserved 0100b : Reserved 0101b : Reserved 0110b : Reserved 0111b : eSRAM Flush/Init 1000b : Remote Management Unit
7:3	00h RO	**Reserved (RSV0):** Reserved.
2:0	0h RO	**IMR Violation Sub Agent (IMRViolationSubAgent):** This 3-bit value indicates which sub-agent caused the last IMR violation, if the IMR interrupt is enabled 000b : Host Bridge Arbiter Sub-Channel 0 (Anonymous) 001b : Host Bridge Arbiter Sub-Channel 1 010b : Host Bridge Arbiter Sub-Channel 2 100b : Host Bridge Arbiter Sub-Channel 3

12.7.4.4 Debug 1 (DEBUG1)—Offset 31h

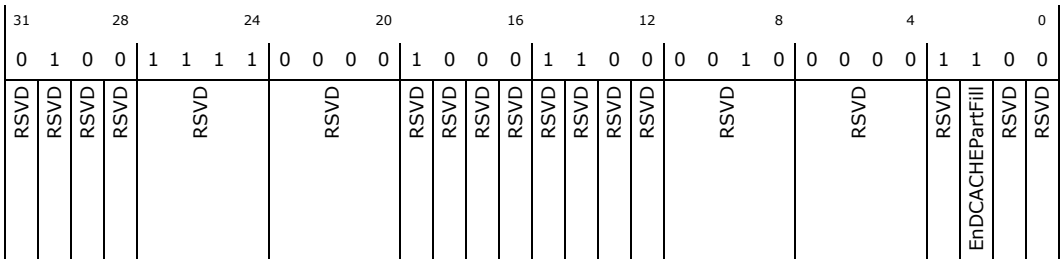
Access Method

Type: Message Bus Register (Size: 32 bits)

DEBUG1: [Port: 0x05] + 31h

Op Codes:
10h - Read, 11h - Write

Default: 4F08C20Ch



Bit Range	Default & Access	Description
31	0b RO	**Reserved (RSVD):** Reserved.
30	1b RO	**Reserved (RSVD):** Reserved.
29	0b RO	**Reserved (RSVD):** Reserved.
28	0h RO	**Reserved (RSVD):** Reserved.
27:24	Fh RO	**Reserved (RSVD):** Reserved.
23:20	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
19	1b RO	**Reserved (RSVD):** Reserved.
18	0b RO	**Reserved (RSVD):** Reserved.
17	0b RO	**Reserved (RSVD):** Reserved.
16	0b RO	**Reserved (RSVD):** Reserved.
15	1b RO	**Reserved (RSVD):** Reserved.
14	1b RO	**Reserved (RSVD):** Reserved.
13	0h RO	**Reserved (RSVD):** Reserved.
12	0h RO	**Reserved (RSVD):** Reserved.
11:8	2h RO	**Reserved (RSVD):** Reserved.
7:4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RW	**Enable DCACHE Partial Entries (EnDCACHEPartFill):** If the Memory Manager admitted Request or CPU Snoop Response that has not all Byte Enables set the Entry is considered Partial. Setting this bit will cause the Memory Manager to fill the partial entry from DRAM, before it is being flushed to DRAM. This bit needs to be set if the Memory Controller enables ECC mode, where partial writes are forbidden
1	0b RO	**Reserved (RSVD):** Reserved.
0	0b RO	**Reserved (RSVD):** Reserved.

12.7.4.5 Isolated Memory Region 0 Low Address (IMR0L)—Offset 40h

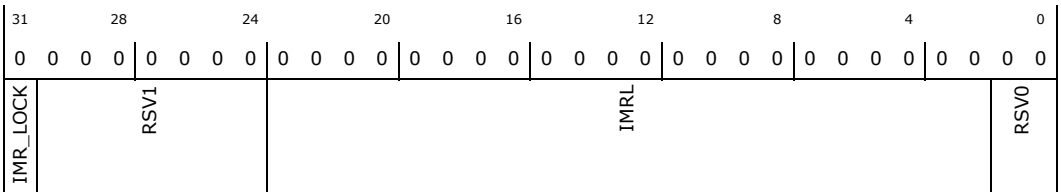
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR0L: [Port: 0x05] + 40h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.6 Isolated Memory Region 0 High Address (IMR0H)—Offset 41h

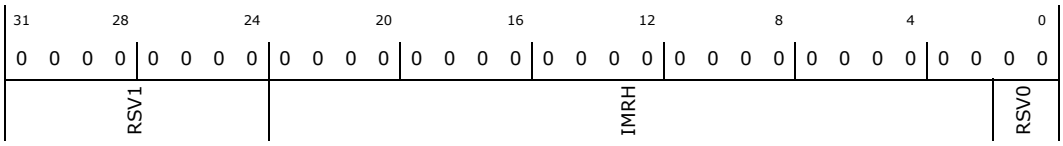
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR0H: [Port: 0x05] + 41h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.7 Isolated Memory Region 0 Read Mask (IMR0RM)—Offset 42h

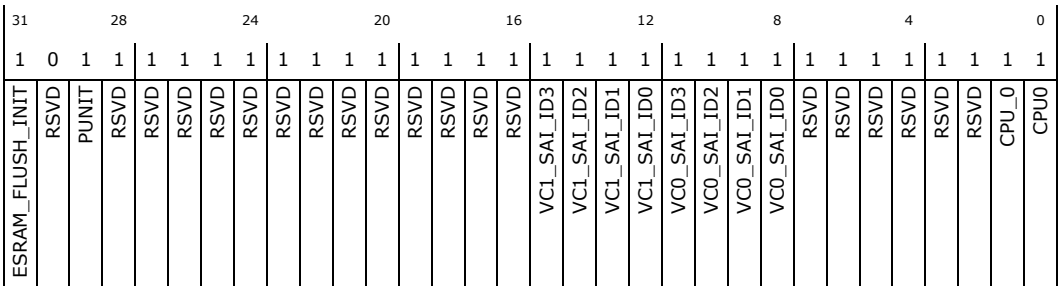
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR0RM: [Port: 0x05] + 42h

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.8 Isolated Memory Region 0 Write Mask (IMR0WM)—Offset 43h

Access Method

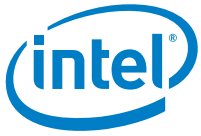
Type: Message Bus Register (Size: 32 bits)

IMR0WM: [Port: 0x05] + 43h

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.9 Isolated Memory Region 1 Low Address (IMR1L)—Offset 44h

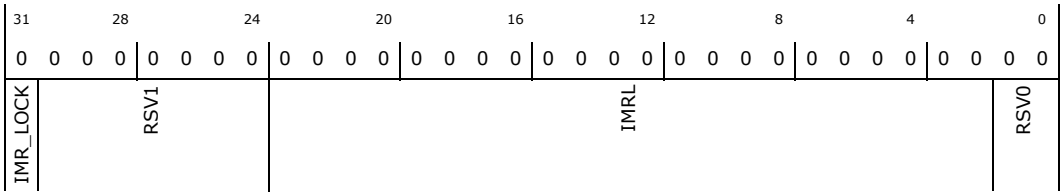
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR1L: [Port: 0x05] + 44h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.10 Isolated Memory Region 1 High Address (IMR1H)—Offset 45h

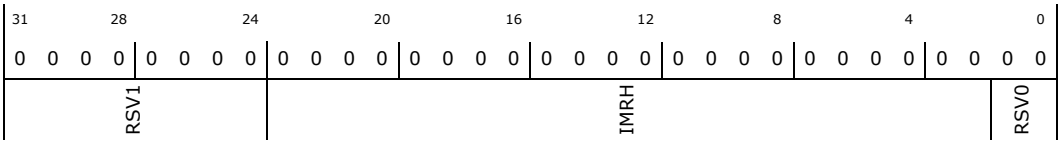
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR1H: [Port: 0x05] + 45h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.11 Isolated Memory Region 1 Read Mask (IMR1RM)—Offset 46h

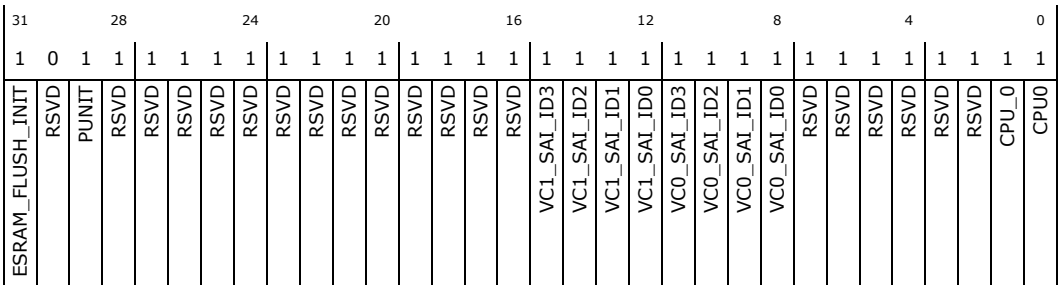
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR1RM: [Port: 0x05] + 46h

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.12 Isolated Memory Region 1 Write Mask (IMR1WM)—Offset 47h

Access Method

Type: Message Bus Register (Size: 32 bits)

IMR1WM: [Port: 0x05] + 47h

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0

Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

Bit Range	Default & Access	Description
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.13 Isolated Memory Region 2 Low Address (IMR2L)—Offset 48h

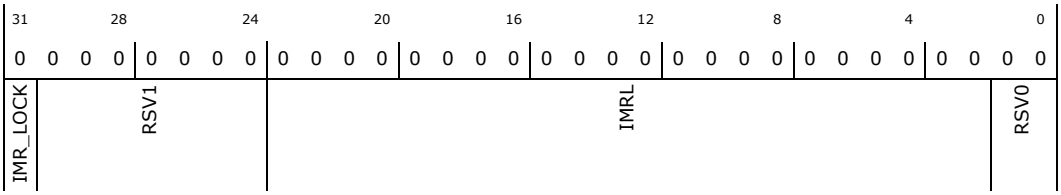
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR2L: [Port: 0x05] + 48h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.14 Isolated Memory Region 2 High Address (IMR2H)—Offset 49h

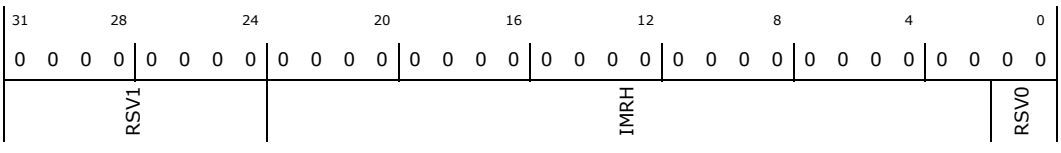
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR2H: [Port: 0x05] + 49h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Description
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.15 Isolated Memory Region 2 Read Mask (IMR2RM)—Offset 4Ah

Access Method

Type: Message Bus Register (Size: 32 bits)

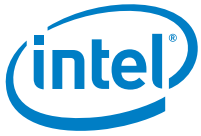
IMR2RM: [Port: 0x05] + 4Ah

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	RSVD	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0

Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.



Bit Range	Default & Access	Description
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.16 Isolated Memory Region 2 Write Mask (IMR2WM)—Offset 4Bh

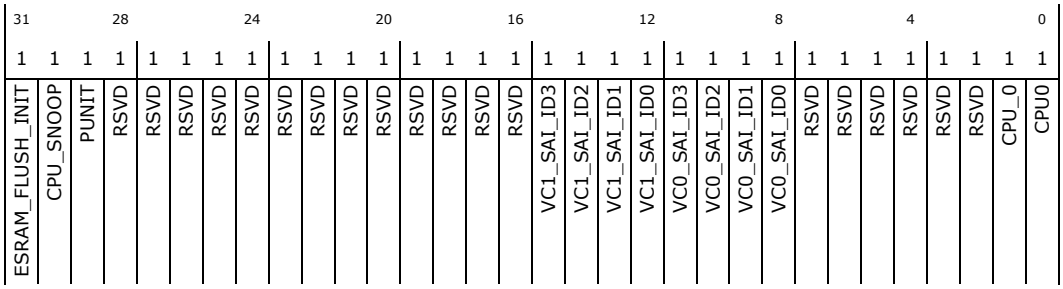
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR2WM: [Port: 0x05] + 4Bh

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.17 Isolated Memory Region 3 Low Address (IMR3L)—Offset 4Ch

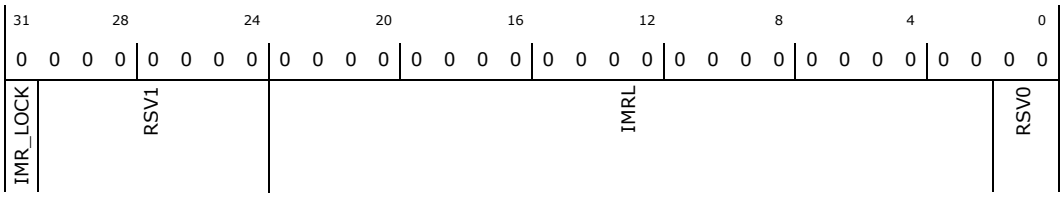
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR3L: [Port: 0x05] + 4Ch

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.18 Isolated Memory Region 3 High Address (IMR3H)—Offset 4Dh

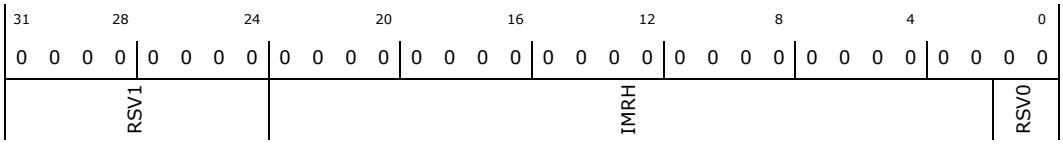
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR3H: [Port: 0x05] + 4Dh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.19 Isolated Memory Region 3 Read Mask (IMR3RM)—Offset 4Eh

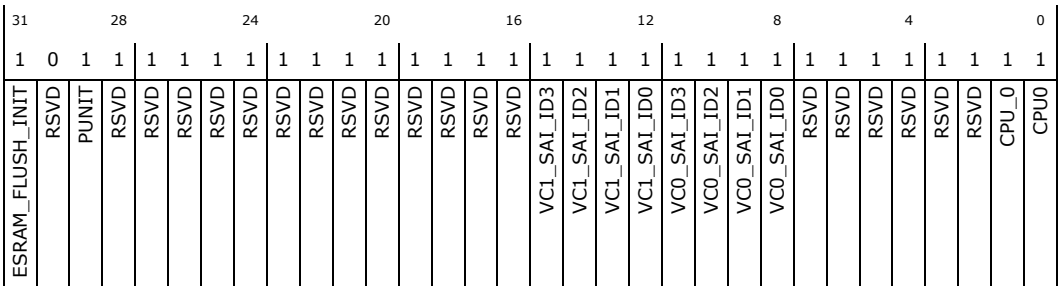
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR3RM: [Port: 0x05] + 4Eh

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.20 Isolated Memory Region 3 Write Mask (IMR3WM)—Offset 4Fh

Access Method

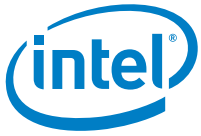
Type: Message Bus Register (Size: 32 bits)

IMR3WM: [Port: 0x05] + 4Fh

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.21 Isolated Memory Region 4 Low Address (IMR4L)—Offset 50h

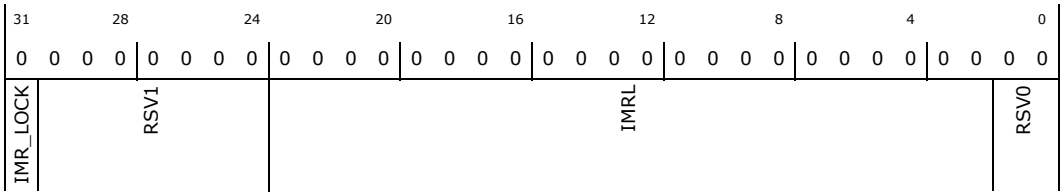
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR4L: [Port: 0x05] + 50h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.22 Isolated Memory Region 4 High Address (IMR4H)—Offset 51h

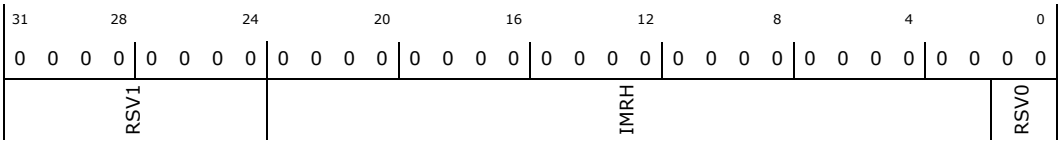
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR4H: [Port: 0x05] + 51h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.23 Isolated Memory Region 4 Read Mask (IMR4RM)—Offset 52h

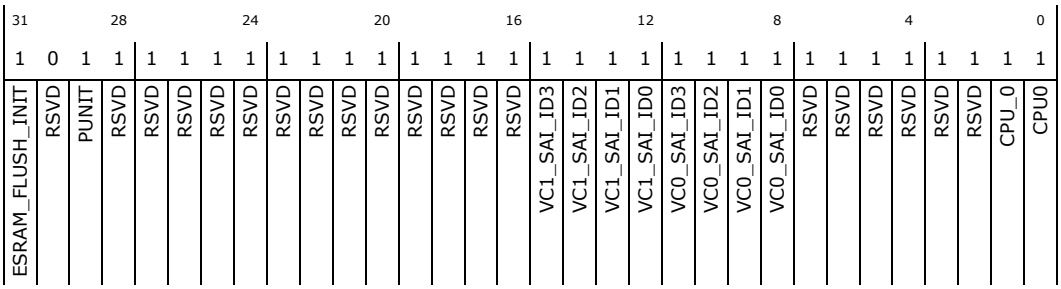
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR4RM: [Port: 0x05] + 52h

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.24 Isolated Memory Region 4 Write Mask (IMR4WM)—Offset 53h

Access Method

Type: Message Bus Register (Size: 32 bits)

IMR4WM: [Port: 0x05] + 53h

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0

Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

Bit Range	Default & Access	Description
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.25 Isolated Memory Region 5 Low Address (IMR5L)—Offset 54h

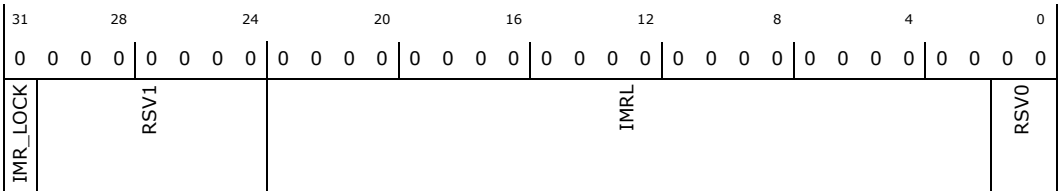
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR5L: [Port: 0x05] + 54h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.26 Isolated Memory Region 5 High Address (IMR5H)—Offset 55h

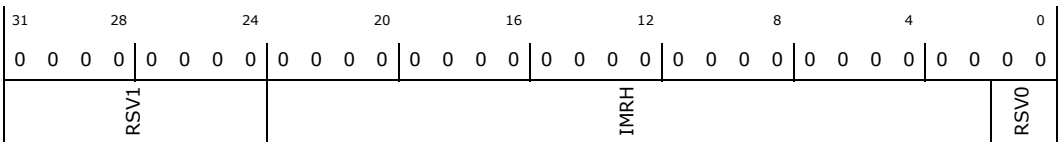
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR5H: [Port: 0x05] + 55h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Description
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.27 Isolated Memory Region 5 Read Mask (IMR5RM)—Offset 56h

Access Method

Type: Message Bus Register (Size: 32 bits)

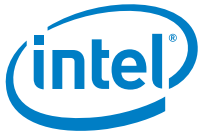
IMR5RM: [Port: 0x05] + 56h

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	RSVD	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0

Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.



Bit Range	Default & Access	Description
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.28 Isolated Memory Region 5 Write Mask (IMR5WM)—Offset 57h

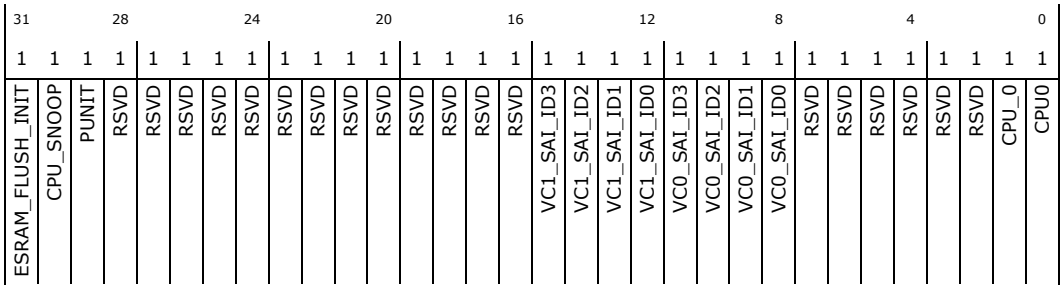
Access Method

Type: Message Bus Register (Size: 32 bits)

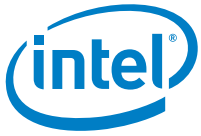
IMR5WM: [Port: 0x05] + 57h

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.



Bit Range	Default & Access	Description
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.29 Isolated Memory Region 6 Low Address (IMR6L)—Offset 58h

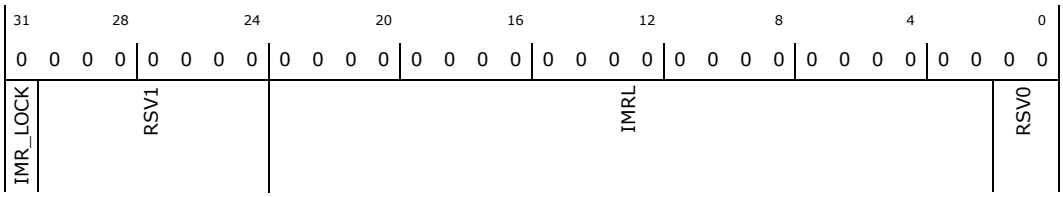
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR6L: [Port: 0x05] + 58h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.30 Isolated Memory Region 6 High Address (IMR6H)—Offset 59h

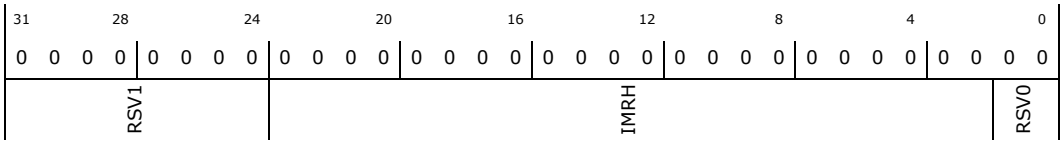
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR6H: [Port: 0x05] + 59h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.31 Isolated Memory Region 6 Read Mask (IMR6RM)—Offset 5Ah

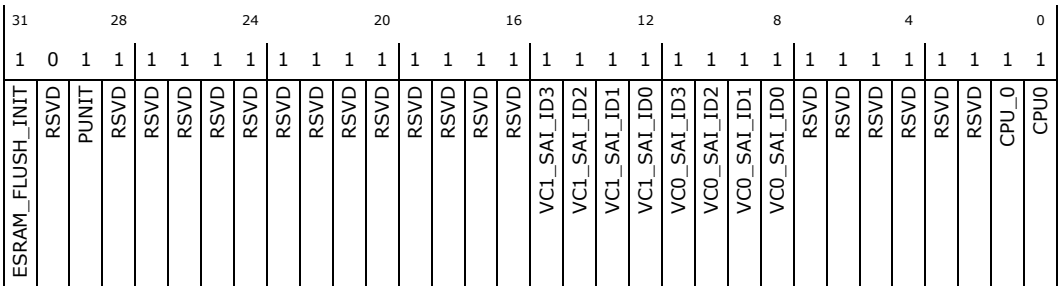
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR6RM: [Port: 0x05] + 5Ah

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.32 Isolated Memory Region 6 Write Mask (IMR6WM)—Offset 5Bh

Access Method

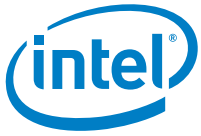
Type: Message Bus Register (Size: 32 bits)

IMR6WM: [Port: 0x05] + 5Bh

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.

Bit Range	Default & Access	Description
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.33 Isolated Memory Region 7 Low Address (IMR7L)—Offset 5Ch

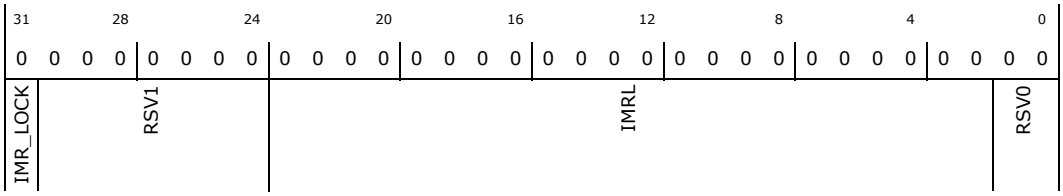
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR7L: [Port: 0x05] + 5Ch

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/O	**IMR Lock (IMR_LOCK):** This bit locks the IMRX registers, preventing further updates
30:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR Low Address (IMRL):** These bits are compared with bits 31:10 of the incoming address to determine the lower 1KB aligned value of the protected range
1:0	00h RO	**Reserved (RSV0):** Reserved.

12.7.4.34 Isolated Memory Region 7 High Address (IMR7H)—Offset 5Dh

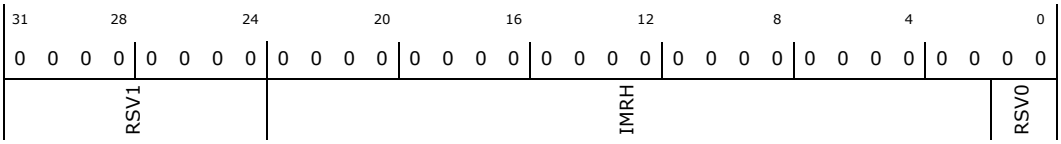
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR7H: [Port: 0x05] + 5Dh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**Reserved (RSV1):** Reserved.
23:2	000000h RW/L	**IMR High Address (IMRH):** These bits are compared with bits 31:10 of the incoming address to determine the upper 1KB aligned value of the protected range
1:0	0h RO	**Reserved (RSV0):** Reserved.

12.7.4.35 Isolated Memory Region 7 Read Mask (IMR7RM)—Offset 5Eh

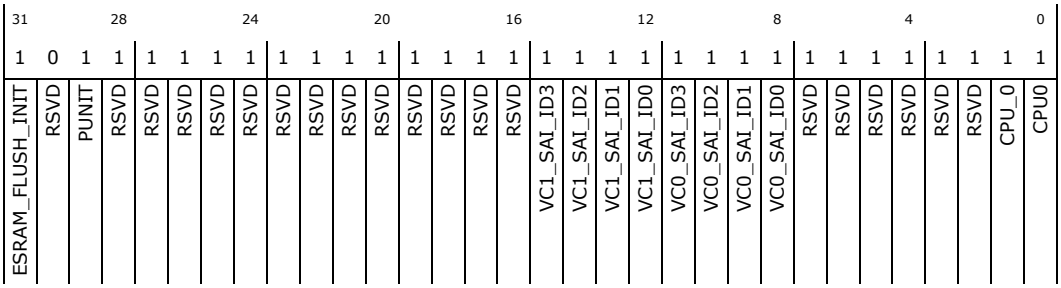
Access Method

Type: Message Bus Register (Size: 32 bits)

IMR7RM: [Port: 0x05] + 5Eh

Op Codes:
10h - Read, 11h - Write

Default: BFFFFFFFh



Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Read Access Allowed to memory delineated by IMRxL and IMRxH
30	0b RO	**Reserved (RSVD):** Reserved.
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Read Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Read Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value
0	1b RW/L	**CPU (CPU0):** CPU Read Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.36 Isolated Memory Region 7 Write Mask (IMR7WM)—Offset 5Fh

Access Method

Type: Message Bus Register (Size: 32 bits)

IMR7WM: [Port: 0x05] + 5Fh

Op Codes:
10h - Read, 11h - Write

Default: FFFFFFFFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ESRAM_FLUSH_INIT	CPU_SNOOP	PUNIT	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	VC1_SAI_ID3	VC1_SAI_ID2	VC1_SAI_ID1	VC1_SAI_ID0	VC0_SAI_ID3	VC0_SAI_ID2	VC0_SAI_ID1	VC0_SAI_ID0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	CPU_0	CPU0

Bit Range	Default & Access	Description
31	1b RW/L	**eSRAM Flush/Init (ESRAM_FLUSH_INIT):** eSRAM Flush/Init Write Access Allowed to memory delineated by IMRxL and IMRxH
30	1b RW/L	**CPU Snoop (CPU_SNOOP):** Dirty CPU Snoop Response Write Allowed to memory delineated by IMRxL and IMRxH. Clean snoop responses are always allowed and does not cause an IMR violation
29	1b RW/L	**Remote Management Unit (PUNIT):** Remote Management Unit Write Access Allowed to memory delineated by IMRxL and IMRxH
28	1b RO	**Reserved (RSVD):** Reserved.
27	1b RO	**Reserved (RSVD):** Reserved.
26	1b RO	**Reserved (RSVD):** Reserved.
25	1b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
23	1b RO	**Reserved (RSVD):** Reserved.
22	1b RO	**Reserved (RSVD):** Reserved.
21	1b RO	**Reserved (RSVD):** Reserved.
20	1b RO	**Reserved (RSVD):** Reserved.
19	1b RO	**Reserved (RSVD):** Reserved.
18	1b RO	**Reserved (RSVD):** Reserved.
17	1b RO	**Reserved (RSVD):** Reserved.
16	1b RO	**Reserved (RSVD):** Reserved.
15	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 3 (VC1_SAI_ID3):** Host Bridge Arbiter VC1 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
14	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 2 (VC1_SAI_ID2):** Host Bridge Arbiter VC1 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
13	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 1 (VC1_SAI_ID1):** Host Bridge Arbiter VC1 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
12	1b RW/L	**Host Bridge Arbiter VC1 Sub-Channel 0 (VC1_SAI_ID0):** Host Bridge Arbiter VC1 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
11	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 3 (VC0_SAI_ID3):** Host Bridge Arbiter VC0 Sub-Channel 3 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
10	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 2 (VC0_SAI_ID2):** Host Bridge Arbiter VC0 Sub-Channel 2 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
9	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 1 (VC0_SAI_ID1):** Host Bridge Arbiter VC0 Sub-Channel 1 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
8	1b RW/L	**Host Bridge Arbiter VC0 Sub-Channel 0 (VC0_SAI_ID0):** Host Bridge Arbiter VC0 Sub-Channel 0 Write Accesses Allowed to memory delineated by IMRxL and IMRxH.
7	1b RO	**Reserved (RSVD):** Reserved.
6	1b RO	**Reserved (RSVD):** Reserved.
5	1b RO	**Reserved (RSVD):** Reserved.
4	1b RO	**Reserved (RSVD):** Reserved.
3	1b RO	**Reserved (RSVD):** Reserved.
2	1b RO	**Reserved (RSVD):** Reserved.
1	1b RW/L	**CPU (CPU_0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

Bit Range	Default & Access	Description
0	1b RW/L	**CPU (CPU0):** CPU Write Access Allowed to memory region delineated by IMRxL and IMRxH. Note: Bit[0] and bit [1] of the IMR Read Mask register must always be programmed to the same value

12.7.4.37 eSRAM Control (ESRAMCTRL)—Offset 81h

Provides control of attributes which affect all eSRAM pages.

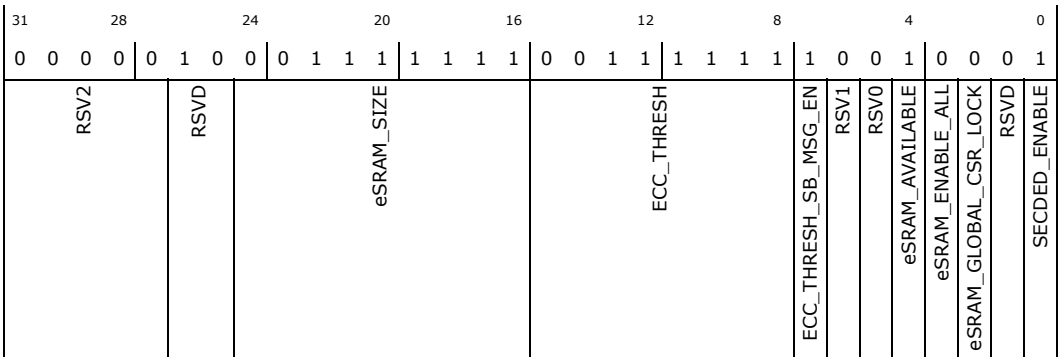
Access Method

Type: Message Bus Register (Size: 32 bits)

ESRAMCTRL: [Port: 0x05] + 81h

Op Codes:
10h - Read, 11h - Write

Default: 047F3F91h



Bit Range	Default & Access	Description
31:27	0h RO	**Reserved (RSV2):** Reserved.
26:25	2h RO	**Reserved (RSVD):** Reserved.
24:16	07Fh RO	**eSRAM Size (eSRAM_SIZE):** eSRAM size in 4k pages (0 means 1)
15:8	3Fh RW/L	**ECC Threshold (ECC_THRESH):** Total correctable ECC threshold until the eSRAM Correctable Error Threshold Reached message (opcode 0xD8) is sent to Remote Management Unit. Valid values are 0x1-0xFF. 0x0 may be used as a test mode to send the message immediately without waiting for occurrence of any correctable ECC errors. Once the message has been sent in this way, ESRAMCERR. CORRECTABLE_ERR_CNT_RST must be written with 1, before another message can be generated using this test mode. Note that the test mode generation of an eSRAM Correctable Error Threshold Reached message is not dependent on the value of ESRAMCTRL.SECDED_ENABLE.
7	1h RW/L	**ECC Threshold Message Enable (ECC_THRESH_SB_MSG_EN):** Set to 1 to enable the message to the Remote Management Unit which is generated when ECC_THRESH correctable ECC errors have been generated. If this field is 0, no ECC threshold message will be generated.
6	0h RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Description
5	0h RO	**Reserved (RSV0):** Reserved.
4	1h RO	**eSRAM Available (eSRAM_AVAILABLE):** Indicates eSRAM is available. Used as qualifier for eSRAM_SIZE.
3	0h RW/HC	**eSRAM Enable All Ranges (eSRAM_ENABLE_ALL):** Used during the early BIOS stage to enable eSRAM mapping into the system address space. Forces all eSRAM pages which are not already enabled, and are unlocked to be ECC initialised and enabled, and stays 0x1 until all such pages have been initialised. NOTE : This is a locking field, locks on ESRAM_GLOBAL_CSR_LOCK.
2	0h RW	**eSRAM Global CSR Lock (eSRAM_GLOBAL_CSR_LOCK):** When set to 1, all eSRAM global and page (4KB and 512KB) registers are locked. A locked page can still be flushed if FLUSH_PG_ENABLE/BLOCK_FLUSH_PG_ENABLE is set to 1. Once set, this field can only be cleared by a warm reset. This is a locking field, it is locked by being set to 1.
1	0h RO	**Reserved (RSVD):** Reserved.
0	1h RW/L	**SECDED Enable (SECDED_ENABLE):** SECDED ECC enable for the eSRAM memory array.

12.7.4.38 eSRAM Block Page Control (ESRAMPGCTRL_BLOCK)—Offset 82h

This register allows all eSRAM pages to be mapped and controlled as a single 512KB block page. If this page is enabled, no 4KB pages may be individually mapped.

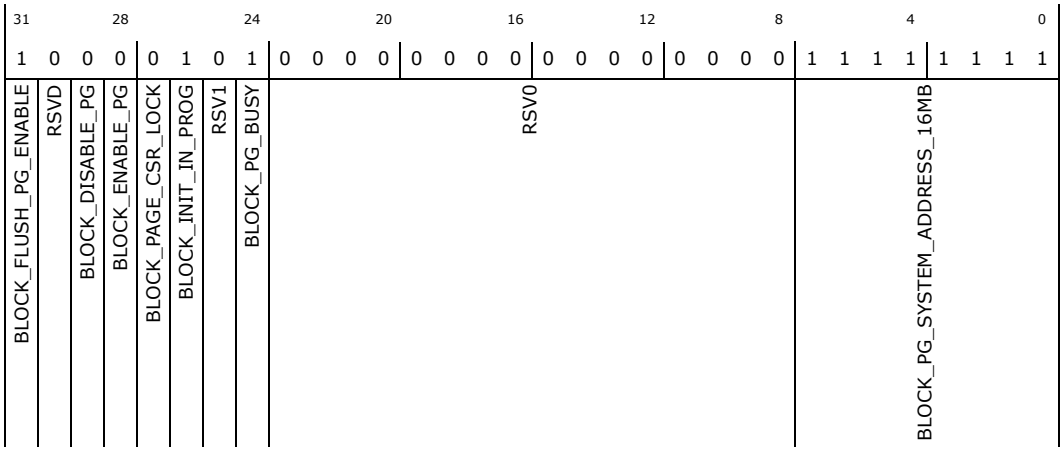
Access Method

Type: Message Bus Register (Size: 32 bits)

ESRAMPGCTRL_BLOCK: [Port: 0x05] + 82h

Op Codes:
10h - Read, 11h - Write

Default: 850000FFh



Bit Range	Default & Access	Description
31	1h RW/L	**Block Flush Page Enable (BLOCK_FLUSH_PG_ENABLE):** This field is used to enable or disable flushing of the block page to DRAM by the S3 entry firmware code. The block page may be flushed only by the S3 entry firmware code.

Bit Range	Default & Access	Description
30	0h RO	**Reserved (RSVD):** Reserved.
29	0h RW/HC	**Block Disable Page (BLOCK_DISABLE_PG):** When written with 0x1 disables block page decoding by eSRAM. This bit stays 0x1 until the block page has been disabled and ECC initialized. Note that this is a locking field, which locks on BLOCK_PAGE_CSR_LOCK=1 and ESRAMCTRL.eSRAM_GLOBAL_CSR_LOCK. This field should only be used by BIOS code.
28	0h RW/HC	**Block Enable Page (BLOCK_ENABLE_PG):** When written with 0x1 enables block page mapping of the eSRAM. When the block page is enabled, address mapping for all pages will be controlled by the block page address, instead of the 4KB page address fields. Cleared when the page flush/disable completes. Note that this is a locking field, locks on BLOCK_PAGE_CSR_LOCK=1 and ESRAMCTRL.eSRAM_GLOBAL_CSR_LOCK.
27	0h RW	**Block Page Register Lock (BLOCK_PAGE_CSR_LOCK):** When set to 1, the block page register (ESRAMPGCTRL_BLOCK) is locked. When locked, the block page may still be flushed to DRAM by the firmware S3 entry code by setting BLOCK_FLUSH_PG_ENABLE to 1.
26	1h RO/V	**Block Page Initialization in Progress (BLOCK_INIT_IN_PROG):** Reads 0x1 as long as the block page is being re-initialized following the disable. Note that while page is being flushed or re-initialized the eSRAM will block the access to the page stalling any requestor trying to access it. It also stays high until the ECC initialization completes after the reset.
25	0h RO	**Reserved (RSV1):** Reserved.
24	1h RO/V	**Block Page Busy (BLOCK_PG_BUSY):** Reads 0x1 when the block page is enabled and stays 0x1 until the block page has been flushed (if flush was to be performed) and reinitialized following disable. It also stays high until the ECC initialization completes after the reset.
23:8	0000h RO	**Reserved (RSV0):** Reserved.
7:0	FFh RW/L	**Block Page Base Address (BLOCK_PG_SYSTEM_ADDRESS_16MB):** Base address of the 512KB eSRAM block page (bits [31:24] of the system memory address). The eSRAM block page may only be placed on 16MB boundaries. Writes to this register will only update the contents when ESRAMPGCTRL_BLOCK.BLOCK_ENABLE_PG is 0. Note that the value in this register field is locked until the page has been disabled and is free.

12.7.4.39 eSRAM Correctable Error (ESRAMCERR)—Offset 83h

Provides status information for correctable ECC errors.

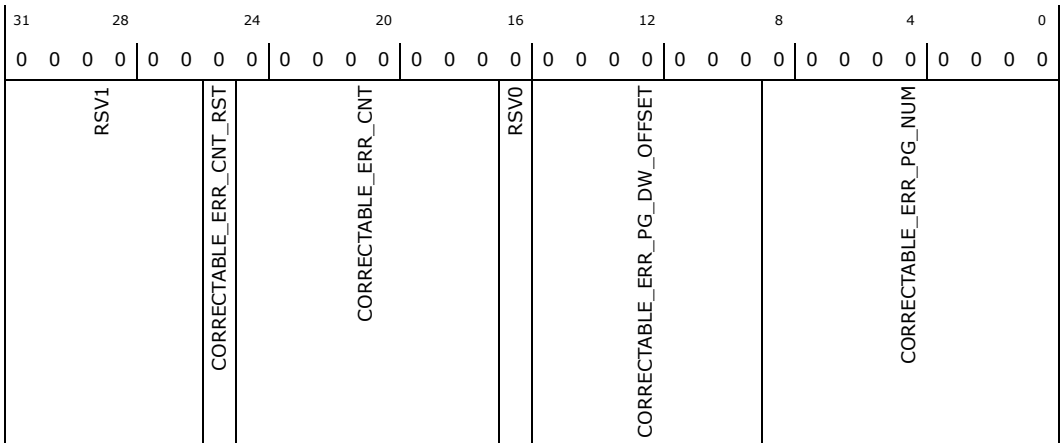
Access Method

Type: Message Bus Register (Size: 32 bits)

ESRAMCERR: [Port: 0x05] + 83h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:26	0h RO	**Reserved (RSV1):** Reserved.
25	0h RW/HC	**Correctable Error Counter Reset (CORRECTABLE_ERR_CNT_RST):** Resets the correctable ECC error counter.
24:17	0h RO/P	**Correctable Error Counter (CORRECTABLE_ERR_CNT):** Correctable ECC error count. Saturates at 8'hFF
16	0h RO	**Reserved (RSV0):** Reserved.
15:9	0h RO/P	**Correctable Error Page DW Offset (CORRECTABLE_ERR_PG_DW_OFFSET):** Page DW offset for the last Correctable ECC Error
8:0	0h RO/P	**Correctable Error Page Number (CORRECTABLE_ERR_PG_NUM):** Page number for the last Correctable ECC Error

12.7.4.40 eSRAM Uncorrectable Error (ESRAMUERR)—Offset 84h

Provides status information for the uncorrectable ECC error.

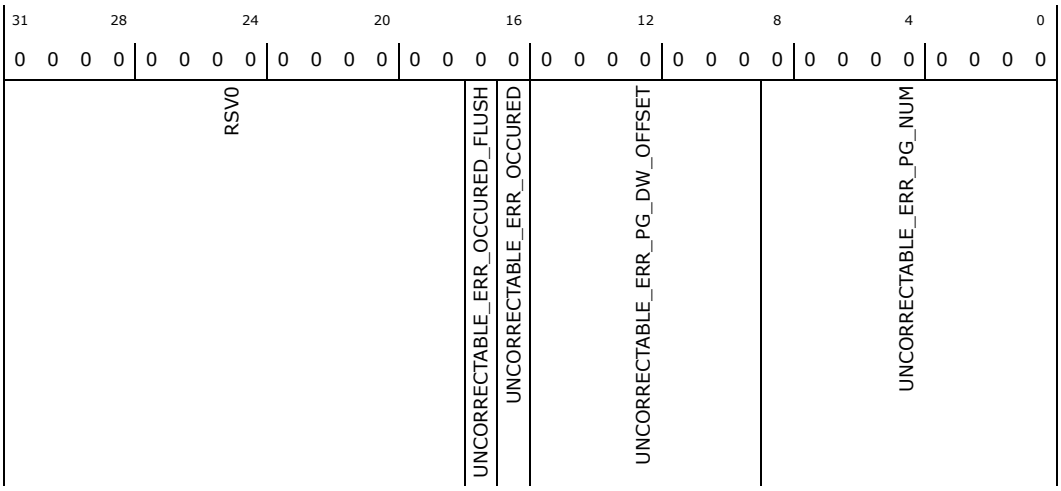
Access Method

Type: Message Bus Register (Size: 32 bits)

ESRAMUERR: [Port: 0x05] + 84h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:18	0000h RO	**Reserved (RSV0):** Reserved.
17	0h RW/1C/P	**Uncorrectable Error Occurred during Flush (UNCORRECTABLE_ERR_OCCURED_FLUSH):** Sticky register field which asserts when an uncorrectable ECC error has occurred during an eSRAM flush to DRAM. Set by hardware, cleared by software writing a 1.
16	0h RW/1C/P	**Uncorrectable Error Occurred (UNCORRECTABLE_ERR_OCCURED):** Sticky register field which asserts when an uncorrectable ECC error has occurred. Set by hardware, cleared by software writing a 1.
15:9	0h RO/P	**Uncorrectable Error Page DW Offset (UNCORRECTABLE_ERR_PG_DW_OFFSET):** Page DW offset for the uncorrectable ECC Error
8:0	0h RO/P	**Uncorrectable Error Page Number (UNCORRECTABLE_ERR_PG_NUM):** Page number for the uncorrectable ECC Error

12.7.4.41 eSRAM ECC Error Syndrome (ESRAMSDROME)—Offset 88h

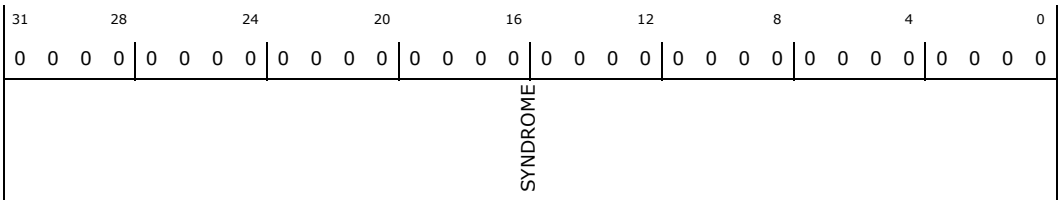
Access Method

Type: Message Bus Register (Size: 32 bits)

ESRAMSDROME: [Port: 0x05] + 88h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0h RO/P	**Syndrome (SYNDROME):** Syndrome for the last ECC (SECDED) error. For single bit errors, can be used to decode which bit of the eSRAM read data or ECC code was incorrect. Bits [7:0] relate to esram_ecc_out[7:0] and esram_data_out[63:0], Bits [15:8] relate to esram_ecc_out[15:8] and esram_data_out[127:64], Bits [23:16] relate to esram_ecc_out[23:16] and esram_data_out[191:128], Bits [31:24] relate to esram_ecc_out[31:24] and esram_data_out[255:192].

12.7.5 Memory Manager eSRAM (Port 0x05)

Table 73. Summary of Message Bus Registers—0x05

Offset	Register Name (Register Symbol)	Default Value
0h + [0-127]*4h	"eSRAM Page Control Register[0-127] (ESRAMPGCTRL[0-127])—Offset 0h, Count 128, Stride 4h" on page 228	850FFFFFh

12.7.5.1 eSRAM Page Control Register[0-127] (ESRAMPGCTRL[0-127])—Offset 0h, Count 128, Stride 4h

This register provides individual 4KB page control and status information.

Access Method

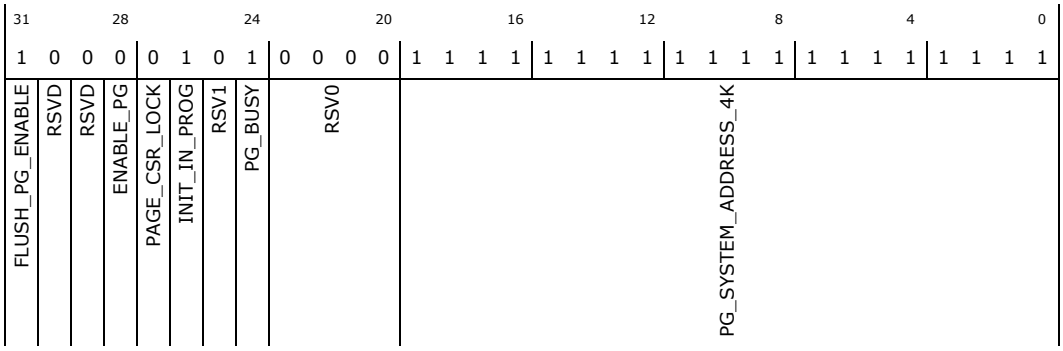
Type: Message Bus Register (Size: 32 bits)

Offset[0-127]: [Port: 0x05] + 0h + [0-127]*4h

Op Codes:

12h - Read, 13h - Write

Default: 850FFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31	1h RW/L	**Flush Page Enable (FLUSH_PG_ENABLE):** This field is used to enable or disable flushing of the eSRAM page to DRAM by the S3 entry firmware code. eSRAM pages may be flushed only by the S3 entry firmware code.
30	0h RO	**Reserved (RSVD):** Reserved.
29	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW/HC	**Enable Page (ENABLE_PG):** When written with 0x1 enables page decoding by eSRAM. Same effect will be achieved on all eSRAM page CSRs when the ESRAMCTRL. eSRAM_Enable_All is set. Cleared when the page flush/disable completes. Note that this is a locking field, locks on PAGE_CSR_LOCK.
27	0h RW	**Lock Page (PAGE_CSR_LOCK):** When set to 1, this 4KB page register (ESRAMPGCTRLX) for this page is locked. Locked pages may still be flushed to DRAM by firmware S3 entry code by setting FLUSH_PG_ENABLE to 1.
26	1h RO/V	**Initialisation In Progress (INIT_IN_PROG):** Reads 0x1 as long as the page is being re-initialized following the disable. Note that while page is being flushed or re-initialized the eSRAM will block the access to the page stalling any requestor trying to access it. It also stays high until the ECC initialization completes after the reset.
25	0h RO	**Reserved (RSV1):** Reserved.
24	1h RO/V	**Page Busy (PG_BUSY):** Reads 0x1 when the page is enabled and stays 0x1 until the page has been flushed (if flush was to be performed) and reinitialized following disable. It also stays high until the ECC initialization completes after the reset.
23:20	0h RO	**Reserved (RSV0):** Reserved.
19:0	1FFFFFh RW/L	**Page 4K Address (PG_SYSTEM_ADDRESS_4K):** 20b of base address (bits [31:12] of the system memory address) for the 4K page. Writes to this register field will only update the contents when ENABLE_PG is 0. Note that the value in this register field is locked until the page has been disabled and is free.

12.7.6 SoC Unit (Port 0x31)

Table 74. Summary of Message Bus Registers—0x31

Offset	Register Name (Register Symbol)	Default Value
34h	"Thermal Sensor Configuration 4 (SCU_TSCFG4_Config)—Offset 34h" on page 229	00057801h
50h	"Sticky Write Once (CFGSTICKY_W1)—Offset 50h" on page 230	00000000h
51h	"Sticky Read/Write (CFGSTICKY_RW)—Offset 51h" on page 230	00000000h
52h	"Non-Sticky Read/Write Once (CFGNONSTICKY_W1)—Offset 52h" on page 231	00000000h

12.7.6.1 Thermal Sensor Configuration 4 (SCU_TSCFG4_Config)—Offset 34h

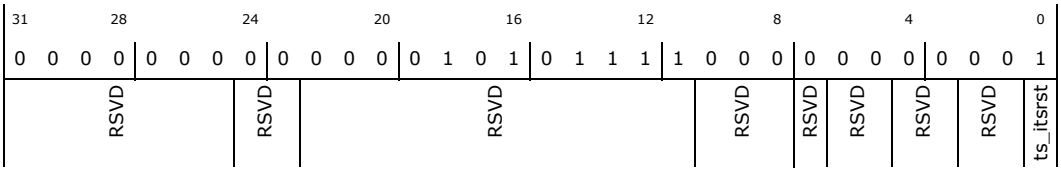
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x31] + 34h

Op Codes:
06h - Read, 07h - Write

Default: 00057801h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0h RO	**Reserved (RSVD):** Reserved.
24:23	0h RO	**Reserved (RSVD):** Reserved.
22:11	AFh RO	**Reserved (RSVD):** Reserved.
10:8	0h RO	**Reserved (RSVD):** Reserved.
7	0h RO	**Reserved (RSVD):** Reserved.
6:5	0h RO	**Reserved (RSVD):** Reserved.
4:3	0h RO	**Reserved (RSVD):** Reserved.
2:1	0h RO	**Reserved (RSVD):** Reserved.
0	1h RW	**Thermal Sensor Reset (ts_itsrst):** Resets all Thermal Sensor registers.

12.7.6.2 Sticky Write Once (CFGSTICKY_W1)—Offset 50h

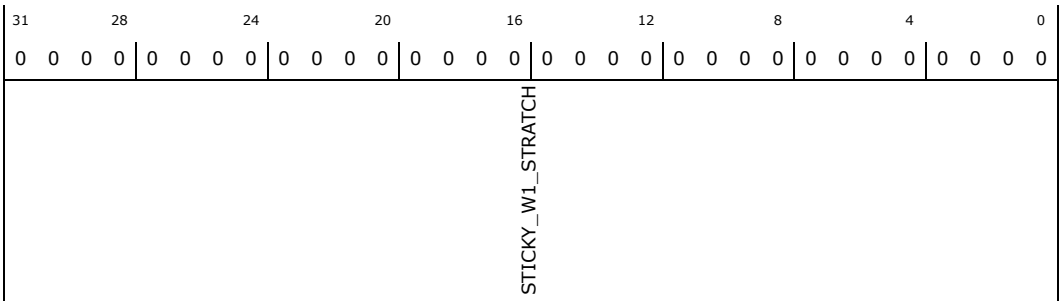
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x31] + 50h

Op Codes:
06h - Read, 07h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0000h RW/O	**Sticky Write Once Scratchpad (STICKY_W1_STRATCH):** Assigned by Software, write once requires a S0 exit to clear

12.7.6.3 Sticky Read/Write (CFGSTICKY_RW)—Offset 51h

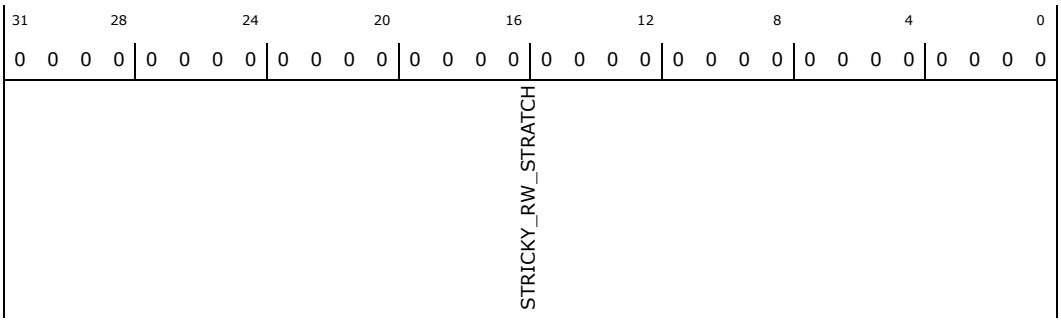
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x31] + 51h

Op Codes:
06h - Read, 07h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0000h RW/P	**Sticky Read/Write Scratchpad (STRICKY_RW_STRATCH):** Assigned by Software, reset by a S0 exit

12.7.6.4 Non-Sticky Read/Write Once (CFGNONSTICKY_W1)—Offset 52h

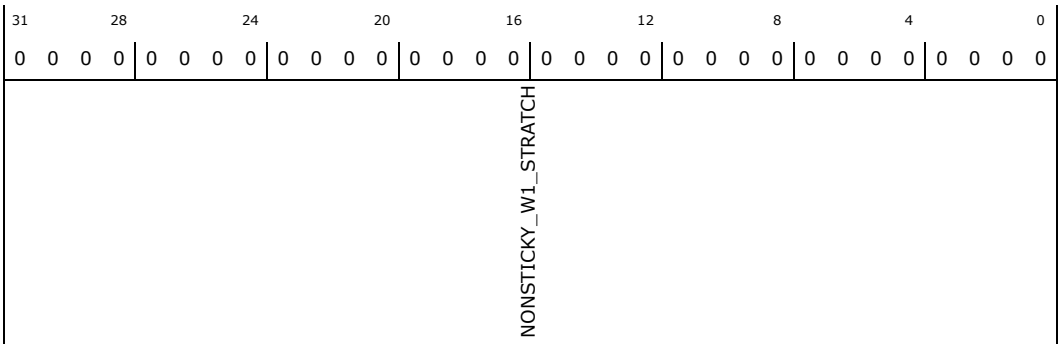
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x31] + 52h

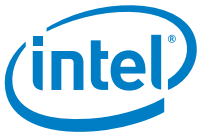
Op Codes:
06h - Read, 07h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0000h RW/O	**Non-Sticky Write Once Scratchpad (NONSTICKY_W1_STRATCH):** Assigned by Software, reset by a warm reset while in S0

§ §



13.0 System Memory Controller

The system memory controller supports DDR3 protocol with one 16-bit wide data channel and up to 2 ranks of memory, allowing for population of up to 2Gbyte of system memory using 1, 2 or 4 Gbit standard DDR3 devices. It is capable of data rates up to 800 MT/s.

13.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 75. Memory Signals (Sheet 1 of 2)

Signal Name	Direction Type	Description
DDR3_CK[1:0] DDR3_CKB[1:0]	O DDR3	DRAM Differential Clock Pair: (1 pair per Rank) The differential clock pair is used to latch the command into DRAM. Each pair corresponds to a rank on the DRAM side.
DDR3_CSB[1:0]	O DDR3	Chip Select: (1 per Rank). Used to qualify the command on the command bus for a particular rank.
DDR3_CKE[1:0]	O DDR3	Clock Enable: (power management - 1 per Rank) It is used during DRAM power up/power down and self refresh.
DDR3_MA[15:0]	O DDR3	Memory Address: Multiplexed Memory address bus (Row, Column) for writing data to memory and reading data from memory. These signals follow common clock protocol w.r.t. DDR3_CK, DDR3_CKB pairs.
DDR3_BS[2:0]	O DDR3	Bank Select: These signals define which banks are selected within each DRAM rank.
DDR3_RASB	O DDR3	Row Address Select: Used with DDR3_CASB and DDR3_WEB (along with DDR3_CSB) to define the DRAM Commands.
DDR3_CASB	O DDR3	Column Address Select: Used with DDR3_RASB and DDR3_WEB (along with DDR3_CSB) to define the DRAM Commands.
DDR3_WEB	O DDR3	Write Enable Control Signal: Used with DDR3_WEB and DDR3_CASB (along with control signal, DDR3_CSB) to define the DRAM Commands.
DDR3_DQ[15:0]	I/O DDR3	Bidirectional Data Lines

Table 75. Memory Signals (Sheet 2 of 2)

Signal Name	Direction Type	Description
DDR_DM[1:0]	O DDR3	Data Mask: DM is an output mask signal for write data. Output data is masked when DM is sampled HIGH coincident with that output data during a Write access. DM is sampled on both edges of DQS.
DDR3_DQS[1:0] DDR3_DQSB[1:0]	I/O DDR3	Data Strobes: DDR3_DQSB[1:0] and its complement signal group make up a differential strobe pair for each 8 data bits - DQ. The data is captured at the crossing point of DDR3_DQS[1:0] and its DDR3_DQSB[1:0] during read and write transactions. For Read, the Strobe crossover and data are edge aligned, whereas in the Write command, the strobe crossing is in the centre of the data window.
DDR3_ODT[1:0]	O DDR3	ODT signal going to DRAM in order to turn ON the DRAM ODT during Write.
DDR3_ODTPU	O Analog	This signal must be terminated to VSS on board (refer to the *Platform Design Guide* for resistor value). This external resistor termination scheme is used for Resistor compensation of DRAM ODT strength.
DDR3_DQPU	O Analog	This signal must be terminated to VSS on board (refer to the *Platform Design Guide* for resistor value). This external resistor termination scheme is used for Resistor compensation of DQ buffers
DDR3_CMDPU	O Analog	This signal must be terminated to VSS on board (refer to the *Platform Design Guide* for resistor value). This external resistor termination scheme is used for Resistor compensation of CMD buffers.
DDR3_VREF	I Analog	DRAM Interface Reference Voltage: This signal voltage level is used for qualifying logical levels on the DQ bits on reads. The Memory interface can also use internally generated reference voltage to qualify the crossing point between logical levels on Data bits. Internal Vref is a default setting for Memory Interface and this interface signal and can be tied to VSS on board.
DDR3_ISYSPWRGOOD	I Asynchronous CMOS	This signal indicates the status of the DRAM Core power supply.
DDR3_IDRAM_PWROK	I Asynchronous CMOS	This signal indicates the status of the DRAM S3 power supply. Used primarily in the DRAM PHY to determine ACPI S3 power state.
DDR3_DRAMRSTB	O	This signal is used to reset DRAM devices.

13.2 Features

13.2.1 System Memory Technology Supported

The system memory controller supports the following DDR3 Data Transfer Rates and DRAM Device Technologies:

- DDR3 Data Transfer Rate: 800 MT/s
- DDR3 (1.5V DRAM interface I/Os)
- DDR3 x8 memory modules

Note: x8 means that each DRAM component has 8 data lines. Standard 1Gbit, 2Gbit, and 4Gbit technologies and addressing are supported for x8 devices.

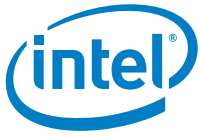


Table 76. Supported DDR3 DRAM Devices

DRAM Density	Data Width	Banks	Bank Address	Row Address	Column Address	Page Size
1 Gbit	x8	8	BA[2:0]	A[13:0]	A[9:0]	1 Kbyte
2 Gbit	x8	8	BA[2:0]	A[14:0]	A[9:0]	1 Kbyte
4 Gbit	x8	8	BA[2:0]	A[15:0]	A[9:0]	1 Kbyte

Table 77. Supported DDR3 Memory Configurations

DRAM Chip Density	DRAM Device Width	# of DRAM Devices per Rank	Rank Size	# of Ranks	Total Memory Size
1 Gbit	x8	2	256 Mbyte	1	256 Mbyte
1 Gbit	x8	2	256 Mbyte	2	512 Mbyte
2 Gbit	x8	2	512 Mbyte	1	512 Mbyte
2 Gbit	x8	2	512 Mbyte	2	1 Gbyte
4 Gbit	x8	2	1 Gbyte	1	1 Gbyte
4 Gbit	x8	2	1 Gbyte	2	2 Gbyte

13.2.2 Rules for Populating Memory Down Ranks

The devices density and width for both ranks must be the same. Rank0 must be always populated and Rank1 is optional.

13.2.3 DRAM Error Detection & Correction (EDC)

For high reliability applications the system memory controller supports inclusion of Error Correction Codes (ECC) in DRAM transactions. The algorithm used allows for on-the-fly correction of single bit errors and detection of double bit errors. In ECC mode 12.5% of the total system memory is allocated for EDC data storage.

13.2.4 DRAM Data Scrambling

The Memory Controller supports data scrambling. This feature helps in lowering the MTBF (Mean Time Between Failures) by reducing the probability of occurrence of the specific bit patterns on the DRAM I/O interface that could cause unpredictable behavior at the platform design stage, i.e., signal integrity issues.

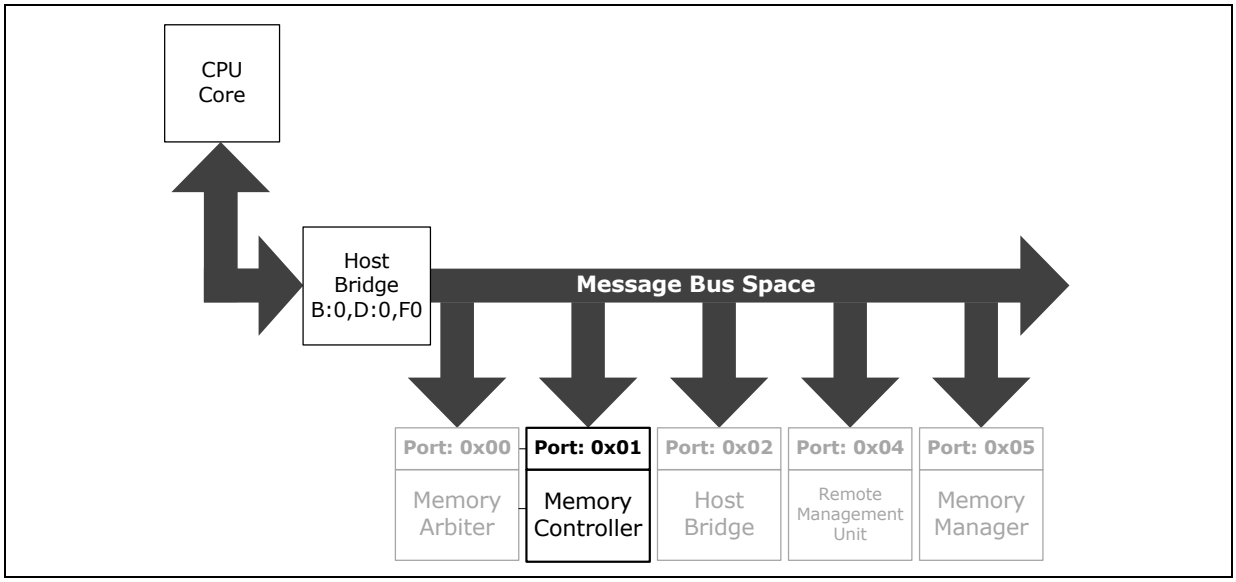
13.2.5 Power Management

The System Memory Controller Power Management features are detailed in Section 8.4.

13.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 25. Register Map



13.4 Message Bus Registers

Table 78. Summary of Message Bus Registers—0x01

Offset	Register Name (Register Symbol)	Default Value
0h	"DRAM Rank Population (DRP)—Offset 0h" on page 237	00000000h
1h	"DRAM Timing Register 0 (DTR0)—Offset 1h" on page 238	43001110h
2h	"DRAM Timing Register 1 (DTR1)—Offset 2h" on page 240	02690320h
3h	"DRAM Timing Register 2 (DTR2)—Offset 3h" on page 242	00040504h
4h	"DRAM Timing Register 3 (DTR3)—Offset 4h" on page 243	06406205h
5h	"DRAM Timing Register 4 (DTR4)—Offset 5h" on page 244	00000022h
6h	"DRAM Power Management Control 0 (DPMC0)—Offset 6h" on page 245	03000000h
8h	"DRAM Refresh Control (DRFC)—Offset 8h" on page 247	00012CA7h
9h	"DRAM Scheduler Control (DSCH)—Offset 9h" on page 248	00071108h
Ah	"DRAM Calibration Control (DCAL)—Offset Ah" on page 249	00001300h
Bh	"DRAM Reset Management Control (DRMC)—Offset Bh" on page 250	00000000h
Ch	"Power Management Status (PMSTS)—Offset Ch" on page 251	00000000h
Fh	"DRAM Control Operation (DCO)—Offset Fh" on page 252	00000000h
4Ah	"Sticky Scratchpad 0 (SSKPD0)—Offset 4Ah" on page 252	00000000h
4Bh	"Sticky Scratchpad 1 (SSKPD1)—Offset 4Bh" on page 253	00000000h
60h	"DRAM ECC Control Register (DECCCTRL)—Offset 60h" on page 253	00000000h
61h	"DRAM ECC Status (DECCSTAT)—Offset 61h" on page 254	00000000h
62h	"DRAM ECC Single Bit Error Count (DECCSBECNT)—Offset 62h" on page 254	00000000h
68h	"DRAM Single Bit ECC Error Captured Address (DECCSBECA)—Offset 68h" on page 255	00000000h
69h	"DRAM Single Bit ECC Error Captured Syndrome (DECCSBECS)—Offset 69h" on page 256	00000000h

Table 78. Summary of Message Bus Registers—0x01 (Continued)

Offset	Register Name (Register Symbol)	Default Value
6Ah	"DRAM Double Bit ECC Error Captured Address (DECCDBECA)—Offset 6Ah" on page 256	00000000h
6Bh	"DRAM Double Bit ECC Error Captured Syndrome (DECCDBECS)—Offset 6Bh" on page 257	00000000h
70h	"Memory Controller Fuse Status (DFUSESTAT)—Offset 70h" on page 257	00000000h
80h	"Scrambler Seed (DSCRMSEED)—Offset 80h" on page 258	00000000h

13.4.1 DRAM Rank Population (DRP)—Offset 0h

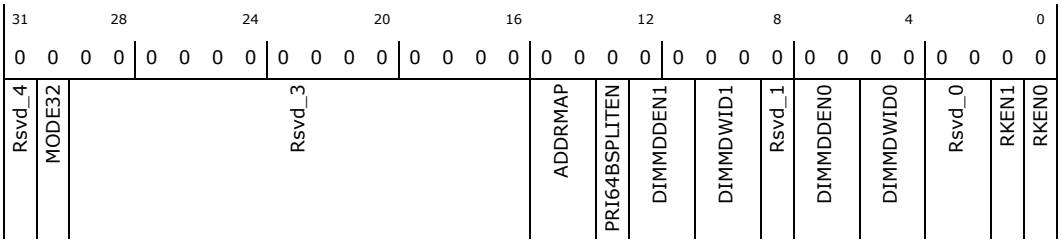
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 0h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RO	**Rsvd_4:** Reserved
30	0h RW/P/L	**16-bit/32-bit Mode Select (MODE32):** 0 - Selects 16-bit DRAM Data Interface. 1 - Selects 32-bit DRAM Data Interface.
29:16	0000h RO	**Rsvd_3:** Reserved
15:14	0h RW/P/L	**Address Map Select (ADDRMAP):** See Address Mapping section for full description. 0 Map 0 1 Map 1 2 Map 2 Note: The address map select should be set the same for both the Memory Controller and the Memory Manager.
13	0h RW	**64B Split Enable (PRI64BSPLITEN):** Setting this bit to '1' enables logic to split 64B PRI transactions to two 32B transactions. This bit must be set if ECC mode is enabled.
12:11	0h RW/P/L	**Rank 1 Device Density (DIMMDDEN1):** This sets the density of the DRAM devices populated in Rank 1. 00 1Gbit 01 2Gbit 10 4Gbit 11 Reserved
10:9	0h RW/P/L	**Rank 1 Device Width (DIMMDWID1):** Indicates the width of the DRAM devices populated in Rank 1. 00 x8 01 x16 10 Reserved 11 Reserved

Bit Range	Default & Access	Field Name (ID): Description
8	0h RO	**Rsvd_1:** Reserved
7:6	0h RW/P/L	**Rank 0 Device Density (DIMMDDEN0):** This sets the density of the DRAM devices populated in Rank 0. 00 1Gbit 01 2Gbit 10 4Gbit 11 Reserved
5:4	0h RW/P/L	**Rank 0 Device Width (DIMMDWID0):** Indicates the width of the DRAM devices populated in Rank 0. 00 x8 01 x16 10 Reserved 11 Reserved
3:2	0h RO	**Rsvd_0:** Reserved
1	0h RW/P/L	**Rank Enable 1 (RKEN1):** Should be set to 1 when device has 2 ranks to enable the use of second rank. Otherwise, must be set to 0.
0	0h RW/P/L	**Rank Enable 0 (RKEN0):** Should be set to 1 when Rank 0 is populated to enable the use of this rank. Otherwise, must be set to 0.

13.4.2 DRAM Timing Register 0 (DTR0)—Offset 1h

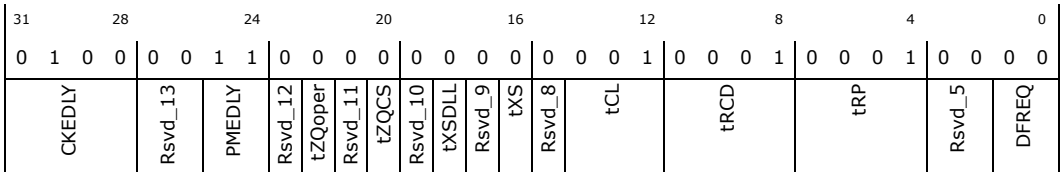
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 1h

Op Codes:
10h - Read, 11h - Write

Default: 43001110h



Bit Range	Default & Access	Field Name (ID): Description
31:28	4h RW	**Clock Valid To Self-Refresh Exit Delay (CKEDLY):** Additional delay between CK/CKB start and SRX command. This delay is needed for clock to stabilize to meet JEDEC requirements. Delay is CKEDLY multiples of 256 DRAM Clocks. 0ns to 9,600ns (DDR3-800) 0ns to 7,200ns (Future DDR3-1066)
27:26	0h RO	**Rsvd_13:** Reserved
25:24	3h RW	**Power Mode Entry Delay (PMEDLY):** The delay, in DRAM clocks, between SR Entry command and Power-Mode message to DDRIO. 0h - 6 DRAM Clocks. 1h - 8 DRAM Clocks. 2h - 10 DRAM Clocks. 3h - 12 DRAM Clocks.
23	0h RO	**Rsvd_12:** Reserved

Bit Range	Default & Access	Field Name (ID): Description
22	0h RW	**ZQCal Long Delay (tZQoper):** The delay, in DRAM clocks, between ZQC-Long command to any command. Note: ZQCL command during DRAM Init flow requires longer latency which is controlled be BIOS. 0h - 256 DRAM Clocks. 1h - 384 DRAM Clocks. Note: This field defines the ZQ Calibration Long delay during normal operation. It is not the same as tZQinit, which uses the same ZQCL command but the delay is longer. tZQinit applies only during power-on initialization of the DRAM devices, and tZQoper applies during normal operation. BIOS executes the DRAM initialization sequence, so it has to ensure tZQinit is met, and not the Memory Controller.
21	0h RO	**Rsvd_11:** Reserved
20	0h RW	**ZQCal Short Delay (tZQCS):** The delay, in DRAM clocks, between a ZQC-Short command to any command. 0h - 64 DRAM Clocks. 1h - 96 DRAM Clocks.
19	0h RO	**Rsvd_10:** Reserved
18	0h RW	**Self-Refresh Exit To DLL Delay (tXSDLL):** The delay, in DRAM clocks, between SRX command to any command requiring locked DLL. Only ZQCL can be sent before tXSDLL is done. 0h - tXS + 256 DRAM Clocks. 1h - tXS + 384 DRAM Clocks.
17	0h RO	**Rsvd_9:** Reserved
16	0h RW	**Self-Refresh Exit Delay (tXS):** The delay, in DRAM clocks, between SRX command to command not requiring locked DLL. The Memory Controller can send a ZQCL command after tXS. JEDEC defines MAX(5CK, tRFC(min)+10ns) so both values take safety margin. 0h - 256 DRAM Clocks. 1h - 384 DRAM Clocks.
15	0h RO	**Rsvd_8:** Reserved
14:12	1h RW	**CAS Latency (tCL):** Specifies the delay, in DRAM clocks, between the issue of a RD command and the return of valid data on the DQ bus. 0h - 5 DRAM Clocks (DDR3-800) 1h - 6 DRAM Clocks (DDR3-800, Future DDR3-1066) 2h - 7 DRAM Clocks (Future DDR3-1066) 3h - 8 DRAM Clocks (Future DDR3-1066) 4h - 9 DRAM Clocks (Future DDR3-1333) 5h - 10 DRAM Clocks (Future DDR3-1333) 6h - 11 DRAM Clocks (Future DDR3-1600) 7h - Reserved
11:8	1h RW	**Activate (RAS) to CAS Delay (tRCD):** Specifies the delay, in DRAM clocks, between an ACT command and a RD/WR command to the same bank. 0h - 5 DRAM Clocks (DDR3-800) 1h - 6 DRAM Clocks (DDR3-800, Future DDR3-1066) 2h - 7 DRAM Clocks (Future DDR3-1066, 1333) 3h - 8 DRAM Clocks (Future DDR3-1066, 1333) 4h - 9 DRAM Clocks (Future DDR3-1333) 5h - 10 DRAM Clocks (Future DDR3-1333) 6h - 11 DRAM Clocks (Future DDR3-1600) 7h - Reserved
7:4	1h RW	**Precharge to Activate Delay (tRP):** Specifies the delay, in DRAM clocks, between a PRE command and an ACT command to the same bank. 0h - 5 DRAM Clocks (DDR3-800) 1h - 6 DRAM Clocks (DDR3-800, Future DDR3-1066) 2h - 7 DRAM Clocks (Future DDR3-1066, 1333) 3h - 8 DRAM Clocks (Future DDR3-1066, 1333) 4h - 9 DRAM Clocks (Future DDR3-1333) 5h - 10 DRAM Clocks (Future DDR3-1333) 6h - 11 DRAM Clocks (Future DDR3-1600) 7h - Reserved

Bit Range	Default & Access	Field Name (ID): Description
3:2	0h RO	**Rsvd_5:** Reserved
1:0	0h RW	**DRAM Frequency (DFREQ):** Specifies the DDR3 frequency used by the Memory Controller for computing proper cycle to cycle timings. Note this configuration has no impact on the actual DRAM clock. 0h - DDR3-800 1h - Future DDR3-1066 2h - Future DDR3-1333 3h - Reserved Note: This configuration has no impact on the actual DRAM clock frequency.

13.4.3 DRAM Timing Register 1 (DTR1)—Offset 2h

Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 2h

Op Codes:
10h - Read, 11h - Write

Default: 02690320h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
Rsvd_18	tRTP			Rsvd_17	tRRD			tRAS				tFAW				Rsvd_16	tCCD			tWTP				Rsvd_15	tCMD			Rsvd_14	tWCL		

Bit Range	Default & Access	Field Name (ID): Description
31	0h RO	**Rsvd_18:** Reserved
30:28	0h RW	**Read to Precharge Delay (tRTP):** The minimal delay between RD command and PRE command to same bank. 001 - 4 DRAM Clocks (DDR3-800 Future DDR3-1066) 010 - 5 DRAM Clocks (Future DDR3-1333) 011 - 6 DRAM Clocks 100 - 7 DRAM Clocks
27:26	0h RO	**Rsvd_17:** Reserved
25:24	2h RW	**Row Activation to Row Activation Delay (tRRD):** The minimal time interval between 2 ACT commands to any bank in the same DRAM device. Limits peak current profile. 00 - 4 DRAM Clocks (1KB page DDR3-800, Future DDR3-1066), (2KB page DDR3-800), (Future 1KB page DDR3-1333) 01 - 5 DRAM Clocks (Future 2KB page DDR3-1333) 10 - 6 DRAM Clocks (Future 2KB page DDR3-1066) 11 - 7 DRAM Clocks Note: This timing parameter applies to both Ranks, so set it based on the rank with the large tRRD value.

Bit Range	Default & Access	Field Name (ID): Description
23:20	6h RW	**Row Activation Period (tRAS):** The minimal delay, in DRAM clocks, between ACT command and PRE command to same bank. At least equal to tRCD + tCWL + tCCD + tWR 0h -14 DRAM Clocks. 1h -15 DRAM Clocks (DDR3-800) 2h -16 DRAM Clocks. 3h -17 DRAM Clocks. 4h -18 DRAM Clocks. 5h -19 DRAM Clocks. 6h -20 DRAM Clocks (Future DDR3-1066) 7h -21 DRAM Clocks. 8h -22 DRAM Clocks. 9h -23 DRAM Clocks. Ah -24 DRAM Clocks (Future DDR3-1333) Others - Reserved
19:16	9h RW	**Four Bank Activation Window (tFAW):** A rolling time-frame, in which a maximum of 4 ACT commands (per rank) can be sent. Limits peak current profile. 0h - Reserved. 1h - Reserved. 2h - 14 DRAM Clocks. 3h - 16 DRAM Clocks (1KB page DDR3-800). 4h - 18 DRAM Clocks. 5h - 20 DRAM Clocks (2KB page DDR3-800, Future 1KB page DDR3-1066, 1333). 6h - 22 DRAM Clocks. 7h - 24 DRAM Clocks (Future 1KB page DDR3-1600). 8h - 26 DRAM Clocks. 9h - 28 DRAM Clocks (Future 2KB page DDR3-1066). Ah - 30 DRAM Clocks (Future 2KB page DDR3-1333). Bh - 32 DRAM Clocks Ch - Reserved. Dh - Reserved. Eh - Reserved. Fh - Reserved. Note: This timing parameter applies to both Ranks, so set it based on the rank with the large tFAW value.
15:14	0h RO	**Rsvd_16:** Reserved
13:12	0h RW	**CAS to CAS delay (tCCD):** The minimum delay, in DRAM clocks, between 2 RD/WR commands. 0h - 4 DRAM Clocks. Functional mode. (DDR3-800, 1066, 1333, 1600). 1h - 12 DRAM Clocks. DFX stretch mode (x2). 2h - 18 DRAM Clocks. DFX stretch mode (x4). 3h - Reserved
11:8	3h RW	**Write To Prechange Delay (tWTP):** The minimum delay, in DRAM clocks, between a WR command and a PRE command to the same bank. Value should be computed as 4 + tWCL + tWR. 1h - 15 DRAM Clocks DDR3-800). 2h - 16 DRAM Clocks. 3h - 17 DRAM Clocks. 4h - 18 DRAM Clocks Future DDR3-1066). 5h - 19 DRAM Clocks. 6h - 20 DRAM Clocks. 7h - 21 DRAM Clocks Future DDR3-1333). 8h - 22 DRAM Clocks. Others - Reserved Note: This is not a JEDEC timing parameter. It is derived from other JEDEC timing parameters.
7:6	0h RO	**Rsvd_15:** Reserved
5:4	2h RW	**Command Transport Duration (tCMD):** The time period, in DRAM clocks, that a command occupies the DRAM command bus. 1N is the DDR3 basic requirement. 2N and 3N are extended modes for board signal-integrity. 0h - 1 DRAM Clock (1N). 1h - 2 DRAM Clocks (2N). 2h - 3 DRAM Clocks (3N). Note: This is a board design timing parameter and not part of JEDEC spec.

Bit Range	Default & Access	Field Name (ID): Description
3	0h RO	**Rsvd_14:** Reserved
2:0	0h RW	**CAS Write Latency (tWCL):** The delay, in DRAM clocks, between the internal write command and the availability of the first bit of DRAM input data. 0h - 5 DRAM Clocks (DDR3-800) 1h - 6 DRAM Clocks (Future DDR3-1066) 2h - 7 DRAM Clocks (Future DDR3-1333) 3h - 8 DRAM Clocks

13.4.4 DRAM Timing Register 2 (DTR2)—Offset 3h

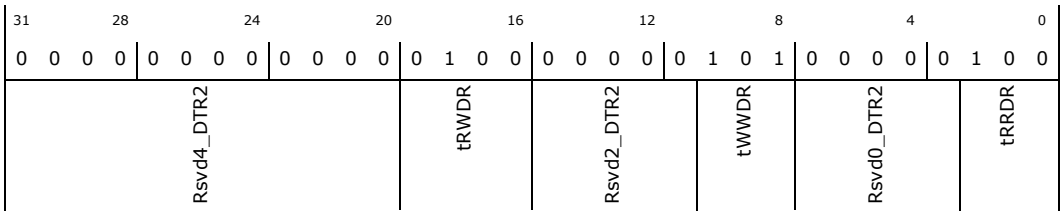
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 3h

Op Codes:
10h - Read, 11h - Write

Default: 00040504h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	**Rsvd4_DTR2:** Reserved
19:16	4h RW	**Read to Write Delay (tRWDR):** Read to Write DQ delay, different ranks. 1h - 6 DRAM Clocks. 2h - 7 DRAM Clocks. 3h - 8 DRAM Clocks. 4h - 9 DRAM Clocks. Note: This is a board design timing parameter and not part of JEDEC spec.
15:11	00h RO	**Rsvd2_DTR2:** Reserved
10:8	5h RW	**Write to Write Delay (tWWDR):** Write to Write DQ delay, different ranks. 0h - Reserved 1h - Reserved 2h - 6 DRAM Clocks. 3h - 7 DRAM Clocks. 4h - 8 DRAM Clocks. 5h - 9 DRAM Clocks. 6h - Reserved 7h - Reserved Note: This is a board design timing parameter and not part of JEDEC spec.
7:3	00h RO	**Rsvd0_DTR2:** Reserved

Bit Range	Default & Access	Field Name (ID): Description
2:0	4h RW	**Read to Read Delay (tRRDR):** Read to Read DQ delay, different ranks. 0h - Reserved 1h - 6 DRAM Clocks. 2h - 7 DRAM Clocks. 3h - 8 DRAM Clocks. 4h - 9 DRAM Clocks. Others - Reserved Note: This is a board design timing parameter and not part of JEDEC spec.

13.4.5 DRAM Timing Register 3 (DTR3)—Offset 4h

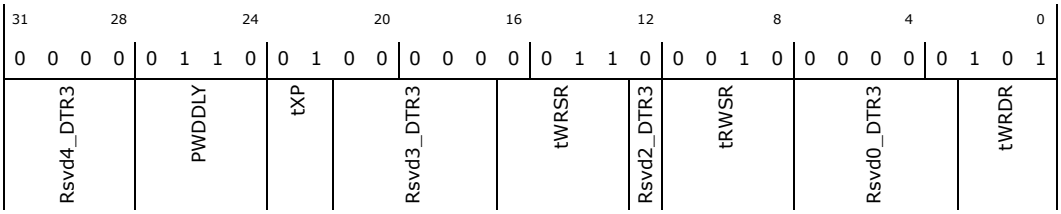
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 4h

Op Codes:
10h - Read, 11h - Write

Default: 06406205h



Bit Range	Default & Access	Field Name (ID): Description
31:28	0h RO	**Rsvd4_DTR3:** Reserved
27:24	6h RW	**RD/WR command to Power-down Delay (PWDDLY):** Non-JEDEC delay for performance enhancement. Delay = PWDDLY x 4 DRAM Clocks.
23:22	1h RW	**CKR to Command Delay (tXP):** Delay from CKE asserted high to any DRAM command 0h - 2 DRAM Clocks (DDR3-800 2N). 1h - 3 DRAM Clocks (DDR3-800 1N, Future DDR3-1066 2N). 2h - 4 DRAM Clocks (Future DDR3-1066 1N). 3h - 5 DRAM Clocks.
21:17	0h RO	**Rsvd3_DTR3:** Reserved
16:13	3h RW	**Write to Read Command Delay (tWRSR):** Write to Read same rank command delay. Should be set to 4 + tWCL + tWTR 2h - 13 DRAM Clocks (DDR3-800). 3h - 14 DRAM Clocks (Future DDR3-1066). 4h - 15 DRAM Clocks. 5h - 16 DRAM Clocks. 6h - 17 DRAM Clocks. 7h - 18 DRAM Clocks. 8h - 19 DRAM Clocks. 9h - 20 DRAM Clocks. Others - Reserved Note: This is a board design timing parameter and not part of JEDEC spec.
12	0h RO	**Rsvd2_DTR3:** Reserved

Bit Range	Default & Access	Field Name (ID): Description
11:8	2h RW	**Read to Write Command Delay (tRWSR):** Read to Write same rank command delay. Should be set to tCL - tWCL + 6 + board delay if needed. 0h - 6 DRAM Clocks. 1h - 7 DRAM Clocks. 2h - 8 DRAM Clocks. 3h - 9 DRAM Clocks. 4h - 10 DRAM Clocks. 5h - 11 DRAM Clocks. Others - Reserved Note: This is a board design timing parameter and not part of JEDEC spec.
7:3	0h RO	**Rsvd0_DTR3:** Reserved
2:0	5h RW	**Write to Read Delay (tWRDR):** Write to Read DQ delay, different ranks. 0h - Reserved 1h - 6 DRAM Clocks. 2h - 7 DRAM Clocks. 3h - 8 DRAM Clocks. 4h - 9 DRAM Clocks. 5h - 10 DRAM Clocks. Others - Reserved Note: This is a board design timing parameter and not part of JEDEC spec.

13.4.6 DRAM Timing Register 4 (DTR4)—Offset 5h

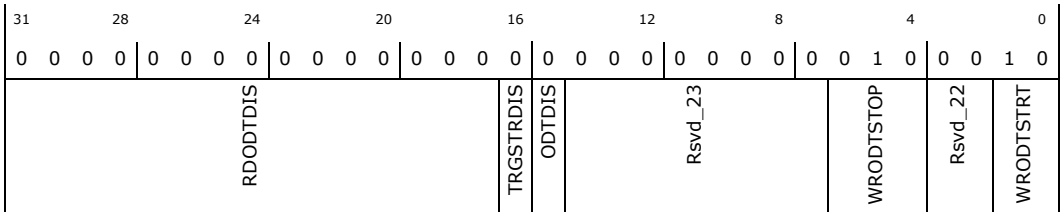
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 5h

Op Codes:
10h - Read, 11h - Write

Default: 00000022h



Bit Range	Default & Access	Field Name (ID): Description
31:17	0h RW	**Rsvr_24 (RDODTDIS):** Reserved
16	0h RW	**Disable Write ODT Stretching (TRGSTRDIS):** Write target rank is not stretched. When set, stretched ODT as defined above is not applied to the write target rank and ODT command is asserted for 6 DRAM clocks. Should not be used when ODT is pulled-in. Note: This bit should be set to 0 for normal operation. Note: This bit should not be set to 1 when ODT is configured to assert earlier than the Write command.
15	0h RW	**ODT Disable (ODTDIS):** 0 - ODT is enabled. 1 - ODT is disabled
14:7	0h RO	**Rsvd_23:** Reserved

Bit Range	Default & Access	Field Name (ID): Description
6:4	2h RW	**Write command to ODT de-assert delay (WRODTSTOP):** For 1N Command Mode 0h WR+6 1h WR+7 2h WR+8 3h N/A 4h N/A Others Reserved For 2N Command Mode 0h N/A 1h WR+6 2h WR+7 3h WR+8 4h N/A Others Reserved For 3N Command Mode 0h N/A 1h N/A 2h WR+6 3h WR+7 4h WR+8 Others Reserved
3:2	0h RO	**Rsvd_22:** Reserved
1:0	2h RW	**Write command to ODT assert delay (WRODTSTRT):** JEDEC requires ODT to be asserted on the same clock with the WR command. The Memory Controller allows to pull-in by 1 clock in 2N mode and by 1-2 clocks in 3N mode. For most DIMM configurations, this register should be programmed to same value as tCMD. A value of tCMD - ODT_PULLIN can be used according to the table below which shows the ODT command assertion with respect to the WR command assertion. For 1N Command Mode 0h WR 1h N/A 2h N/A 3h Reserved For 2N Command Mode 0h WR-1 1h WR 2h N/A 3h Reserved For 3N Command Mode 0h WR-2 1h WR-1 2h WR 3h Reserved

13.4.7 DRAM Power Management Control 0 (DPMC0)—Offset 6h

Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 6h

Op Codes:
10h - Read, 11h - Write

Default: 03000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rsvd_31	ENPHYCLKGATE	REUTCLKGTDIS	Rsvd27		DISPWRDN	CLKGTDIS	DYNSREN	Rsvd_30	PREAPWDEN	PCLSWKOK	Rsvd_29	PCLSTO		Rsvd_28		RSVD				SREDLY												

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	**Rsvd_31:** Reserved
29	0h RW	**Enable PHY Clock Gate Disable During SR (ENPHYCLKGATE):** When set to 1, the Memory Controller will turn off the 1x and 2x clock trees to the DDRIO PHY during Self Refresh. The Memory Controller will re-enable the clocks upon Self Refresh exit.
28	0h RW	**MTE Clock Gate Disable (REUTCLKGTDIS):** 0h MTE clock is gated when DCO.PMICTL is set to 0. 1h MTE clock is ungated, overriding the DCO.PMICTL config bit. Note: The DCO.CPGCLOCK bit overrides this bit.
27:26	0h RO	**Rsvd27:** Reserved
25	1h RW	**Disable Power Down (DISPWRDN):** Setting this bit to 1 will block CKE high-)low transitions. May be used by BIOS during init flow and should be set to 0 for functional mode. 0 - The Memory Controller dynamically controls the CKE pins to place the DRAM device in power down mode. 1 - The Memory Controller constantly drives the CKE pins high.
24	1h RW	**Clock Gating Disabled (CLKGTDIS):** Setting this bit to 0 allows a large number of internal Memory Controller clocks to be gated when there is no activity in order to save power. When set to 1, internal clock-gating is disabled. 0 - Enable. 1 - Disable. Note: This bit should be set to 0 for normal operation.
23	0h RW	**Dynamic Self-Refresh Enable (DYNSREN):** Setting this bit to 1, enables automatic SR command to DRAM and PM message to DDRIO when the PRI bus is idle, all pending requests have been served and the and PRI status is less than 2, SREDLY has timed-out, and all JEDEC requirements are satisfied. This register may be changed by BIOS/FW on-the-fly.
22	0h RO	**Rsvd_30:** Reserved
21	0h RW	**Close All Pages before Power-Down (PREAPWDEN):** Send Precharge All Command to a Rank before PD-Enter. Setting this bit to 1 will allow sending a PREA command before PDE command. 0 - Disable. 1 - Enable.
20	0h RW	**Wake Allowed for Page Close Timeout (PCLSWKOK):** Setting this bit to 1 indicates the Memory Controller can send DRAM devices a PD-Exit command in order to close single bank if the page timer expired. Note this bit applies only to cases where at least one other bank in the same rank is open but not timed-out. If all banks in the rank timed-out, a PD-Exit command will be sent regardless of this bit. Must be set to 0 during init/training mode. 0 - Disable. 1 - Enable.
19	0h RO	**Rsvd_29:** Reserved
18:16	0h RW	**Page Close Timeout Period (PCLSTO):** Specifies the time frame, in ns, from last access to a DRAM page until that page may be scheduled for closing (by sending a PRE command). 0h - Disable page close timer (init/training). 1h - Immediate page close. 2h - 30-60 ns to page close. 3h - 60-120 ns to page close. 4h - 120-240 ns to page close. 5h - 240-480 ns to page close. 6h - 480-960 ns to page close. 7h - 1-2 s to page close.
15:13	0h RO	**Rsvd_28:** Reserved
12:8	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
7:0	0h RW	**Self-Refresh Entry delay (SREDLY):** The delay, in core-clocks, between PRI idle (no pending requests and PRI status is less than 2) and SR Entry when the Memory Controller is in Dynamic SR mode.

13.4.8 DRAM Refresh Control (DRFC)—Offset 8h

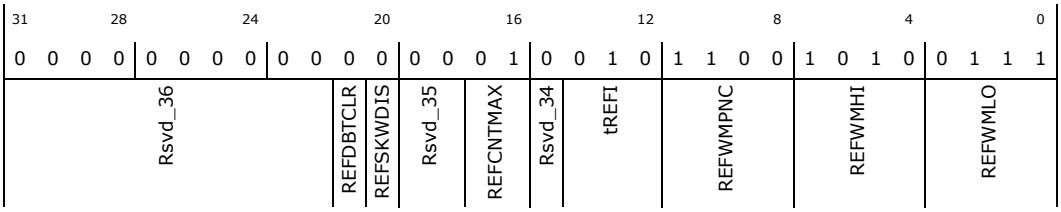
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 8h

Op Codes:
10h - Read, 11h - Write

Default: 00012CA7h



Bit Range	Default & Access	Field Name (ID): Description
31:22	0h RO	**Rsvd_36:** Reserved
21	0h RW	**Clear Refresh Debit before Self Refresh Entry (REFDBTCLR):** To ensure that the Memory Controller sends enough REF commands to the DRAM, it calculates tREFI period with 2% less than the JEDEC tREFI value. So instead of tREFI equaling 7.8us, it's 7.6us, which means over 1000 x tREFI interval, the Memory Controller would have sent 20 REF commands more than required by the JEDEC spec. When this bit is set to 1 and if the Memory Controller was awake for at least 1000 x tREFI period and then enters Self Refresh, the Memory Controller clears the refresh counter and enters Self Refresh without having to send the accumulated REF commands, since it has already issued 20 more REF commands than required by JEDEC. 0h - Disabled. 1h - Enabled.
20	0h RW	**Disable Skewing of Refresh Counting between Ranks (REFSKWDIS):** Each rank has its own refresh counter. By default, incrementing these refresh counters are skewed by 1/4 the tREFI period. Setting this bit to a 1 disables this feature and all refresh counters will increment at the same time per tREFI period. Skewing the tREFI counters can improve performance, since traffic to all ranks does not have to be block to perform refresh. 0h - counters are updated per rank every tREFI. 1h - all counters are updated every tREFI.
19:18	0h RO	**Rsvd_35:** Reserved
17:16	1h RW	**Refresh Max tREFI Interval (REFCNTMAX):** The maximum interval between ant two REF commands per rank. JEDEC allows a maximum of 9 x tREFI intervals. 0h - 6 x tREFI. 1h - 7 x tREFI. 2h - 8 x tREFI. 3h - Reserved. Should not be changed after initial setting.
15	0h RO	**Rsvd_34:** Reserved

Bit Range	Default & Access	Field Name (ID): Description
14:12	2h RW	**Refresh Period (tREFI):** Specifies the average time between sending REF commands to DRAM. The Memory Controller will guarantee that the average time is met, but maintains a certain degree of flexibility in the exact REF scheduling in order to increase overall performance. 0h - Refresh disabled 1h - Reserved for pre-silicon simulation. 2h - 3.9 s (Extended Temperature Range, 85-95 C) 3h - 7.8 s (Normal Temperature Range, 0-85 C)
11:8	Ch RW	**Refresh Panic Watermark (REFWMPNC):** When the refresh debit counter, per rank, is greater than this value, the Memory Controller will send a REF command even if there are some pending requests and regardless of the PRI status level. See DDR3 spec for Refresh Postponing/Pulling-In flexibility. May be changed to functional value after init sequence. Value should be greater than, or equal, to REFWMHI. 0-6h - Reserved 7h - Postpone 2 REF commands. 8h - Postpone 3 REF commands. 9h - Postpone 4 REF commands. Ah - Postpone 5 REF commands. Bh - Postpone 6 REF commands. Ch - Postpone 7 REF commands. Dh - Postpone 8 REF commands. E-Fh - Reserved.
7:4	Ah RW	**Refresh High Watermark (REFWMHI):** When the refresh debit counter, per rank, is greater than this value, the Memory Controller will send a REF command even if there are some pending requests to the rank but not if the PRI status is equal to 3. See DDR3 spec for Refresh Postponing/Pulling-In flexibility. May be changed to functional value after init sequence. Value should be greater than, or equal, to REFWMLO. 0-6h - Reserved 7h - Postpone 2 REF commands. 8h - Postpone 3 REF commands. 9h - Postpone 4 REF commands. Ah - Postpone 5 REF commands. Bh - Postpone 6 REF commands. Ch - Postpone 7 REF commands. Dh - Postpone 8 REF commands. E-Fh - Reserved.
3:0	7h RW	**Refresh Low Watermark (REFWMLO):** When the refresh debit counter, per rank, is greater than this value, the Memory Controller will send a REF command only if there are no pending requests to the rank and the PRI status is less than 3. See DDR3 spec for Refresh Postponing/Pulling-In flexibility. May be changed to functional value after init sequence. 0-6h - Reserved 7h - Postpone 2 REF commands. 8h - Postpone 3 REF commands. 9h - Postpone 4 REF commands. Ah - Postpone 5 REF commands. Bh - Postpone 6 REF commands. Ch - Postpone 7 REF commands. Dh - Postpone 8 REF commands. E-Fh - Reserved.

13.4.9 DRAM Scheduler Control (DSCH)—Offset 9h

Access Method

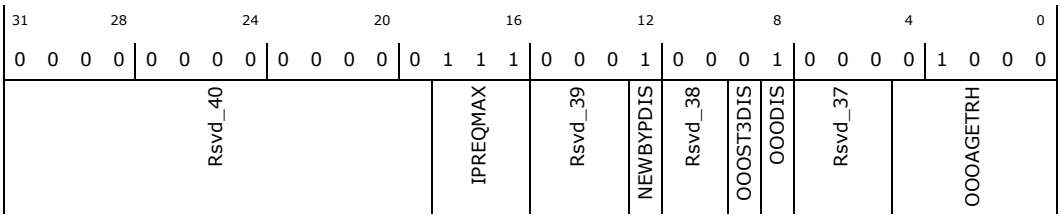
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 9h

Op Codes:

10h - Read, 11h - Write

Default: 00071108h



Bit Range	Default & Access	Field Name (ID): Description
31:19	0h RO	**Rsvd_40:** Reserved
18:16	7h RW	**In-Progress Request Queue Depth (IPREQMAX):** Maximal number of In-Progress Requests stored in the Memory Controller. Number of pending requests = IPREQMAX+1; Value may be changed after init/training when PRI is idle.
15:13	0h RO	**Rsvd_39:** Reserved
12	1h RW	**Disable New Request Bypass (NEWBYPDIS):** Setting this bit to 0 will allow a new request to bypass the normal Memory Controller internal arbiter when there are no pending commands. 0h - Enable New Request Bypass. 1h - Disable New Request Bypass.
11:10	0h RO	**Rsvd_38:** Reserved
9	0h RW	**Out-of-Order Disabled when PRI status is 3 (OOOST3DIS):** Valid only if OOODIS is 0; 0 - Remain OOO if status goes up to 3 1 - Disable OOO if status goes to 3 May be changed after init/training flow.
8	1h RW	**Disable Out-of-Order (OOODIS):** 0h - OOO enabled. 1h - OOO disabled. Should be disabled during init/training and can be enabled for functional mode.
7:5	0h RO	**Rsvd_37:** Reserved
4:0	08h RW	**Out-of-Order Aging Threshold (OOOAGETRH):** Specifies the number of requests that can be processed ahead of another request sitting in the In-Progress request (IPreq) queue before OOO is disabled. Once this threshold is met for any request sitting in the IPreq queue, OOO is disabled until the aged request is processed. This mechanism prevents starvation of pending requests in the IPreq queue. Each request sitting in the IPreq queue has its own age timer. OOOAGETRH sets the default value of an age timer when a request is loaded into the IPreq queue.

13.4.10 DRAM Calibration Control (DCAL)—Offset Ah

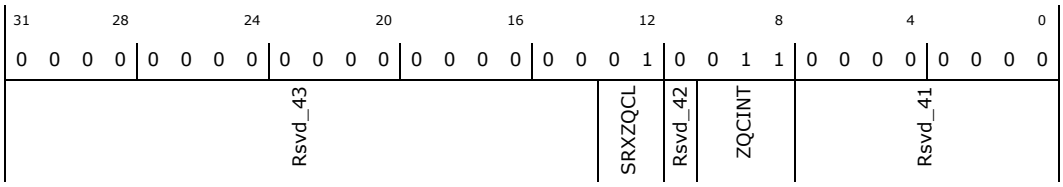
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + Ah

Op Codes:
10h - Read, 11h - Write

Default: 00001300h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	**Rsvd_43:** Reserved
13:12	1h RW/P	**ZQ Calibration Long (SRXZQCL):** Issued After SR Exit Control 0h - ZQCL commands after SRX are sent in parallel. 1h - ZQCL commands are sent serially to ranks. 2h - No ZQCL is sent after SR Exit. (Debug only). 3h Reserved.
11	0h RO	**Rsvd_42:** Reserved
10:8	3h RW/P	**ZQ Calibration Short Interval (ZQCINT):** The time interval, in ms, between ZQCS commands to a DRAM device. ZQCS commands are sent to a single DRAM device and commands are distributed and non-overlapping in the interval. 0h - Disabled. 1h - 62s (for pre-silicon simulation only) 2h - 31ms. 3h - 63ms. 4h - 126ms. 5-7h - Reserved. May be changed on-the-fly in response to thermal events.
7:0	0h RO	**Rsvd_41:** Reserved

13.4.11 DRAM Reset Management Control (DRMC)—Offset Bh

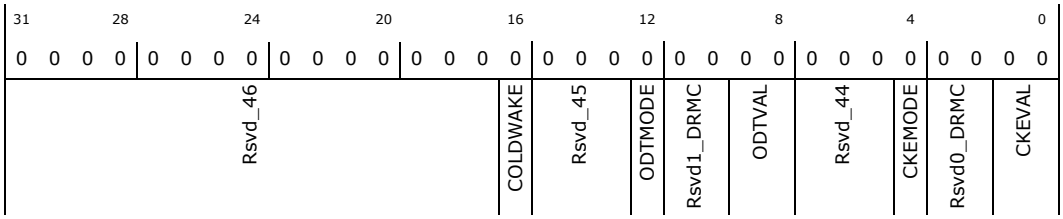
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + Bh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:17	0h RO	**Rsvd_46:** Reserved
16	0h RW	**Cold Wake (COLDWAKE):** BIOS should set this bit to 1 before sending WAKE command to Memory Controller after Cold Reset. For S3 Exit, or any other mode in which the DRAM is in SR, this bit must be set to 0.

Bit Range	Default & Access	Field Name (ID): Description
15:13	0h RO	**Rsvd_45:** Reserved
12	0h RW	**ODT Control Mode (ODTMODE):** 0 - Memory Controller auto controls the ODT pins based on DRAM Write transactions. 1 - The value of ODTVAL above directly controls the ODT pins.
11:10	0h RO	**Rsvd1_DRMC:** Reserved
9:8	0h RW	**ODT Control Value (ODTVAL):** When ODTMODE is set to 1, ODT pins to DRAM are overridden by ODTVAL. Used only during init flow by BIOS.
7:5	0h RO	**Rsvd_44:** Reserved
4	0h RW	**CKE Control Mode (CKEMODE):** 0 - Memory Controller auto controls the CKE pins based on Power-Down and Self Refresh entry and exit . 1 - The value of CKEVAL directly controls the CKE pins
3:2	0h RO	**Rsvd0_DRMC:** Reserved
1:0	0h RW	**CKE Control Value (CKEVAL):** When CKEMODE is set to 1, CKE pins to DRAM are overridden by CKEVAL. Used only during init flow by BIOS.

13.4.12 Power Management Status (PMSTS)—Offset Ch

Access Method

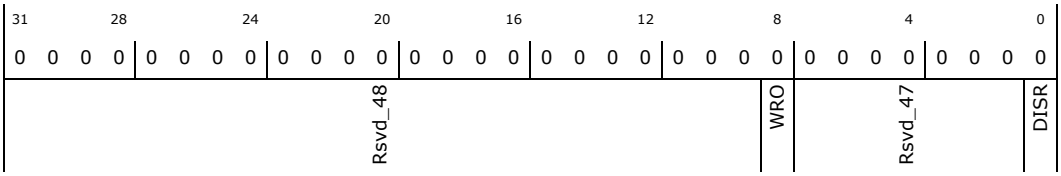
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + Ch

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	**Rsvd_48:** Reserved
8	0h RW/P	**Warm Reset Occurred (WRO):** The Remote Management Unit writes a 1 to this bit to indicate to BIOS a warm reset has just occurred. Can write a 0 to clear it. This is also cleared when powergood = 0. This bit will not clear with reset
7:1	0h RO	**Rsvd_47:** Reserved
0	0h RW/P	**DRAM In Self-Refresh Status (DISR):** The Memory Controller sets this bit to a 1 after it has placed the DRAM devices in Self Refresh mode. The Memory Controller clears this bit when it brings the DRAM devices out of Self Refresh mode. Writing a 1 to this bit, when the COLDWAKE bit is set to 0, will also clear it. This will not clear with system reset, but will clear when powergood = 0. 0 - DRAM not guaranteed to be in Self-Refresh. 1 - DRAM in Self-Refresh

13.4.13 DRAM Control Operation (DCO)—Offset Fh

Access Method

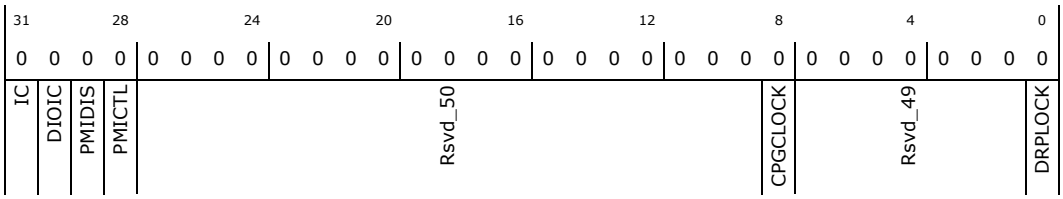
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + Fh

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW	**Memory Controller Initialization Complete (IC):** Indicates that initialization of the Memory Controller has completed. Memory accesses are permitted and maintenance operation begins. Until this bit is set to a 1, the memory controller will not accept DRAM requests from the Memory Manager or the MTE. Note: Set this bit to 1 only when all other Memory Controller registers has been configured. Usually set at the last configuration step.
30	0h RO	**DDRIO PHY initialization complete (DIOIC):** Status indication that DDRIO initialization is complete.
29	0h RW	**Disable PRI interface (PMIDIS):** When this bit is set to 1, the Memory Controller will not respond to requests from either the Memory Manager or the MTE. Note: This bit should be set to 1, when issuing DRAM Read and Write commands through message 68h. It prevents the Memory Controller trying to pull data from the Memory Manager or the MTE for Writes through message 68h, and also prevents the Memory Manager and the MTE from taking read data returned from Reads through message 68h.
28	0h RW	**PRI Control Select (PMICTL):** 0 - Memory Controller PRI is connected to Memory Manager. 1 - Memory Controller PRI is connected to MTE. When this bit is toggled, the Memory Controller will flush all pending DRAM requests from the previous unit before taking requests from the new unit
27:9	0h RO	**Rsvd_50:** Reserved
8	0h RW/P/L	**MTE Lock (CPGCLOCK):** After this bit is set to 1, the MTE is clock gated and locked and cannot be used. CPGCLOCK can be set only once, and will only reset when powergood = 0.
7:1	0h RO	**Rsvd_49:** Reserved
0	0h RW/P/L	**DRP Register Lock (DRPLOCK):** Write a 1 to this bit to lock the DRP and DTRC registers, and to disable the ability to issues the DRAM Read and Write commands through message 68h. Once locked, the DRP and DTRC registers cannot be written again. Once set to 1, this bit can only be cleared when powergood = 0.

13.4.14 Sticky Scratchpad 0 (SSKPD0)—Offset 4Ah

Access Method

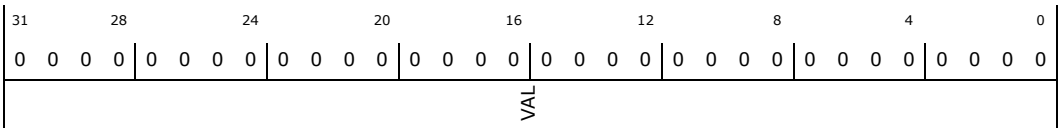
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 4Ah

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW/P	**General Purpose Scratchpad (VAL):** May be used for BIOS for data storage. Value is preserved in warm-reset.

13.4.15 Sticky Scratchpad 1 (SSKPD1)—Offset 4Bh

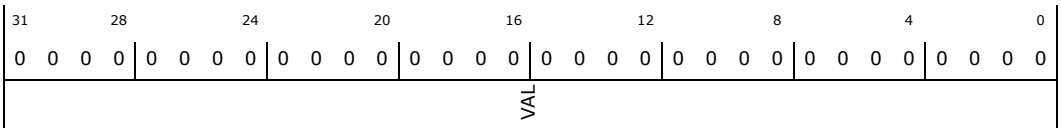
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 4Bh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW/P	**General Purpose Scratchpad (VAL):** May be used for BIOS for data storage. Value is preserved in warm-reset.

13.4.16 DRAM ECC Control Register (DECCCTRL)—Offset 60h

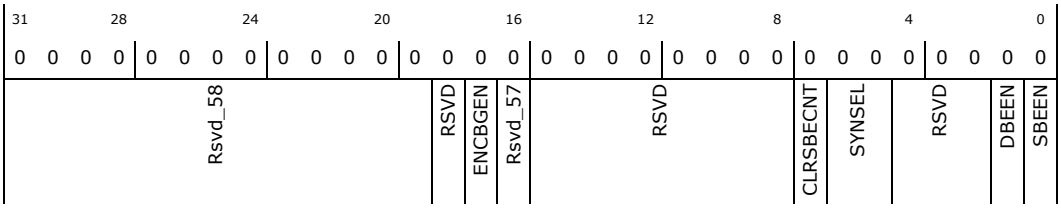
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 60h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:19	0h RO	**Rsvd_58:** Reserved
18	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**Enable Generation of ECC Check Bits (ENCBGEN):** 0 - Disable check bit generation. 1 - Enable check bit generation.
16	0b RO	**Rsvd_57:** Reserved
15:8	00h RO	**Reserved (RSVD):** Reserved.
7	00h RW	**Clear Single Bit Error Count (CLRSBECNT):** Clear ECC Single Bit Error Count: 0 - allow single bit error count to increment. 1 - clear single bit error count.
6:5	00h RW	**Syndrome Select (SYNSEL):** ECC Syndrome Bits Select for Observation: The Syndrome Bits are generated from read data returned from DRAM and used to detect ECC errors. Each 64 bits of read data is used to generate 8 Syndrome Bits. SYNSEL selects which set of Syndrome Bits to mux to the DECCSTAT register for observation. 00 - Selects Syndrome Bits from read data [63:0]. 01 - Selects Syndrome Bits from read data [127:64]. 10 - Selects Syndrome Bits from read data [191:128]. 11 - Selects Syndrome Bits from read data [255:192].
4:2	000h RO	**Reserved (RSVD):** Reserved.
1	0h RW	**Double Bit Enable (DBEEN):** Enable Double Bit Error Detect 0: disable double bit error detect. 1: enable double bit error detect.
0	0h RW	**Single Bit Enable (SBEEN):** Enable Single Bit Error Detect and Correct 0: disable single bit error detect and correct. 1: enable single bit error detect and correct.

13.4.17 DRAM ECC Status (DECCSTAT)—Offset 61h

Access Method

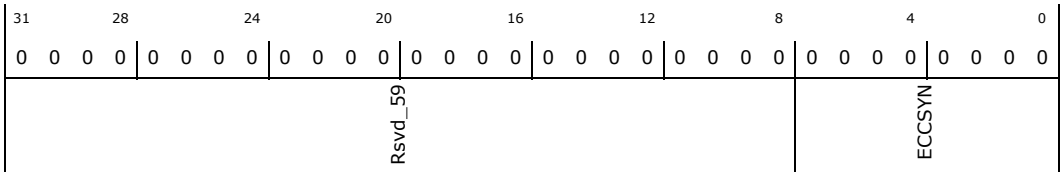
Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 61h

Op Codes:

10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0h RO	**Rsvd_59:** Reserved
7:0	0h RO	**ECC Syndrome Bits (ECCSYN):** This is the 8 ECC Syndrome Bits selected for observation. Selection is made through the DECCCTRL register.

13.4.18 DRAM ECC Single Bit Error Count (DECCSBECNT)—Offset 62h

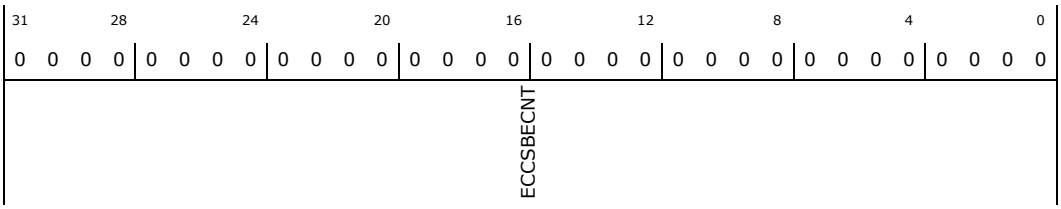
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 62h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO	**ECC Single Bit Error Count (ECCSBECNT):** Write a 1 to the CLRSBECNT bit in the DECCCTRL register to clear this register.

13.4.19 DRAM Single Bit ECC Error Captured Address (DECCSBECA)—Offset 68h

Note: Write any value to this register to clear both this register and the DRAM Single Bit ECC Error Captured Syndrome register

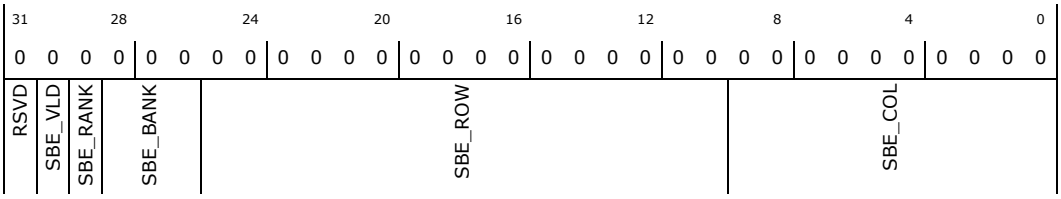
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 68h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RO	**Reserved (RSVD):** Reserved.
30	0h RO/P	**Single Bit ECC Error Valid (SBE_VLD):** 0 - No Single Bit ECC Error was detected. 1 - A Single Bit ECC Error was detected.
29	0h RO/P	**Captured Rank Address (SBE_RANK):** Captured rank address of a read with a Single Bit ECC Error
28:26	0h RO/P	**Captured Bank Address (SBE_BANK):** Captured bank address of a read with a Single Bit ECC Error
25:10	0h RO/P	**Captured Row Address (SBE_ROW):** Captured row address of a read with a Single Bit ECC Error

Bit Range	Default & Access	Field Name (ID): Description
9:0	0h RO/P	**Captured Column Address (SBE_COL):** Captured column address of a read with a Single Bit ECC Error

13.4.20 DRAM Single Bit ECC Error Captured Syndrome (DECCSBECS)—Offset 69h

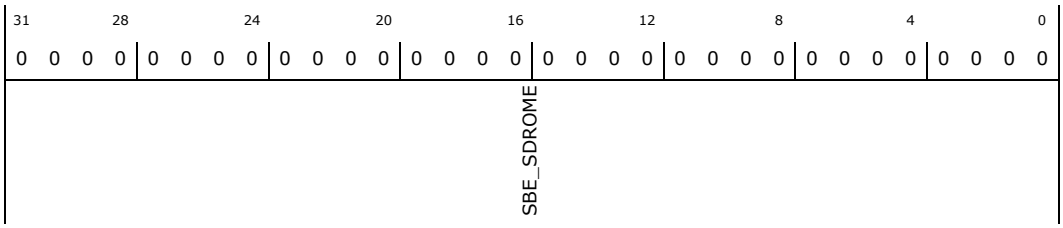
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 69h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO/P	**Captured Syndrome (SBE_SDROME):** Captured syndrome of a read with a Single Bit ECC Error

13.4.21 DRAM Double Bit ECC Error Captured Address (DECCDBECA)—Offset 6Ah

Note: Write any value to this register to clear both this register and the DRAM Double Bit ECC Error Captured Syndrome register

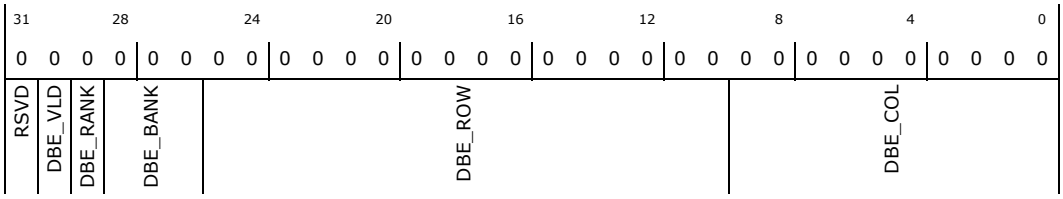
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 6Ah

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
30	0h RO/P	**Double Bit ECC Error Valid (DBE_VLD):** 0 - No Double Bit ECC Error was detected. 1 - A Double Bit ECC Error was detected.
29	0h RO/P	**Captured Rank Address (DBE_RANK):** Captured rank address of a read with a Double Bit ECC Error
28:26	0h RO/P	**Captured Bank Address (DBE_BANK):** Captured bank address of a read with a Double Bit ECC Error
25:10	0h RO/P	**Captured Row Address (DBE_ROW):** Captured row address of a read with a Double Bit ECC Error
9:0	0h RO/P	**Captured Column Address (DBE_COL):** Captured column address of a read with a Double Bit ECC Error

13.4.22 DRAM Double Bit ECC Error Captured Syndrome (DECCDBECS)—Offset 6Bh

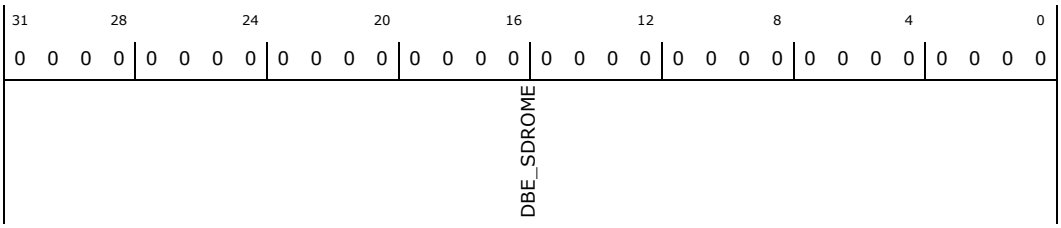
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 6Bh

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO/P	**Captured Syndrome (DBE_SDROME):** Captured syndrome of a read with a Double Bit ECC Error

13.4.23 Memory Controller Fuse Status (DFUSESTAT)—Offset 70h

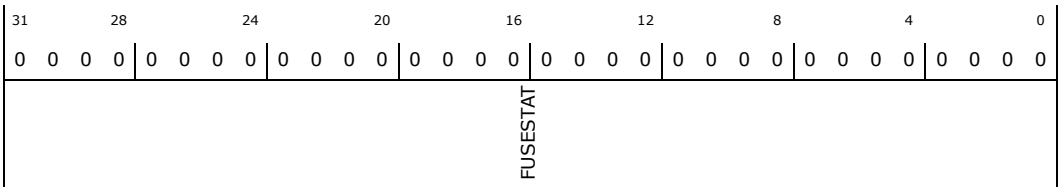
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 70h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO	**Fuse Status (FUSESTAT):** Memory Controller fuse bits are captured in this register. [0] ECC Disable [31:1] Reserved

13.4.24 Scrambler Seed (DSCRMSEED)—Offset 80h

Dynamic data scrambler seed register

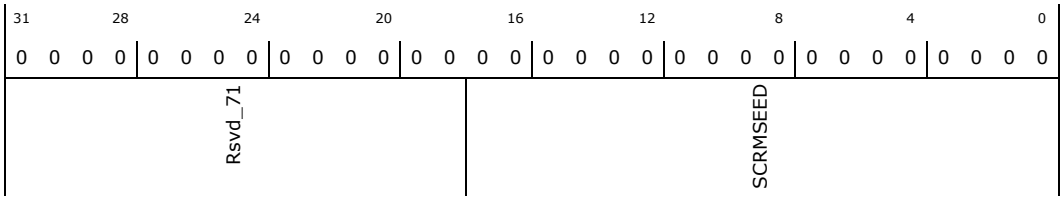
Access Method

Type: Message Bus Register (Size: 32 bits)

Offset: [Port: 0x01] + 80h

Op Codes:
10h - Read, 11h - Write

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0h RO	**Rsvd_71:** Reserved
17:0	0h RW/P	**Scrambler Seed (SCRMSEED):** Holds 18 bit scrambler seed value used to feed into LFSR array matrix.

§ §

14.0 PCI Express* 2.0

There are two lanes and two PCI Express* root ports, each supporting the *PCI Express* Base Specification*, Rev. 2.0 at a maximum 2.5 GT/s signaling rate.

14.1 Signal Descriptions

Please see Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 79. PCI Express* 2.0 Signals

Signal Name	Direction/ Type	Description
PCIE_PETP[1:0] PCIE_PETN[1:0]	O PCIe*	PCI Express* Transmit PCI Express* Ports 3:0 transmit pair (P and N) signals. Each pair makes up the transmit half of a lane.
PCIE_PERP[1:0] PCIE_PERN[1:0]	I PCIe*	PCI Express* Receive PCI Express* Ports 3:0 receive pair (P and N) signals. Each pair makes up the receive half of lane.
PCIE_IRCOMP	IO Analog	Note: Please check the Platform Design Guide for connection details for this COMP pin.
PCIE_RBIAS	I Analog	Note: Please check the Platform Design Guide for connection details for this BIAS pin.

Note: PCIe reference clocks are supplied by the following:
REF[0/1]_OUTCLKP
REF[0/1]_OUTCLKN

14.2 Features

- Conforms to *PCI Express* Base Specification*, Rev. 2.0
- 2.5 GT/s operation per root port
- Virtual Channel support for VC0
- x1 widths
- Supports 2 x1 Root port configurations
- Interrupts and Events
  - — Legacy (INT*x*) and MSI Interrupts
  - — General Purpose Events

  - Express Card Hot Plug Events
  - System Error Events
- Power Management
  - Link State support for L0s, L1, and L2
  - Powered down in ACPI S3 state - L3

14.2.1 Interrupts and Events

A root port is capable of handling interrupts and events from an end point device. A root port can also generate its own interrupts for some events, including power management and hot plug events, and also including error events.

There are two interrupt types a root port receives from an end point device: INTx (legacy), and MSI. MSIs are automatically passed upstream by the root port, just as other memory writes would be. INTx messages are delivered to the Legacy Bridge's interrupt decoding and routing logic by the root port.

Events and interrupts that are handled by the root port are shown in Table 80, with the possible interrupts they can deliver to the interrupt decoder/router.

Table 80. Possible Interrupts Generated From Events/Packets

Packet/Event	Type	INTx	MSI	SERR	SCI	SMI	GPE
INTx	Packet	X	X				
PM_PME	Packet	X	X				
Power Management (PM)	Event	X	X		X	X	
Hot Plug (HP)	Event	X	X		X	X	
ERR_CORR	Packet			X			
ERR_NONFATAL	Packet			X			
ERR_FATAL	Packet			X			
Internal Error	Event			X			
VDM	Packet						X

Note: Table 80 lists the possible interrupts and events generated based on packets received, or events generated in the root port. Configuration is performed by the software to enable the different interrupts as applicable.

Note: GPE is reported as SCI by the root port.

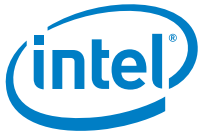
14.2.1.1 Express Card Hot Plug Events

Express Card Hot Plug is available based on Presence Detection for each root port.

Note: A full Hot Plug Controller is not implemented.

Presence detection occurs when a PCI Express* device is plugged in and power is supplied. The physical layer detects the presence of the device, and the root port sets the SLCTL_SLSTS.PDS and SLCTL_SLSTS.PDC bits.

When a device is removed and detected by the physical layer, the root port clears the SLCTL_SLSTS.PDS bit, and sets the SLCTL_SLSTS.PDC bit.



Interrupts can be generated by the root port when a hot plug event occurs. A hot plug event is defined as the transition of the SLCTL_SLSTS.PDC bit from 0 to 1. Software can set the SLCTL_SLSTS.PDE and SLTCTL_SLSTS.HPE bits to allow hot plug events to generate an interrupt.

If SLCTL_SLSTS.PDE and SLTCTL_SLSTS.HPE are both set, and SLCTL_STSTS.PDC transitions from 0 to 1, an interrupt is generated.

14.2.1.2 System Error (SERR)

System Error events are supported by both internal and external sources. See the *PCI Express* Base Specification*, Rev. 2.0 for details.

14.2.2 Power Management

Each root port's link supports L0s, L1, and L2/3 link states per PCI Express* Base Specification, Rev. 2.0. L2/3 is entered on entry to S3.

14.3 References

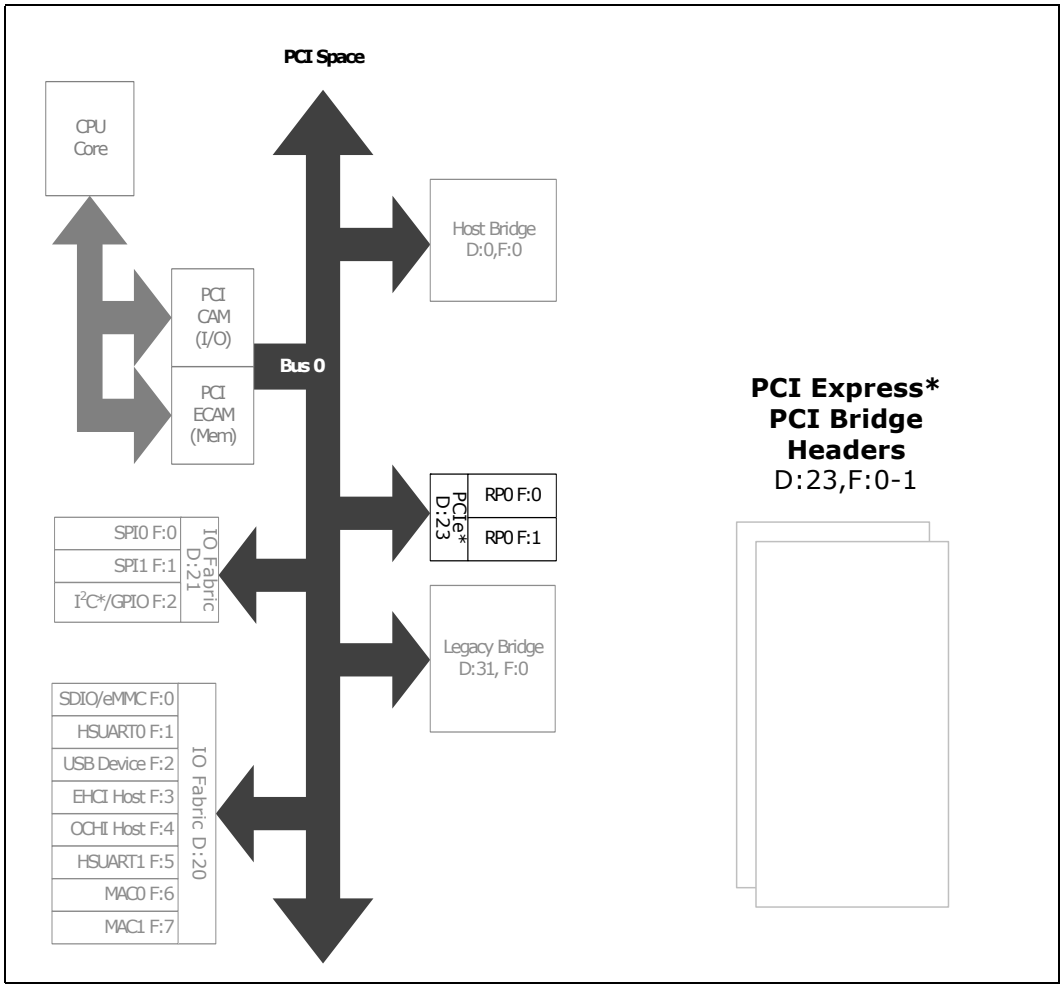
PCI Express Base Specification*, Rev. 2.0

14.4 Register Map

Each root port supports it's own extended PCI bridge header in PCI configuration space. These headers are located on PCI bus 0, device 23, functions 0-1 as shown below. There are no other registers implemented by the root ports or their controller.

See Chapter 5.0, "Register Access Methods" for details on accessing different register types.

Figure 26. PCI Express Register Map



14.5 PCI Configuration Registers

Registers listed are for Function 0 (Root Port 0). Function 1 (Root Port 1) contains the same registers. Differences between Root Ports are noted in the individual registers.

Table 81. Summary of PCI Configuration Registers—0/23/0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Identifiers (ID)—Offset 0h" on page 264	11C38086h
4h	7h	"Primary Status (CMD_PSTS)—Offset 4h" on page 264	00100000h
8h	Bh	"Class Code (RID_CC)—Offset 8h" on page 266	06040000h
Ch	Fh	"Header Type (CLS_PLT_HTYPE)—Offset Ch" on page 266	00810000h
18h	1Bh	"Secondary Latency Timer (BNUM_SLT)—Offset 18h" on page 267	00000000h
1Ch	1Fh	"Secondary Status (IOBL_SSTS)—Offset 1Ch" on page 267	00000000h
20h	23h	"Memory Base and Limit (MBL)—Offset 20h" on page 268	00000000h

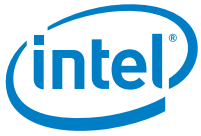


Table 81. Summary of PCI Configuration Registers—0/23/0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
24h	27h	"Prefetchable Memory Base and Limit (PMBL)—Offset 24h" on page 269	00010001h
28h	2Bh	"Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h" on page 269	00000000h
2Ch	2Fh	"Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch" on page 270	00000000h
34h	37h	"Capabilities List Pointer (CAPP)—Offset 34h" on page 270	00000040h
3Ch	3Fh	"Bridge Control (INTR_BCTRL)—Offset 3Ch" on page 271	00000000h
40h	43h	"PCI Express Capabilities (CLIST_XCAP)—Offset 40h" on page 272	00428010h
44h	47h	"Device Capabilities (DCAP)—Offset 44h" on page 273	00008000h
48h	4Bh	"Device Status (DCTL_DSTS)—Offset 48h" on page 274	00100000h
4Ch	4Fh	"Link Capabilities (LCAP)—Offset 4Ch" on page 275	00110C01h
50h	53h	"Link Status (LCTL_LSTS)—Offset 50h" on page 277	10010000h
54h	57h	"Slot Capabilities (SLCAP)—Offset 54h" on page 278	00040060h
58h	5Bh	"Slot Status (SLCTL_SLSTS)—Offset 58h" on page 279	00000000h
5Ch	5Fh	"Root Control (RCTL)—Offset 5Ch" on page 281	00000000h
60h	63h	"Root Status (RSTS)—Offset 60h" on page 281	00000000h
64h	67h	"Device Capabilities 2 (DCAP2)—Offset 64h" on page 282	00000016h
68h	6Bh	"Device Status 2 (DCTL2_DSTS2)—Offset 68h" on page 283	00000000h
6Ch	6Fh	"Link Capability 2 (LCAP2)—Offset 6Ch" on page 284	00000000h
70h	73h	"Link Status 2 (LCTL2_LSTS2)—Offset 70h" on page 284	00000001h
74h	77h	"Slot Capabilities 2 (SLCAP2)—Offset 74h" on page 286	00000000h
78h	7Bh	"Slot Status 2 (SLCTL2_SLSTS2)—Offset 78h" on page 286	00000000h
80h	83h	"Message Signaled Interrupt Message Control (MID_MC)—Offset 80h" on page 287	00009005h
84h	87h	"Message Signaled Interrupt Message Address (MA)—Offset 84h" on page 287	00000000h
88h	8Bh	"Message Signaled Interrupt Message Data (MD)—Offset 88h" on page 288	00000000h
90h	93h	"Subsystem Vendor Capability (SVCAP)—Offset 90h" on page 288	0000A00Dh
94h	97h	"Subsystem Vendor IDs (SVID)—Offset 94h" on page 289	00000000h
A0h	A3h	"PCI Power Management Capabilities (PMCAP_PMC)—Offset A0h" on page 289	C8020001h
A4h	A7h	"PCI Power Management Control And Status (PMCS)—Offset A4h" on page 290	00000000h
D0h	D3h	"Channel Configuration (CCFG)—Offset D0h" on page 291	01000000h
D4h	D7h	"Miscellaneous Port Configuration 2 (MPC2)—Offset D4h" on page 292	00000000h
D8h	DBh	"Miscellaneous Port Configuration (MPC)—Offset D8h" on page 293	01110000h
DCh	DFh	"SMI / SCI Status (SMSCS)—Offset DCh" on page 294	00000000h
F4h	F7h	"Message Bus Control (PHYCTL_PHYCTL2_IOSFSBCTL)—Offset F4h" on page 295	000C3043h
100h	103h	"Advanced Error Reporting Capability Header (AECH)—Offset 100h" on page 296	00000000h
104h	107h	"Uncorrectable Error Status (UES)—Offset 104h" on page 297	00000000h
108h	10Bh	"Uncorrectable Error Mask (UEM)—Offset 108h" on page 298	00000000h
10Ch	10Fh	"Uncorrectable Error Severity (UEV)—Offset 10Ch" on page 299	00060011h
110h	113h	"Correctable Error Status (CES)—Offset 110h" on page 300	00000000h
114h	117h	"Correctable Error Mask (CEM)—Offset 114h" on page 301	00002000h
118h	11Bh	"Advanced Error Capabilities and Control (AECC)—Offset 118h" on page 302	00000000h
11Ch	11Fh	"Header Log (HL_DW1)—Offset 11Ch" on page 302	00000000h

Table 81. Summary of PCI Configuration Registers—0/23/0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
120h	123h	"Header Log (HL_DW2)—Offset 120h" on page 303	00000000h
124h	127h	"Header Log (HL_DW3)—Offset 124h" on page 303	00000000h
128h	12Bh	"Header Log (HL_DW4)—Offset 128h" on page 303	00000000h
12Ch	12Fh	"Root Error Command (REC)—Offset 12Ch" on page 304	00000000h
130h	133h	"Root Error Status (RES)—Offset 130h" on page 304	00000000h
134h	137h	"Error Source Identification (ESID)—Offset 134h" on page 305	00000000h

14.5.1 Identifiers (ID)—Offset 0h

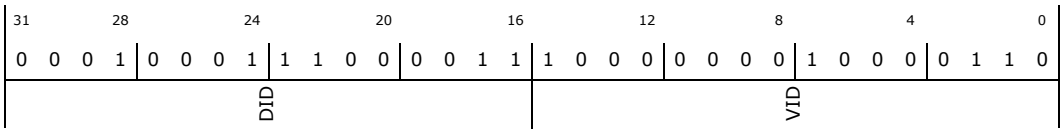
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 0h

Power Well: Core

Default: 11C38086h



Bit Range	Default & Access	Field Name (ID): Description
31:16	11C3h RO/V	**Device Identification (DID):** PCI Device ID
15:0	8086h RO	**Vendor Identification (VID):** PCI Vendor ID

14.5.2 Primary Status (CMD_PSTS)—Offset 4h

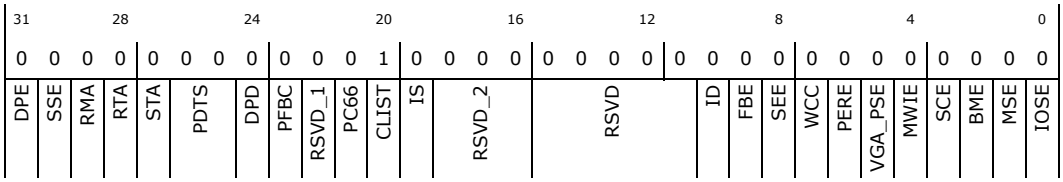
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 4h

Power Well: Core

Default: 00100000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RWC	**DPE Detected Parity Error (DPE):** Set when the root port receives a command or data from the backbone with a parity error. This is set even if PCMD.PERE is not set.

Bit Range	Default & Access	Field Name (ID): Description
30	0b RWC	**Signaled System Error (SSE):** Set when the root port signals a system error to the internal SERR# logic.
29	0b RWC	**Received Master Abort (RMA):** Set when the root port receives a completion with unsupported request status from the backbone.
28	0b RWC	**Received Target Abort (RTA):** Set when the root port receives a completion with completer abort from the backbone.
27	0b RWC	**Signaled Target Abort (STA):** Set whenever the root port forwards a target abort received from the downstream device onto the backbone.
26:25	00b RO	**Primary DEVSEL# Timing Status (PDTS):** Reserved per PCI-Express spec.
24	0b RWC	**Master Data Parity Error Detected (DPD):** Set when the root port receives a completion with a data parity error on the backbone and PCMD.PERE is set.
23	0b RO	**Primary Fast Back to Back Capable (PFBC):** Reserved per PCI-Express spec.
22	0b RO	**Reserved (RSVD_1):** Reserved.
21	0b RO	**Primary 66 MHz Capable (PC66):** Reserved per PCI-Express spec.
20	1b RO	**Capabilities List (CLIST):** Indicates the presence of a capabilities list.
19	0b RO/V	**Interrupt Status (IS):** Indicates status of hot plug and power management interrupts on the root port that result in INTx# message generation. This bit is not set if MSI is enabled. If MSI is not enabled, this bit is set regardless of the state of CMD.ID.
18:16	000b RO	**Reserved (RSVD_2):** Reserved.
15:11	00h RO	**Reserved (RSVD):** Reserved.
10	0b RW/RO	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management and MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set.
9	0b RO	**Fast Back to Back Enable (FBE):** Reserved per PCI-Express spec.
8	0b RW	**SERR# Enable (SEE):** When set, enables the root port to generate an SERR# message when PSTS.SSE is set.
7	0b RO	**Wait Cycle Control (WCC):** Reserved per PCI-Express spec.
6	0b RW	**Parity Error Response Enable (PERE):** Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0b RO	**VGA Palette Snoop (VGA_PSE):** Reserved per PCI-Express spec.
4	0b RO	**Memory Write and Invalidate Enable (MWIE):** Reserved per PCI-Express spec.
3	0b RO	**Special Cycle Enable (SCE):** Reserved per PCI-Express and PCI bridge spec.

Bit Range	Default & Access	Field Name (ID): Description
2	0b RW	**Bus Master Enable (BME):** When set, allows the root port to forward Memory and I/O Read/Write cycles onto the backbone from a PCI-Express device. When this bit is 0b, Memory and I/O requests received at a Root Port must be handled as Unsupported Requests (UR). This bit does not affect forwarding of Completions in either the Upstream or Downstream direction. The forwarding of Requests other than Memory or I/O requests is not controlled by this bit.
1	0b RW	**Memory Space Enable (MSE):** When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are master aborted on the backbone.
0	0b RW	**I/O Space Enable (IOSE):** When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are master aborted on the backbone.

14.5.3 Class Code (RID_CC)—Offset 8h

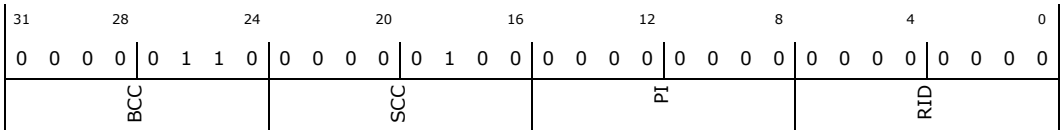
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 8h

Power Well: Core

Default: 06040000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	06h RO	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	04h RO/V	**Sub-Class Code (SCC):** The default indicates the device is a PCI-to-PCI bridge.
15:8	00h RO/V	**Programming Interface (PI):** This is a read only register.
7:0	00h RO/V	**Revision ID (RID):** Indicates the revision of the bridge.

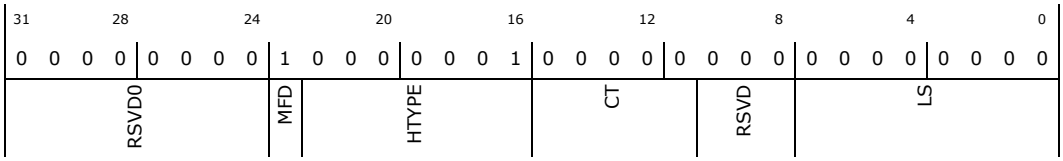
14.5.4 Header Type (CLS_PLT_HTYPE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + Ch

Default: 00810000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0b RO	**RSVD0:** Reserved
23	1b RO	**Multi-function Device (MFD):** This bit is '1' to indicate a multi-function device.
22:16	01h RO/V	**Header Type (HTYPE):** The default mode identifies the header layout of the configuration space, which is a PCI-to-PCI bridge.
15:11	00h RO	**Latency Count (CT):** Reserved per PCI-Express spec.
10:8	000b RO	**Reserved (RSVD):** Reserved.
7:0	00h RW	**Line Size (LS):** This is read/write but contains no functionality, per PCI-Express spec.

14.5.5 Secondary Latency Timer (BNUM_SLT)—Offset 18h

This register is reserved for a root port per PCI-Express spec.

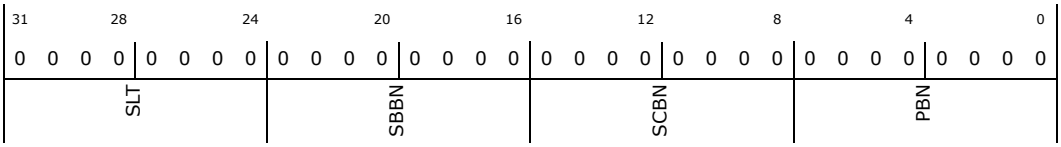
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 18h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RW/RO	**Secondary Latency Timer (SLT):** This register is RO and returns 0. This register does not affect the behavior of any HW logic.
23:16	00h RW	**Subordinate Bus Number (SBBN):** Indicates the highest PCI bus number below the bridge.
15:8	00h RW	**Secondary Bus Number (SCBN):** Indicates the bus number the port.
7:0	00h RW	**Primary Bus Number (PBN):** Indicates the bus number of the backbone.

14.5.6 Secondary Status (IOBL_SSTS)—Offset 1Ch

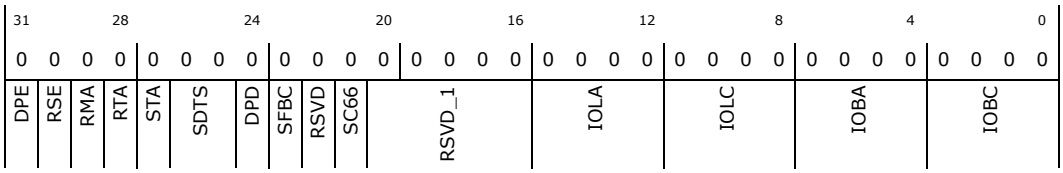
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 1Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RWC	**Detected Parity Error (DPE):** Set when the port receives a poisoned TLP.
30	0b RWC	**Received System Error (RSE):** Set when the port receives an ERR_FATAL or ERR_NONFATAL message from the device.
29	0b RWC	**Received Master Abort (RMA):** Set when the port receives a completion with Unsupported Request status from the device.
28	0b RWC	**Received Target Abort (RTA):** Set when the port receives a completion with Completion Abort status from the device.
27	0b RWC	**Signaled Target Abort (STA):** Set when the port generates a completion with Completion Abort status to the device.
26:25	00b RO/V	**Secondary DEVSEL# Timing Status (SDTS):** Reserved per PCI-Express spec.
24	0b RWC	**Data Parity Error Detected (DPD):** Set when the BCTRL.PERE, and either of the following two conditions occurs: Port receives completion marked poisoned. Port poisons a write request to the secondary side.
23	0b RO/V	**Secondary Fast Back to Back Capable (SFBC):** Reserved per PCI-Express spec.
22	0b RO	**Reserved (RSVD):** Reserved.
21	0b RO	**Secondary 66 MHz Capable (SC66):** Reserved per PCI Express spec.
20:16	00h RO	**Reserved (RSVD_1):** Reserved.
15:12	0h RW	**I/O Address Limit (IOLA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to FFFh.
11:8	0h RO	**I/O Limit Address Capability (IOLC):** Indicates that the bridge does not support 32-bit I/O addressing.
7:4	0h RW	**I/O Base Address (IOBA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to 000h.
3:0	0h RO	**I/O Base Address Capability (IOBC):** Indicates that the bridge does not support 32-bit I/O addressing.

14.5.7 Memory Base and Limit (MBL)—Offset 20h

Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set.

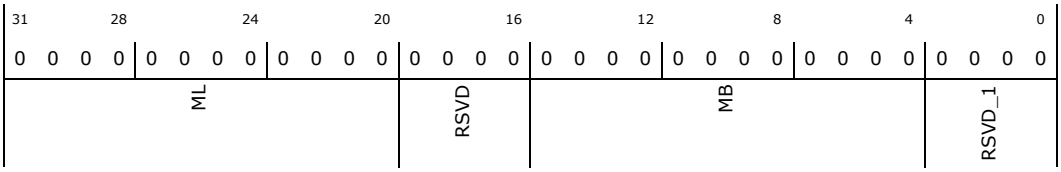
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 20h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	000h RW	**Memory Limit (ML):** These bits are compared with bits 31:20 of the incoming address to determine the upper 1MB aligned value of the range.
19:16	0h RO	**Reserved (RSVD):** Reserved.
15:4	000h RW	**Memory Base (MB):** These bits are compared with bits 31:20 of the incoming address to determine the lower 1MB aligned value of the range.
3:0	0h RO	**Reserved (RSVD_1):** Reserved.

14.5.8 Prefetchable Memory Base and Limit (PMBL)—Offset 24h

Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set.

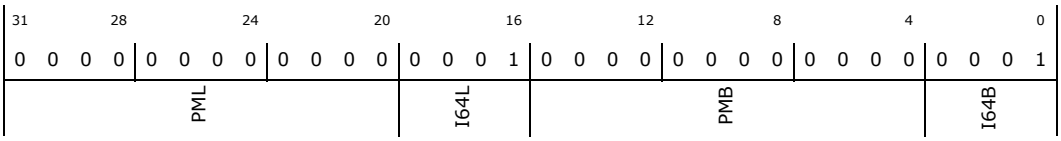
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 24h

Power Well: Core

Default: 00010001h



Bit Range	Default & Access	Field Name (ID): Description
31:20	000h RW	**Prefetchable Memory Limit (PML):** These bits are compared with bits 31:20 of the incoming address to determine the upper 1MB aligned value of the range.
19:16	1h RO	**64-bit Indicator (I64L):** Indicates support for 64-bit addressing.
15:4	000h RW	**Prefetchable Memory Base (PMB):** These bits are compared with bits 31:20 of the incoming address to determine the lower 1MB aligned value of the range.
3:0	1h RO	**64-bit Indicator (I64B):** Indicates support for 64-bit addressing.

14.5.9 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h

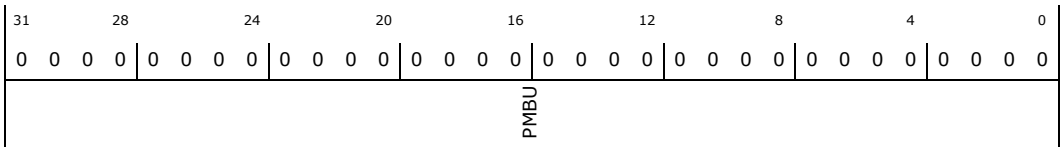
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 28h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Prefetchable Memory Base Upper Portion (PMBU):** Upper 32-bits of the prefetchable address base.

14.5.10 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch

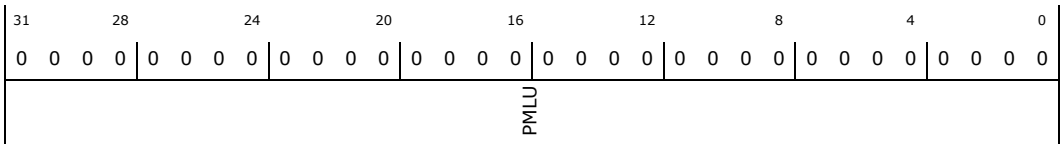
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 2Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Prefetchable Memory Limit Upper Portion (PMLU):** Upper 32-bits of the prefetchable address limit.

14.5.11 Capabilities List Pointer (CAPP)—Offset 34h

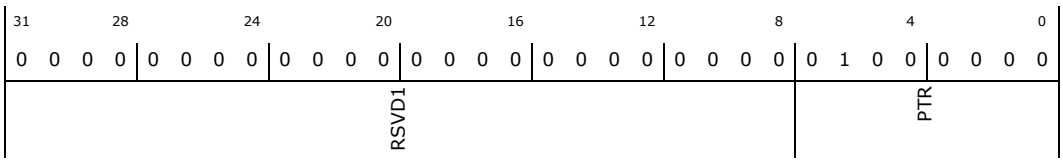
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 34h

Power Well: Core

Default: 00000040h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0000000h RO	**Reserved (RSVD1):** Reserved
7:0	40h RWO	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value. Capability Linked List (Default Settings) Offset Capability Next Pointer 40h PCI Express 80h 80h Message Signaled Interrupt (MSI) 90h 90h Subsystem Vendor A0h A0h PCI Power Management 00h Extended PCIe Capability Linked List Offset Capability Next Pointer 100h Advanced Error Reporting 000h

14.5.12 Bridge Control (INTR_BCTRL)—Offset 3Ch

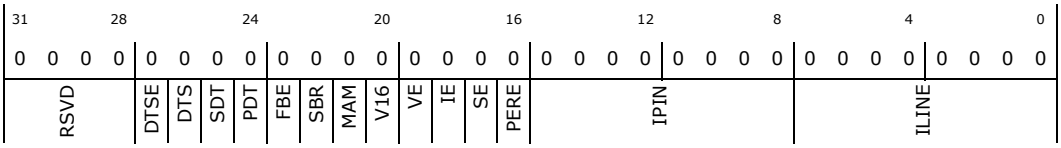
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 3Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:28	0h RO	**Reserved (RSVD):** Reserved.
27	0b RO/RW	**Discard Timer SERR# Enable (DTSE):** Reserved per PCI-Express spec.
26	0b RO	**Discard Timer Status (DTS):** Reserved per PCI-Express spec.
25	0b RO/RW	**Secondary Discard Timer (SDT):** Reserved per PCI-Express spec.
24	0b RO/RW	**Primary Discard Timer (PDT):** Reserved per PCI-Express spec.
23	0b RO	**Fast Back to Back Enable (FBE):** Reserved per Express spec.
22	0b RW	**Secondary Bus Reset (SBR):** Triggers a Hot Reset on the PCI-Express port.
21	0b RO/RW	**Master Abort Mode (MAM):** Reserved per PCI-Express spec.

Bit Range	Default & Access	Field Name (ID): Description
20	0b RW	**VGA 16-Bit Decode (V16):** When set, indicates that the I/O aliases of the VGA range (see BCTRL.VE definition below), are not enabled, and only the base I/O ranges can be decoded. 0: Execute 10-bit address decode on VGA I/O accesses. 1: Execute 16-bit address decode on VGA I/O accesses.
19	0b RW	**VGA Enable (VE):** When set, the following ranges will be claimed off the backbone by the root port: Memory ranges A0000h-BFFFFh I/O ranges 3B0h-3BBh and 3C0h-3DFh, and all aliases of bits 15:10 in any combination of 1's
18	0b RW	**ISA Enable (IE):** This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64KB of PCI I/O space. If this bit is set, the root port will block any forwarding from the backbone to the device of I/O transactions addressing the last 768 bytes in each 1KB block (offsets 100h to 3FFh).
17	0b RW	**SERR# Enable (SE):** When set, ERR_COR, ERR_NONFATAL, and ERR_FATAL messages received are forwarded to the backbone. When cleared, they are not.
16	0b RW	**Parity Error Response Enable (PERE):** When set, poisoned write TLPs and completions indicating poisoned TLPs will set the SSTS.DPD.
15:8	00h RO/V	**Interrupt Pin (IPIN):** Indicates the interrupt pin driven by the root port. At reset, this register takes on the following values, which reflect the reset state of the D28IP register in chipset config space: Port Bits(15:12) Bits(11:08) 1 0h D28IP.P1IP 2 0h D28IP.P2IP 3 0h D28IP.P3IP 4 0h D28IP.P4IP 5 0h D28IP.P5IP 6 0h D28IP.P6IP 7 0h D28IP.P7IP 8 0h D28IP.P8IP The value that is programmed into D28IP is always reflected in this register.
7:0	00h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

14.5.13 PCI Express Capabilities (CLIST_XCAP)—Offset 40h

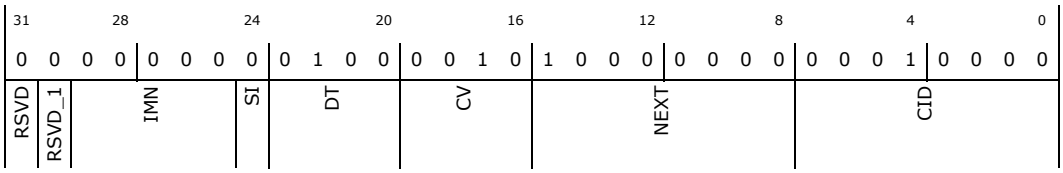
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 40h

Power Well: Core

Default: 00428010h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Reserved (RSVD):** Reserved.
30	0b RO	**Reserved (RSVD_1):** This register at one time was for TCS Routing but that was later removed from the PCIe 2.0 spec.

Bit Range	Default & Access	Field Name (ID): Description
29:25	00h RO	**Interrupt Message Number (IMN):** The PCH does not have multiple MSI interrupt numbers.
24	0b RWO	**Slot Implemented (SI):** Indicates whether the root port is connected to a slot. Slot support is platform specific. BIOS programs this field, and it is maintained until a platform reset.
23:20	4h RO	**Device / Port Type (DT):** Indicates this is a PCI-Express root port.
19:16	2h RO	**Capability Version (CV):** Version 2.0 indicates devices compliant to the PCI Express 2.0 specification which incorporates the Register Expansion ECN.
15:8	80h RWO	**Next Capability (NEXT):** Indicates the location of the next capability. The default value of this register is 80h which points to the MSI Capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	10h RO	**Capability ID (CID):** Indicates this is a PCI Express capability.

14.5.14 Device Capabilities (DCAP)—Offset 44h

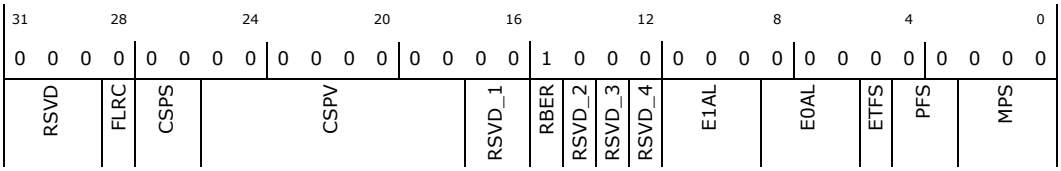
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 44h

Power Well: Core

Default: 00008000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	000b RO	**Reserved (RSVD):** Reserved.
28	0b RO	**Function Level Reset Capable (FLRC):** Not supported in Root Ports.
27:26	00b RO	**Captured Slot Power Limit Scale (CSPS):** Not supported.
25:18	00h RO	**Captured Slot Power Limit Value (CSPV):** Not supported.
17:16	00b RO	**Reserved (RSVD_1):** Reserved.
15	1b RO	**Role Based Error Reporting (RBER):** Indicates that this device implements the functionality defined in the Error Reporting ECN as required by the PCI Express 1.1 spec.
14	0b RO	**Reserved (RSVD_2):** On previous version of the specification this was Power Indicator Present (PIP).
13	0b RO	**Reserved (RSVD_3):** On previous version of the specification this was Attention Indicator Present (AIP).

Bit Range	Default & Access	Field Name (ID): Description
12	0b RO	**Reserved (RSVD_4):** On previous version of the specification this was Attention Button Present (ABP).
11:9	000b RO	**Endpoint L1 Acceptable Latency (E1AL):** Reserved for Root port.
8:6	000b RO	**Endpoint L0 Acceptable Latency (E0AL):** Reserved for Root port.
5	0b RO	**Extended Tag Field Supported (ETFS):** The PCH root port never needs to initiate a transaction as a Requester with the Extended Tag bits being set. This bit does not affect the root port's ability to forward requests as a bridge as the root port always supports forwarding requests with extended tags.
4:3	00b RO	**Phantom Functions Supported (PFS):** No phantom functions supported.
2:0	000b RO	**Max Payload Size Supported (MPS):** Indicates the maximum payload size supported is 128B.

14.5.15 Device Status (DCTL_DSTS)—Offset 48h

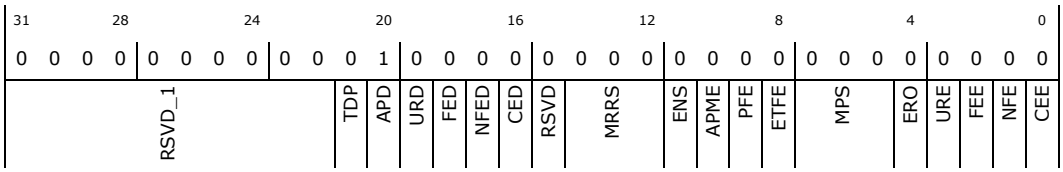
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 48h

Power Well: Core

Default: 00100000h



Bit Range	Default & Access	Field Name (ID): Description
31:22	000h RO	**Reserved (RSVD_1):** Reserved.
21	0b RO	**Transactions Pending (TDP):** This bit has no meaning for the root port since it never initiates a non-posted request with its own RequesterID.
20	1b RO	**AUX Power Detected (APD):** The root port contains AUX power for wakeup.
19	0b RWC	**Unsupported Request Detected (URD):** Indicates an unsupported request was detected.
18	0b RWC	**Fatal Error Detected (FED):** Indicates a fatal error was detected. Set when a fatal error occurred from a data link protocol error, buffer overflow, or malformed TLP.
17	0b RWC	**Non-Fatal Error Detected (NFED):** Indicates a non-fatal error was detected. Set when received a non-fatal error occurred from a poisoned TLP, unexpected completions, unsupported requests, completer abort, or completer timeout.
16	0b RWC	**Correctable Error Detected (CED):** Indicates a correctable error was detected. Set when received an internal correctable error from receiver errors / framing errors, TLP CRC error, DLLP CRC error, replay number rollover, or replay timeout.
15	0b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
14:12	000b RO	**Max Read Request Size (MRRS):** Hardwired to 0
11	0b RO	**Enable No Snoop (ENS):** Not supported. The root port will never issue non-snoop requests.
10	0b RW/P	**Aux Power PM Enable (APME):** Must be RW for OS testing. The OS will set this bit to '1' if the device connected has detected aux power. It has no effect on the root port otherwise. This registers is in the resume well.
9	0b RO	**Phantom Functions Enable (PFE):** Not supported.
8	0b RO	**Extended Tag Field Enable (ETFE):** Not supported.
7:5	000b RO	**Max Payload Size (MPS):** The root port only supports 128B payloads.
4	0b RO	**Enable Relaxed Ordering (ERO):** Not supported.
3	0b RW	**Unsupported Request Reporting Enable (URE):** When set, allows signaling ERR_NONFATAL, ERR_FATAL, or ERR_COR to the Root Control register when detecting an unmasked Unsupported Request (UR). An ERR_COR is signaled when a unmasked Advisory Non-Fatal UR is received. An ERR_FATAL, ERR_or NONFATAL, is sent to the Root Control Register when an uncorrectable non-Advisory UR is received with the severity set by the Uncorrectable Error Severity register.
2	0b RW	**Fatal Error Reporting Enable (FEE):** Enables signaling of ERR_FATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.
1	0b RW	**Non-Fatal Error Reporting Enable (NFE):** When set, enables signaling of ERR_NONFATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.
0	0b RW	**Correctable Error Reporting Enable (CEE):** When set, enables signaling of ERR_CORR to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.

14.5.16 Link Capabilities (LCAP)—Offset 4Ch

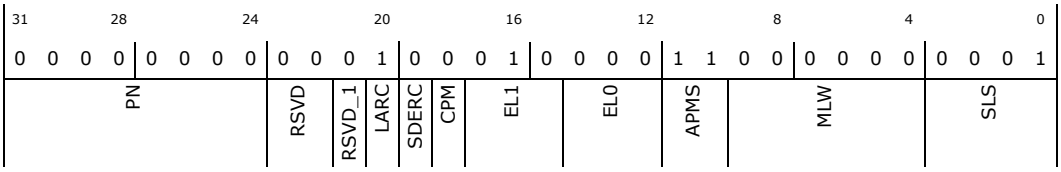
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 4Ch

Power Well: Core

Default: 00110C01h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RO/V	**Port Number (PN):** Indicates the port number for the root port. This value is different for each implemented port: Port # Value of PN field 1 01h 2 02h 3 03h 4 04h 5 05h 6 06h 7 07h 8 08h
23:22	00b RO	**Reserved (RSVD):** Reserved.
21	0b RO	**Reserved (RSVD_1):** This port does not support Link Bandwidth Notification Capability.
20	1b RO	**Link Active Reporting Capable (LARC):** This port supports the optional capability of reporting the DL_Active state of the Data Link Control and Management State Machine.
19	0b RO	**Surprise Down Error Reporting Capable (SDERC):** Set to '0' to indicate the PCH does not support Surprise Down Error Reporting.
18	0b RO	**Clock Power Management (CPM):** '0' Indicates that PCH root ports do not support the CLKREQ# mechanism.
17:15	010b RWO	**L1 Exit Latency (EL1):** Indicates an exit latency of 2 s to 4 s. 000b Less than 1 s 001b 1 s to less than 2 s 010b 2 s to less than 4 s 011b 4 s to less than 8 s 100b 8 s to less than 16 s 101b 16 s to less than 32 s 110b 32 s to 64 s 111b More than 64 s Note: If PXP PLL shutdown is enabled, BIOS should program this latency to comprehend PLL lock latency.
14:12	000b RO/V	**L0s Exit Latency (EL0):** Indicates an exit latency based upon common-clock configuration: LCAP.CCC Value 0 MPC.UCEL 1 MPC.CCEL
11:10	11b RWO	**Active State Link PM Support (APMS):** Indicates the level of active state power management on this link: Bits Definition 00 (Reserved) 01 L0s Entry supported 10 Reserved 11 Both L0s and L1 supported
9:4	000000b RO/V	**Maximum Link Width (MLW):** For the root ports, several values can be taken, based upon the value of the chipset configuration register field RPC.PC1 for ports 1-4 and RPC.PC2 for ports 5-6: Port # Value of PN field RPC.PC1 00 01 10 11 1 01h 02h 02h 04h 2 01h 01h 01h 01h 3 01h 01h 02h 01h 4 01h 01h 01h 01h Port # Value of PN field RPC.PC2 00 01 10 11 5 01h 02h 02h 04h 6 01h 01h 01h 01h 7 01h 01h 02h 01h 8 01h 01h 01h 01h
3:0	1h RO/V	**Supported Link Speeds (SLS):** Indicates the supported link speeds of the Root Port. 0001b 2.5 GT/s Link speed supported 0010b 5.0 GT/s and 2.5GT/s Link speeds supported

14.5.17 Link Status (LCTL_LSTS)—Offset 50h

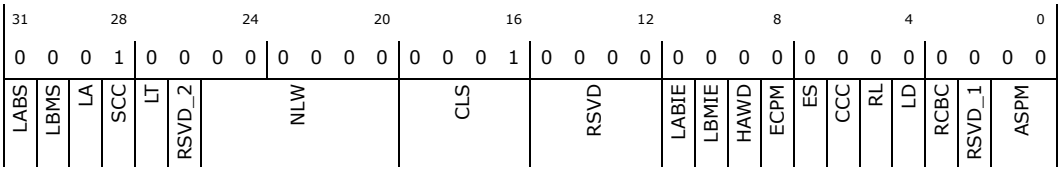
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 50h

Power Well: Core

Default: 10010000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RWC	**Link Autonomous Bandwidth Status (LABS):** This bit is Set by hardware to indicate that hardware has autonomously changed Link speed or width, without the Port transitioning through DL_Down status, for reasons other than to attempt to correct unreliable Link operation. This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was indicated as an autonomous change. The default value of this bit is 0b.
30	0b RWC	**Link Bandwidth Management Status (LBMS):** This bit is Set by hardware to indicate that either of the following has occurred without the Port transitioning through DL_Down status: * A Link retraining has completed following a write of 1b to the Retrain Link bit Note: This bit is Set following any write of 1b to the Retrain Link bit, including when the Link is in the process of retraining for some other reason. * Hardware has changed Link speed or width to attempt to correct unreliable Link operation, either through an LTSSM timeout or a higher level process This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was not indicated as an autonomous change. The default value of this bit is 0b.
29	0b RO/V	**Link Active (LA):** Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.
28	1b RO	**Slot Clock Configuration (SCC):** PCH uses the same reference clock as on the platform and does not generate its own clock.
27	0b RO/V	**Link Training (LT):** The root port sets this bit whenever link training is occurring, or that 1b was written to the Retrain Link bit but Link training has not yet begun. It clears the bit upon completion of link training.
26	0b RO	**Reserved (RSVD_2):** Previously this was defined as Link Training Error (LTE) but support for this bit was removed from subsequent versions of the PCI Express specification.
25:20	000000b RO/V	**Negotiated Link Width (NLW):** For the root ports, this register could take on several values: Port # Value of PN field RPC.PC1 00 01 10 11 1 01h 02h 02h 04h 2 01h 01h 01h 01h 3 01h 01h 02h 01h 4 01h 01h 01h 01h Port # Value of PN field RPC.PC2 00 01 10 11 5 01h 02h 02h 04h 6 01h 01h 01h 01h 7 01h 01h 02h 01h 8 01h 01h 01h 01h The value of this register is undefined if the link has not successfully trained.

Bit Range	Default & Access	Field Name (ID): Description
19:16	1h RO/V	**Current Link Speed (CLS):** 0001b Link is 2.5 GT/s Link 0010b Link is 5.0 GT/s Link The value of this field is undefined if the link is not up.
15:12	0h RO	**Reserved (RSVD):** Reserved.
11	0b RW	**Link Autonomous Bandwidth Interrupt Enable (LABIE):** When Set, this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been Set.
10	0b RW	**Link Bandwidth Management Interrupt Enable (LBMIE):** When Set, this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been Set. This bit is not applicable and is reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches. Functions that do not implement the Link Bandwidth Notification Capability must hardwire this bit to 0b. Default value of this bit is 0b.
9	0b RW	**Hardware Autonomous Width Disable (HAWD):** When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width. Default value of this bit is 0b.
8	0b RO	**Enable Clock Power Management (ECPM):** Reserved. Not supported on PCH Root Ports.
7	0b RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0.
6	0b RW	**Common Clock Configuration (CCC):** When set, indicates that the PCH and device are operating with a distributed common reference clock.
5	0b WO	**Retrain Link (RL):** When set, the root port will train its downstream link. This bit always returns '0' when read. Software uses LSTS.LT and LSTS.LTE to check the status of training. It is permitted to write 1b to this bit while simultaneously writing modified values to other fields in this register. If the LTSSM is not already in Recovery or Configuration, the resulting Link training must use the modified values. If the LTSSM is already in Recovery or Configuration, the modified values are not required to affect the Link training that's already in progress.
4	0b RW	**Link Disable (LD):** When set, the root port will disable the link by directing the LTSSM to the Disabled state.
3	0b RO	**Read Completion Boundary Control (RCBC):** Indicates the read completion boundary is 64 bytes.
2	0b RO	**Reserved (RSVD_1):** Reserved.
1:0	00b RW	**Active State Link PM Control (ASPM):** Indicates whether the root port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled The value of this register is used unless the Root Port ASPM Control Override Enable register is set, in which case the Root Port ASPM Control Override value is used.

14.5.18 Slot Capabilities (SLCAP)—Offset 54h

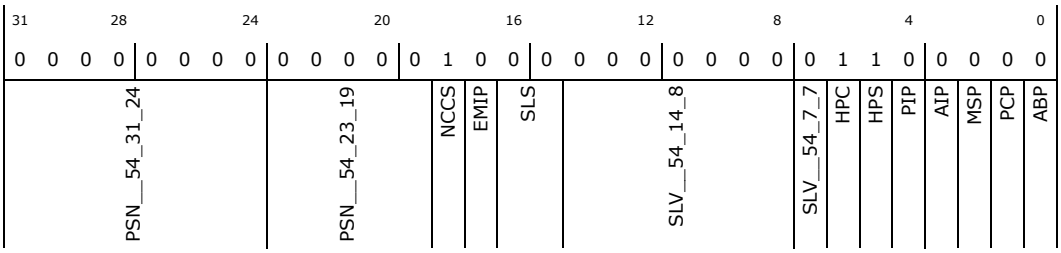
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 54h

Power Well: Core

Default: 00040060h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RWO	**Physical Slot Number (PSN__54_31_24):** This is a value that is unique to the slot number. BIOS sets this field and it remains set until a platform reset.
23:19	00h RWO	**Physical Slot Number (PSN__54_23_19):** This is a value that is unique to the slot number. BIOS sets this field and it remains set until a platform reset.
18	1b RO	**No Command Completed Support (NCCS):** Set to '1' as this port does not implement a Hot Plug controller and can handle back-2-back writes to all fields of the slot control register without delay between successive writes.
17	0b RO	**Electromechanical Interlock Present (EMIP):** Set to 0 to indicate that no electro-mechanical interlock is implemented.
16:15	00b RWO	**Slot Power Limit Scale (SLS):** Specifies the scale used for the slot power limit value. BIOS sets this field and it remains set until a platform reset.
14:8	00h RWO	**Slot Power Limit Value (SLV__54_14_8):** Specifies the upper limit (in conjunction with SLS value), on the upper limit on power supplied by the slot. The two values together indicate the amount of power in watts allowed for the slot. BIOS sets this field and it remains set until a platform reset.
7	0b RWO	**Slot Power Limit Value (SLV__54_7_7):** Specifies the upper limit (in conjunction with SLS value), on the upper limit on power supplied by the slot. The two values together indicate the amount of power in watts allowed for the slot. BIOS sets this field and it remains set until a platform reset.
6	1b RWO	**Hot Plug Capable (HPC):** When set, Indicates that hot plug is supported.
5	1b RWO	**Hot Plug Surprise (HPS):** When set, indicates the device may be removed from the slot without prior notification.
4	0b RO	**Power Indicator Present (PIP):** Indicates that a power indicator LED is not present for this slot.
3	0b RO	**Attention Indicator Present (AIP):** Indicates that an attention indicator LED is not present for this slot.
2	0b RO	**MRL Sensor Present (MSP):** Indicates that an MRL sensor is not present.
1	0b RO	**Power Controller Present (PCP):** Indicates that a power controller is not implemented for this slot.
0	0b RO	**Attention Button Present (ABP):** Indicates that an attention button is not implemented for this slot.

14.5.19 Slot Status (SLCTL_SLSTS)—Offset 58h

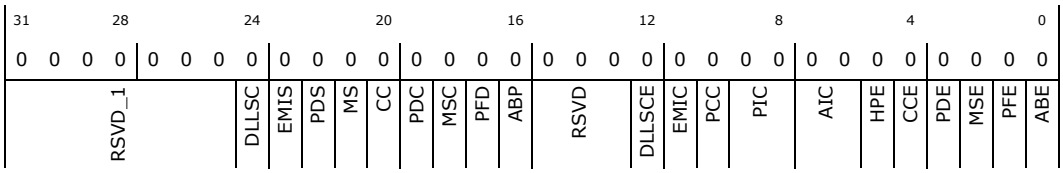
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 58h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	00h RO	**Reserved (RSVD_1):** Reserved.
24	0b RWC	**Data Link Layer State Changed (DLLSC):** This bit is set when the value reported in Data Link Layer Link Active field of the Link Status register is changed. In response to a Data Link Layer State Changed event, software must read Data Link Layer Link Active field of the Link Status register to determine if the link is active before initiating configuration cycles to the hot plugged device.
23	0b RO	**Electromechanical Interlock Status (EMIS):** Reserved as this port does not support and electromechanical interlock.
22	0b RO/V	**Presence Detect State (PDS):** If XCAP.SI is set (indicating that this root port spawns a slot), then this bit indicates whether a device is connected ('1') or empty ('0'). If XCAP.SI is cleared, this bit is a '1'.
21	0b RO	**MRL Sensor State (MS):** Reserved as the MRL sensor is not implemented.
20	0b RO	**Command Completed (CC):** This register is RO as this port does not implement a Hot Plug Controller.
19	0b RWC	**Presence Detect Changed (PDC):** This bit is set by the root port when the PD bit changes state.
18	0b RO	**MRL Sensor Changed (MSC):** Reserved as the MRL sensor is not implemented.
17	0b RO	**Power Fault Detected (PFD):** Reserved as a power controller is not implemented.
16	0b RO	**Attention Button Pressed (ABP):** This register is RO as this port does not implement an attention button.
15:13	000b RO	**Reserved (RSVD):** Reserved.
12	0b RW	**Data Link Layer State Changed Enable (DLLSCE):** When set, this field enables generation of a hot plug interrupt when the Data Link Layer Link Active field is changed.
11	0b RO	**Electromechanical Interlock Control (EMIC):** Reserved as this port does not support an Electromechanical Interlock.
10	0b RO	**Power Controller Control (PCC):** This bit has no meaning for module based hot plug.
9:8	00b RO	**Power Indicator Control (PIC):** This register is RO as this port does not implement a Hot Plug Controller.
7:6	00b RO	**Attention Indicator Control (AIC):** This register is RO as this port does not implement a Hot Plug Controller.
5	0b RW	**Hot Plug Interrupt Enable (HPE):** When set, enables generation of a hot plug interrupt on enabled hot plug events.
4	0b RO	**Command Completed Interrupt Enable (CCE):** This register is RO as this port does not implement a Hot Plug Controller.
3	0b RW	**Presence Detect Changed Enable (PDE):** When set, enables the generation of a hot plug interrupt or wake message when the presence detect logic changes state.

Bit Range	Default & Access	Field Name (ID): Description
2	0b RO	**MRL Sensor Changed Enable (MSE):** This register is RO as this port does not implement a Hot Plug Controller.
1	0b RO	**Power Fault Detected Enable (PFE):** This register is RO as this port does not implement a Hot Plug Controller.
0	0b RO	**Attention Button Pressed Enable (ABE):** This register is RO as this port does not implement a Hot Plug Controller.

14.5.20 Root Control (RCTL)—Offset 5Ch

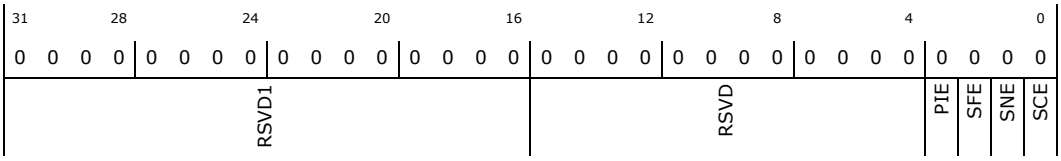
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 5Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	00000h RO	**Reserved (RSVD1):** Reserved
15:4	000h RO	**Reserved (RSVD):** Reserved.
3	0b RW	**PME Interrupt Enable (PIE):** When set, enables interrupt generation when RSTS.PS is in a set state (either due to a '0' to '1' transition, or due to this bit being set with RSTS.PS already set).
2	0b RW	**System Error on Fatal Error Enable (SFE):** When set, an SERR# will be generated if a fatal error is reported by any of the devices in the hierarchy of this root port, including fatal errors in this root port. This register is not dependent on CMD.SEE being set.
1	0b RW	**System Error on Non-Fatal Error Enable (SNE):** When set, an SERR# will be generated if a non-fatal error is reported by any of the devices in the hierarchy of this root port, including non-fatal errors in this root port. This register is not dependent on CMD.SEE being set.
0	0b RW	**System Error on Correctable Error Enable (SCE):** When set, an SERR# will be generated if a correctable error is reported by any of the devices in the hierarchy of this root port, including correctable errors in this root port. This register is not dependent on CMD.SEE being set.

14.5.21 Root Status (RSTS)—Offset 60h

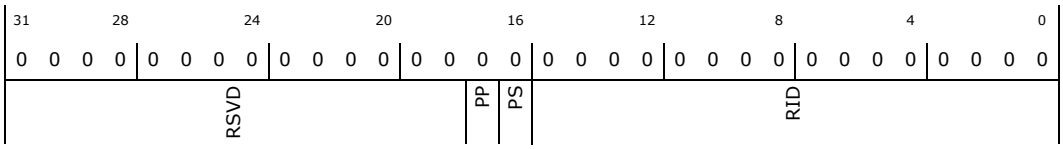
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 60h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0000h RO	**Reserved (RSVD):** Reserved.
17	0b RO/V	**PME Pending (PP):** Indicates another PME is pending when the PME status bit is set. When the original PME is cleared by software, it will be set again, the requestor ID will be updated, and this bit will be cleared. PCH root ports have a one deep PME pending queue.
16	0b RWC	**PME Status (PS):** Indicates that PME was asserted by the requestor ID in RID. Subsequent PMEs are kept pending until this bit is cleared.
15:0	0000h RO/V	**PME Requestor ID (RID):** Indicates the PCI requestor ID of the last PME requestor. Valid only when PS is set. PCH root ports are capable of storing the requester ID for two PM_PME messages, with one active (this register) and a one deep pending queue. Subsequent PM_PME messages will be dropped.

14.5.22 Device Capabilities 2 (DCAP2)—Offset 64h

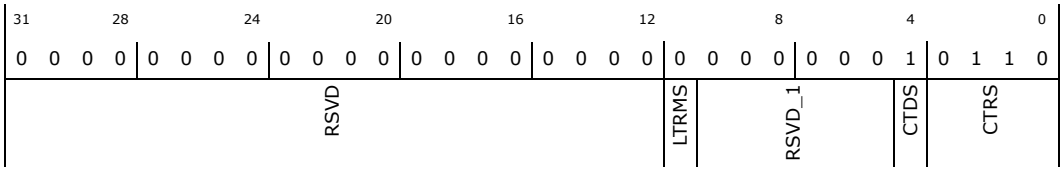
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 64h

Power Well: Core

Default: 00000016h



Bit Range	Default & Access	Field Name (ID): Description
31:12	00000h RO	**Reserved (RSVD):** Reserved.
11	0b RWO	**LTR Mechanism Supported (LTRMS):** A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism capability. BIOS must write to this register with either a '1' or a '0' to enable/disable the root port from declaring support for the LTR capability.
10:5	00h RO	**Reserved (RSVD_1):** Reserved.
4	1b RO	**Completion Timeout Disable Supported (CTDS):** A value of 1b indicates support for the Completion Timeout Disable mechanism.

Bit Range	Default & Access	Field Name (ID): Description
3:0	6h RO	**Completion Timeout Ranges Supported (CTRS):** This field indicates device support for the optional Completion Timeout programmability mechanism. This mechanism allows system software to modify the Completion Timeout value. This field is applicable only to Root Ports, Endpoints that issue requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of requests issued on PCI Express. For all other devices this field is reserved and must be hardwired to 0000b. Four time value ranges are defined: Range A: 50us to 10ms Range B: 10ms to 250ms Range C: 250ms to 4s Range D: 4s to 64s Bits are set according to the table below to show timeout value ranges supported. 0000b Completion Timeout programming not supported. 0001b Range A 0010b Range B 0011b Ranges A and B 0110b Ranges B and C 0111b Ranges A, B and C 1110b Ranges B, C and D 1111b Ranges A, B, C and D All other values are reserved.

14.5.23 Device Status 2 (DCTL2_DSTS2)—Offset 68h

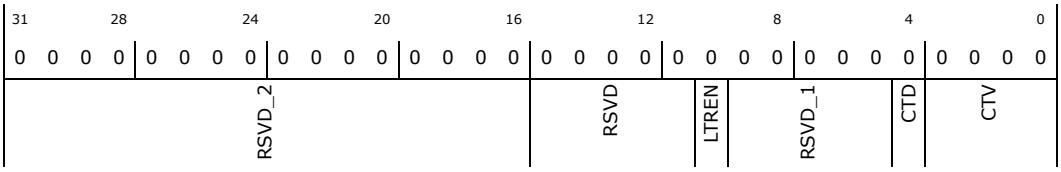
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 68h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSVD_2):** Reserved.
15:11	00h RO	**Reserved (RSVD):** Reserved.
10	0b RW	**LTR Mechanism Enable (LTREN):** When Set to 1b, this bit enables the Latency Tolerance Reporting (LTR) mechanism.
9:5	00h RO	**Reserved (RSVD_1):** Reserved.
4	0b RW	**Completion Timeout Disable (CTD):** When set to 1b, this bit disables the Completion Timeout mechanism. This field is required for all devices that support the Completion Timeout Disable Capability. Software is permitted to set or clear this bit at any time. When set, the Completion Timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued. Only the value from Port 1 (for ports 1-4) or Port 5 (for ports 5-8) is used.

Bit Range	Default & Access	Field Name (ID): Description
3:0	0h RW	**Completion Timeout Value (CTV):** In Devices that support Completion Timeout programmability, this field allows system software to modify the Completion Timeout value. This field is applicable to Root Ports, Endpoints that issue requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of requests issued on PCI Express. For all other devices this field is reserved and must be hardwired to 0000b. A Device that does not support this optional capability must hardwire this field to 0000b and is required to implement a timeout value in the range 50us to 50ms. Devices that support Completion Timeout programmability must support the values given below corresponding to the programmability ranges indicated in the Completion Timeout Values Supported field. The PCH targeted configurable ranges are listed below, along with the range allowed by the PCI Express 2.0 specification. Defined encodings: 0000b Default range: 40-50ms (spec range 50us to 50ms) Values available if Range B (10ms to 250ms) programmability range is supported: 0101b 40-50ms (spec range is 16ms to 55ms) 0110b 160-170ms (spec range is 65ms to 210ms) Values available if Range C (250ms to 4s) programmability range is supported: 1001b 400-500ms (spec range is 260ms to 900ms) 1010b 1.6-1.7s (spec range is 1s to 3.5s) Values not defined above are Reserved. Software is permitted to change the value in this field at any time. For requests already pending when the Completion Timeout Value is changed, hardware is permitted to use either the new or the old value for the outstanding requests, and is permitted to base the start time for each request either on when this value was changed or on when each request was issued.

14.5.24 Link Capability 2 (LCAP2)—Offset 6Ch

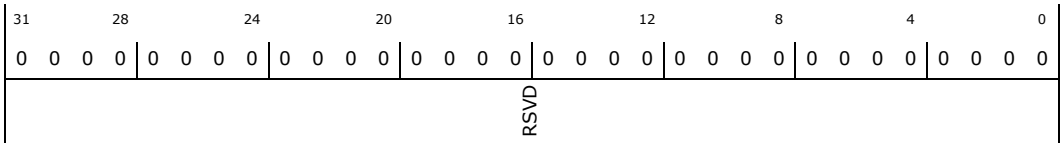
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 6Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Reserved (RSVD):** Reserved.

14.5.25 Link Status 2 (LCTL2_LSTS2)—Offset 70h

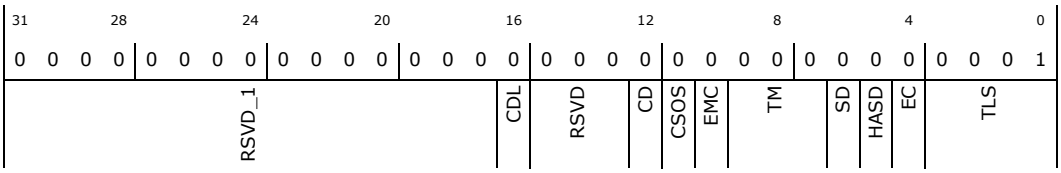
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 70h

Power Well: Core

Default: 00000001h



Bit Range	Default & Access	Field Name (ID): Description
31:17	0000h RO	**Reserved (RSVD_1):** Reserved.
16	0b RO/V	**Current De-emphasis Level (CDL):** When the Link is operating at 5 GT/s speed, this bit reflects the level of de-emphasis. Encodings: 1b -3.5 dB 0b -6 dB The value in this bit is undefined when the Link is operating at 2.5 GT/s speed.
15:13	000b RO	**Reserved (RSVD):** Reserved.
12	0b RW/P	**Compliance De-emphasis (CD):** This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 1b -3.5 dB 0b -6 dB When the Link is operating at 2.5 GT/s, the setting of this bit has no effect. The default value of this bit is 0b. This bit is intended for debug, compliance testing purposes. System firmware and software is allowed to modify this bit only during debug or compliance testing.
11	0b RW/P	**Compliance SOS (CSOS):** When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns. The default value of this bit is 0b.
10	0b RW/P	**Enter Modified Compliance (EMC):** When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate. Default value of this bit is 0b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
9:7	000b RW/P	**Transmit Margin (TM):** This field controls the value of the nondeemphasized voltage level at the Transmitter pins. This field is reset to 000b on entry to the LTSSM Polling.Configuration substate (see PCI Express Chapter 4 for details of how the Transmitter voltage level is determined in various states). Encodings: 000b Normal operating range 001b 800-1200 mV for full swing and 400-700 mV for half-swing 010b-(n-1) Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n : 200-400 mV for full-swing and 100-200 mV for half-swing n-111b reserved For a Multi-Function device associated with an Upstream Port, the field in Function 0 is of type RWS, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this field is of type RsvdP. Default value of this field is 000b. Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 000b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
6	0b RW/P	**Selectable De-emphasis (SD):** When the Link is operating at 5.0 GT/s speed, this bit selects the level of de-emphasis for an Upstream component. Encodings: 1b -3.5 dB 0b -6 dB When the Link is operating at 2.5 GT/s speed, the setting of this bit has no effect.

Bit Range	Default & Access	Field Name (ID): Description
5	0b RO	**Hardware Autonomous Speed Disable (HASD):** This port cannot autonomously change speeds.
4	0b RW/P	**Enter Compliance (EC):** Software is permitted to force a Link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link. Default value of this bit following Fundamental Reset is 0b.
3:0	1h RW/F/P	**Target Link Speed (TLS):** This field sets an upper limit on Link operational speed by restricting the values advertised by the upstream component in its training sequences.

14.5.26 Slot Capabilities 2 (SLCAP2)—Offset 74h

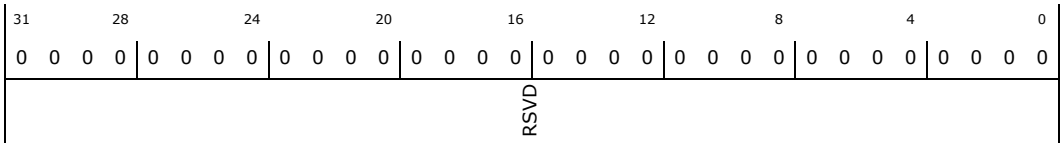
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 74h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Reserved (RSVD):** Reserved.

14.5.27 Slot Status 2 (SLCTL2_SLSTS2)—Offset 78h

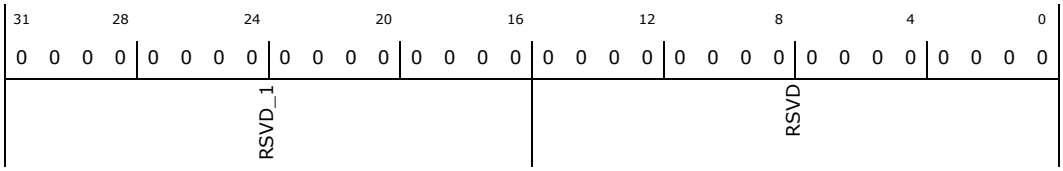
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 78h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSVD_1):** Reserved.
15:0	0000h RO	**Reserved (RSVD):** Reserved.

14.5.28 Message Signaled Interrupt Message Control (MID_MC)—Offset 80h

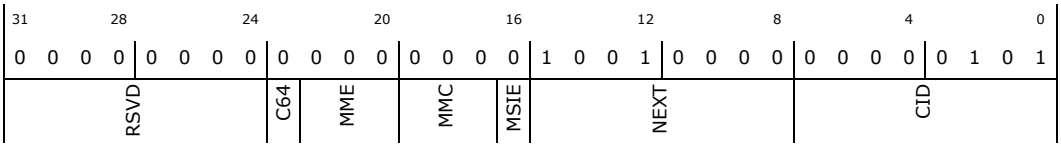
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 80h

Power Well: Core

Default: 00009005h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RO	**Reserved (RSVD):** Reserved.
23	0b RO	**64-Bit Address Capable (C64):** Capable of generating a 32-bit message only.
22:20	000b RW	**Multiple Message Enable (MME):** These bits are RW for software compatibility, but only one message is ever sent by the root port.
19:17	000b RO	**Multiple Message Capable (MMC):** Only one message is required.
16	0b RW	**MSI Enable (MSIE):** If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. CMD.BME must be set for an MSI to be generated. If CMD.BME is cleared, and this bit is set, no interrupts (not even pin based) are generated.
15:8	90h RWO	**Next Pointer (NEXT):** Indicates the location of the next capability in the list. The default value of this register is 90h which points to the Subsystem Vendor capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	05h RO	**Capability ID (CID):** Capabilities ID indicates MSI.

14.5.29 Message Signaled Interrupt Message Address (MA)—Offset 84h

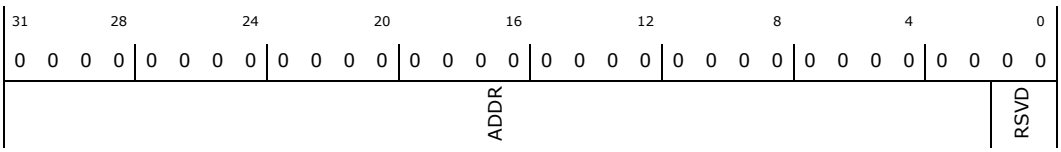
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 84h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	00000000h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DW aligned.
1:0	00b RO	**Reserved (RSVD):** Reserved.

14.5.30 Message Signaled Interrupt Message Data (MD)—Offset 88h

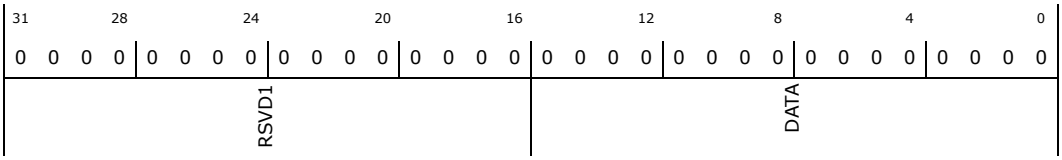
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 88h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	00000h RO	**Reserved (RSVD1):** Reserved
15:0	0000h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD[15:0]) during the data phase of the MSI memory write transaction.

14.5.31 Subsystem Vendor Capability (SVCAP)—Offset 90h

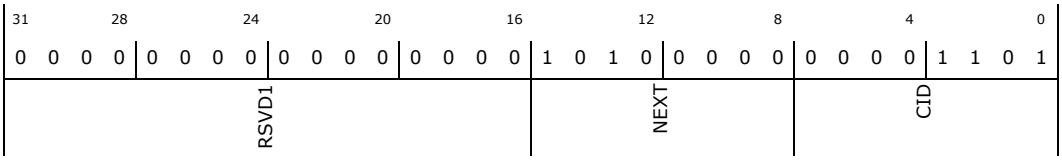
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 90h

Power Well: Core

Default: 0000A00Dh



Bit Range	Default & Access	Field Name (ID): Description
31:16	00000h RO	**Reserved (RSVD1):** Reserved

Bit Range	Default & Access	Field Name (ID): Description
15:8	A0h RWO	**Next Capability (NEXT):** Indicates the location of the next capability in the list. The default value of this register is A0h which points to the PCI Power Management capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	0Dh RO	**Capability Identifier (CID):** Value of 0Dh indicates this is a PCI bridge subsystem vendor capability.

14.5.32 Subsystem Vendor IDs (SVID)—Offset 94h

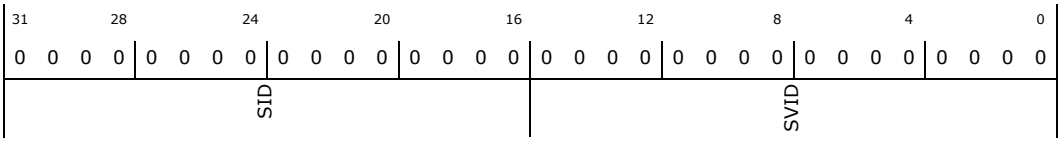
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 94h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RWO	**Subsystem Identifier (SID):** Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).
15:0	0000h RWO	**Subsystem Vendor Identifier (SVID):** Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).

14.5.33 PCI Power Management Capabilities (PMCAP_PMC)—Offset A0h

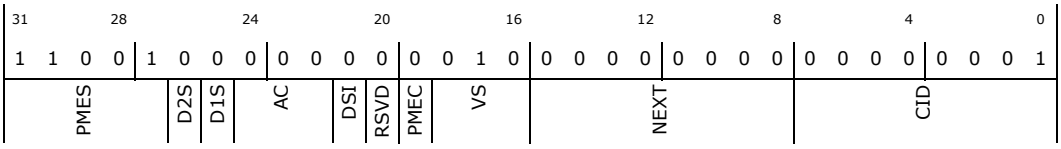
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + A0h

Power Well: Core

Default: C8020001h



Bit Range	Default & Access	Field Name (ID): Description
31:27	11001b RO	**PME Support (PMES):** Indicates PME# is supported for states D0, D3HOT and D3COLD. The root port does not generate PME#, but reporting that it does is necessary for legacy Microsoft operating systems to enable PME# in devices connected behind this root port.

Bit Range	Default & Access	Field Name (ID): Description
26	0b RO	**D2 Support (D2S):** The D2 state is not supported.
25	0b RO	**D1 Support (D1S):** The D1 state is not supported.
24:22	000b RO	**Aux Current (AC):** Reports 375mA maximum suspend well current required when in the D3COLD state.
21	0b RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
20	0b RO	**Reserved (RSVD):** Reserved.
19	0b RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
18:16	010b RO	**Version (VS):** Indicates support for Revision 1.1 of the PCI Power Management Specification.
15:8	00h RO	**Next Capability (NEXT):** Indicates this is the last item in the list.
7:0	01h RO	**Capability Identifier (CID):** Value of 01h indicates this is a PCI power management capability.

14.5.34 PCI Power Management Control And Status (PMCS)—Offset A4h

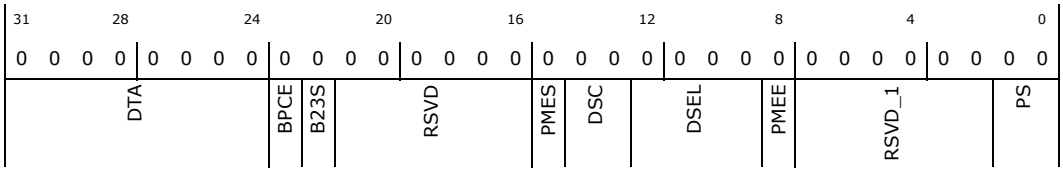
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + A4h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RO	**Data (DTA):** Reserved
23	0b RO	**Bus Power / Clock Control Enable (BPCE):** Reserved per PCI Express specification.
22	0b RO	**B2/B3 Support (B23S):** Reserved per PCI Express specification.
21:16	00h RO	**Reserved (RSVD):** Reserved.
15	0b RO	**PME Status (PMES):** Indicates a PME was received on the downstream link.
14:13	00b RO	**Data Scale (DSC):** Reserved

Bit Range	Default & Access	Field Name (ID): Description
12:9	0h RO	**Data Select (DSEL):** Reserved
8	0b RW/W	**PME Enable (PMEE):** Indicates PME is enabled. The root port takes no action on this bit, but it must be RW for legacy Microsoft operating systems to enable PME# on devices connected to this root port. This register resides in the resume well and is not reset on a resume from S3/S4/S5. The reset for this register is RSMRST# which is not asserted during a Warm Reset.
7:2	00h RO	**Reserved (RSVD_1):** Reserved.
1:0	00b RW	**Power State (PS):** This field is used both to determine the current power state of the root port and to set a new power state. The values are: 00 D0 state 11 D3HOT state When in the D3HOT state, the controller's configuration space is available, but the I/O and memory spaces are not. Type 1 configuration cycles are also not accepted. Interrupts are not required to be blocked as software will disable interrupts prior to placing the port into D3HOT. If software attempts to write a '10' or '01' to these bits, the write will be ignored.

14.5.35 Channel Configuration (CCFG)—Offset D0h

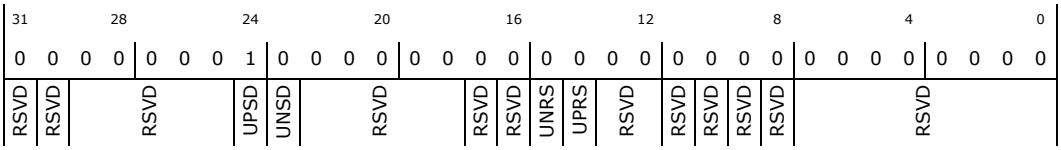
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + D0h

Power Well: Core

Default: 01000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Reserved (RSVD):** Reserved.
30	0b RO	**Reserved (RSVD):** Reserved.
29:25	00h RO	**Reserved (RSVD):** Reserved.
24	1b RW	**Upstream Posted Split Disable (UPSD):** When '0', upstream posted memory requests will be split on boundaries defined by the UPRS bit in this register. When '1', upstream posted memory requests will not be split and will be presented to the backbone as received from the link. This register has no effect on posted messages which are never split. BIOS must program this bit to '0'.
23	0b RW	**Upstream Non-Posted Split Disable (UNSD):** When '0', upstream non-posted requests will be split on boundaries defined by the UNRS bit in this register. When '1', upstream non-posted requests will not be split and will be presented to the backbone as received from the link. BIOS must program this bit to '0'.

Bit Range	Default & Access	Field Name (ID): Description
22:18	00h RO	**Reserved (RSVD):** Reserved.
17	0b RO	**Reserved (RSVD):** Reserved.
16	0b RO	**Reserved (RSVD):** Reserved.
15	0b RW	**Upstream Non-Posted Request Size (UNRS):** Sets the size for splitting upstream memory read requests. Requests will be split on naturally aligned addresses. When '0', requests are split at 128 byte boundaries. When '1', requests are split at 64 byte boundaries. This field is only used if the UNSD bit is '0'. BIOS must program this bit to '1'.
14	0b RW	**Upstream Posted Request Size (UPRS):** Sets the size for splitting upstream memory write requests. Requests will be split on naturally aligned addresses. When '0', requests are split at 128 byte boundaries. When '1', requests are split at 64 byte boundaries. This field is only used if the UPSD bit is '0'. This register has no effect on posted messages which are never split. BIOS must program this bit to '1'.
13:12	00b RO	**Reserved (RSVD):** Reserved.
11	0b RO	**Reserved (RSVD):** Reserved.
10	0b RO	**Reserved (RSVD):** Reserved.
9	0b RO	**Reserved (RSVD):** Reserved.
8	0b RO	**Reserved (RSVD):** Reserved.
7:0	00h RO	**Reserved (RSVD):** Reserved.

14.5.36 Miscellaneous Port Configuration 2 (MPC2)—Offset D4h

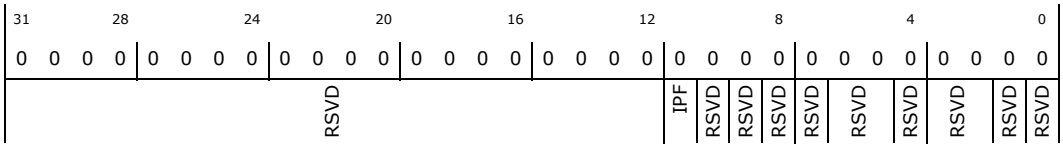
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + D4h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:12	00000h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0b RW	**Packet Fast Transmit Mode (IPF):** When set, the PCIe transmit block will move the packet header from the Tx buffer to the retry buffer without waiting for the corresponding data to be available. When cleared, the packet transfer to the retry buffer will not occur until the Tx buffer has the entire data phase available. BIOS must program this bit to 1.
10	0b RO	**Reserved (RSVD):** Reserved.
9	0b RO	**Reserved (RSVD):** Reserved.
8	0b RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Reserved (RSVD):** Reserved.
6:5	00b RO	**Reserved (RSVD):** Reserved.
4	0b RO	**Reserved (RSVD):** Reserved.
3:2	00b RO	**Reserved (RSVD):** Reserved.
1	0b RO	**Reserved (RSVD):** Reserved.
0	0b RO	**Reserved (RSVD):** Reserved.

14.5.37 Miscellaneous Port Configuration (MPC)—Offset D8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + D8h

Power Well: Core

Default: 01110000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMCE	HPCE	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	UCEL				CCEL			RSVD	RSVD	RSVD	RSVD				RSVD				RSVD	RSVD	HPME	PMME

Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Power Management SCI Enable (PMCE):** Enables the root port to generate SCI whenever a power management event is detected.
30	0b RW	**Hot Plug SCI Enable (HPCE):** Enables the root port to generate SCI whenever a hot plug event is detected.
29	0b RO	**Reserved (RSVD):** Reserved.
28	0b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
27	0b RO	**Reserved (RSVD):** Reserved.
26	0b RO	**Reserved (RSVD):** Reserved.
25	0b RO	**Reserved (RSVD):** Reserved.
24	1b RO	**Reserved (RSVD):** Reserved.
23	0b RO	**Reserved (RSVD):** Reserved.
22	0b RO	**Reserved (RSVD):** Reserved.
21	0b RO	**Reserved (RSVD):** Reserved.
20:18	100b RW	**Unique Clock Exit Latency (UCEL):** This value represents the L0s Exit Latency for unique-clock configurations (LCAP.CCC = '0'). It defaults to 512ns to less than 1us, but may be overridden by BIOS.
17:15	010b RW	**Common Clock Exit Latency (CCEL):** This value represents the L0s Exit Latency for common-clock configurations (LCAP.CCC = '1'). It defaults to 128ns to less than 256ns, but may be overridden by BIOS.
14	0b RO	**Reserved (RSVD):** Reserved.
13	0b RO	**Reserved (RSVD):** Reserved.
12	0b RO	**Reserved (RSVD):** Reserved.
11:8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Reserved (RSVD):** Reserved.
6:4	000b RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Reserved (RSVD):** Reserved.
2	0b RO	**Reserved (RSVD):** Reserved.
1	0b RW	**Hot Plug SMI Enable (HPME):** Enables the root port to generate SMI whenever a hot plug event is detected.
0	0b RW	**Power Management SMI Enable (PMME):** Enables the root port to generate SMI whenever a power management event is detected.

14.5.38 SMI / SCI Status (SMSCS)—Offset DCh

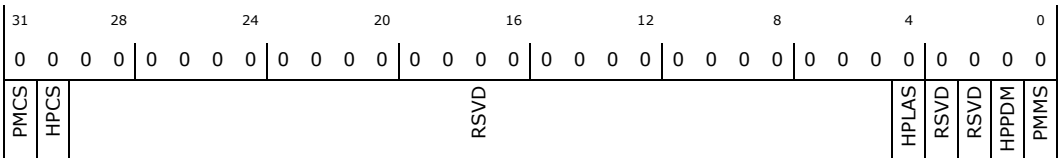
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + DCh

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RWC	**Power Management SCI Status (PMCS):** This bit is set if the root port PME control logic needs to generate an interrupt, and this interrupt has been routed to generate an SCI.
30	0b RWC	**Hot Plug SCI Status (HPCS):** This bit is set if the hot plug controller needs to generate an interrupt, and has this interrupt been routed to generate an SCI.
29:5	0000000h RO	**Reserved (RSVD):** Reserved.
4	0b RWC	**Hot Plug Link Active State Changed SMI Status (HPLAS):** This bit is set when SLSTS.LASC transitions from '0' to '1', and MPC.HPME is set. When this bit is set, an SMI# will be generated.
3	0b RO	**Reserved (RSVD):** Reserved.
2	0b RO	**Reserved (RSVD):** Reserved.
1	0b RWC	**Hot Plug Presence Detect SMI Status (HPPDM):** This bit is set when SLSTS.PDC transitions from '0' to '1', and MPC.HPME is set. When this bit is set, an SMI# will be generated.
0	0b RWC	**Power Management SMI Status (PMMS):** This bit is set when RSTS.PS transitions from '0' to '1', and MPC.PMME is set.

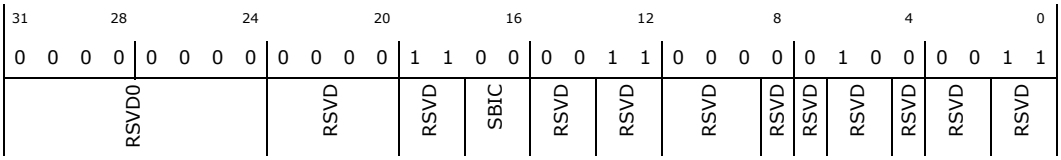
14.5.39 Message Bus Control (PHYCTL_PHYCTL2_IOSFSBCTL)—Offset F4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + F4h

Default: 000C3043h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0b RO	**RSVD0:** Reserved
23:20	0h RO	**Reserved (RSVD):** Reserved.
19:18	11b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
17:16	00b RW	**Message Bus Idle Counter (SBIC):** This register provides configuration flexibility to govern when the Message Bus interface transitions to IDLE. BIOS must program this field to 11b to prevent transitions to IDLE on the Message Bus interface.
15:14	00b RO	**Reserved (RSVD):** Reserved.
13:12	11b RO	**Reserved (RSVD):** Reserved.
11:9	000b RO	**Reserved (RSVD):** Reserved.
8	0b RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Reserved (RSVD):** Reserved.
6:5	10b RO	**Reserved (RSVD):** Reserved.
4	0b RO	**Reserved (RSVD):** Reserved.
3:2	00b RO	**Reserved (RSVD):** Reserved.
1:0	11b RO	**Reserved (RSVD):** Reserved.

14.5.40 Advanced Error Reporting Capability Header (AECH)—Offset 100h

The AER capability can optionally be included or excluded from the capabilities list. The full AER is supported.

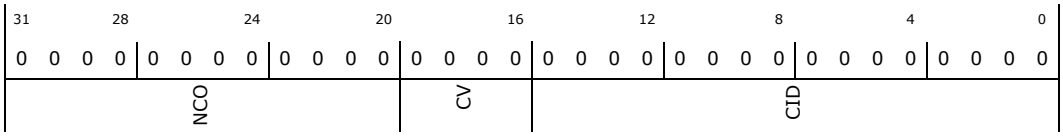
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 100h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	000h RWO	**Next Capability Offset (NCO):** Set to 000h as this is the last capability in the list.
19:16	0h RWO	**Capability Version (CV):** For systems that support AER, BIOS should write a 1h to this register else it should write 0

Bit Range	Default & Access	Field Name (ID): Description
15:0	0000h RWO	**Capability ID (CID):** For systems that support AER, BIOS should write a 0001h to this register else it should write 0

14.5.41 Uncorrectable Error Status (UES)—Offset 104h

This register must maintain its state through a platform reset. It loses its state upon loss of core power.

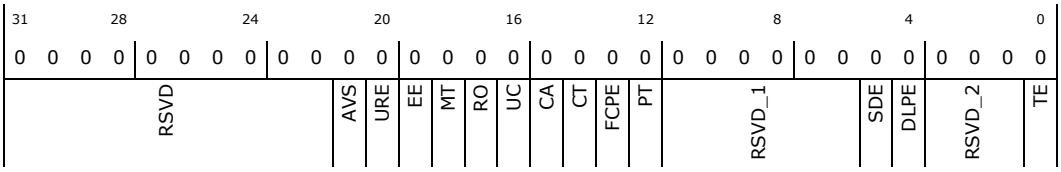
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 104h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:22	000h RO	**Reserved (RSVD):** Reserved.
21	0b RO	**ACS Violation Status (AVS):** Reserved. Access Control Services are not supported.
20	0b RWC/P	**Unsupported Request Error Status (URE):** Indicates an unsupported request was received.
19	0b RO	**ECRC Error Status (EE):** ECRC is not supported.
18	0b RWC/P	**Malformed TLP Status (MT):** Indicates a malformed TLP was received.
17	0b RWC/P	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0b RWC/P	**Unexpected Completion Status (UC):** Indicates an unexpected completion was received.
15	0b RWC/P	**Completer Abort Status (CA):** Indicates a completer abort was received.
14	0b RWC/P	**Completion Timeout Status (CT):** Indicates a completion timed out. This is signaled if Completion Timeout is enabled and a completion fails to return within the amount of time specified by the Completion Timeout Value.
13	0b RO	**Flow Control Protocol Error Status (FCPE):** Not supported.
12	0b RWC/P	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:6	00h RO	**Reserved (RSVD_1):** Reserved.
5	0b RO	**Surprise Down Error Status (SDE):** Surprise Down is not supported.

Bit Range	Default & Access	Field Name (ID): Description
4	0b RWC/P	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:1	000b RO	**Reserved (RSVD_2):** Reserved.
0	0b RO	**Training Error Status (TE):** Not supported.

14.5.42 Uncorrectable Error Mask (UEM)—Offset 108h

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

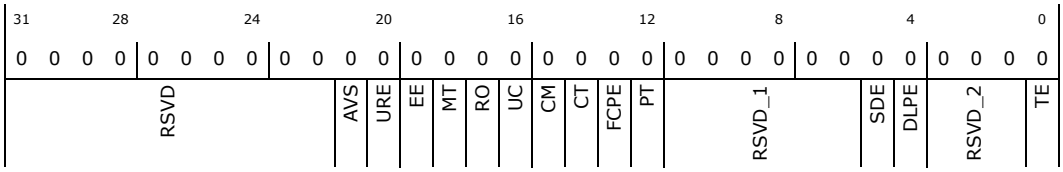
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 108h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:22	000h RO	**Reserved (RSVD):** Reserved.
21	0b RO	**ACS Violation Status (AVS):** Reserved. Access Control Services are not supported.
20	0b RW/P	**Unsupported Request Error Mask (URE):** Mask for uncorrectable errors.
19	0b RO	**ECRC Error Mask (EE):** ECRC is not supported.
18	0b RW/P	**Malformed TLP Mask (MT):** Mask for malformed TLPs.
17	0b RW/P	**Receiver Overflow Mask (RO):** Mask for receiver overflows.
16	0b RW/P	**Unexpected Completion Mask (UC):** Mask for unexpected completions.
15	0b RW/P	**Completer Abort Mask (CM):** Mask for completer abort.
14	0b RW/P	**Completion Timeout Mask (CT):** Mask for completion timeouts.
13	0b RO	**Flow Control Protocol Error Mask (FCPE):** Not supported.

Bit Range	Default & Access	Field Name (ID): Description
12	0b RW/P	**Poisoned TLP Mask (PT):** Mask for poisoned TLPs.
11:6	00h RO	**Reserved (RSVD_1):** Reserved.
5	0b RO	**Surprise Down Error Mask (SDE):** Surprise Down is not supported.
4	0b RW/P	**Data Link Protocol Error Mask (DLPE):** Mask for data link protocol errors.
3:1	000b RO	**Reserved (RSVD_2):** Reserved.
0	0b RO	**Training Error Mask (TE):** Not supported.

14.5.43 Uncorrectable Error Severity (UEV)—Offset 10Ch

This register gives the option to make an uncorrectable error fatal or non-fatal. An error is fatal if the bit is set. An error is non-fatal if the bit is cleared. This register is only reset by a loss of core power.

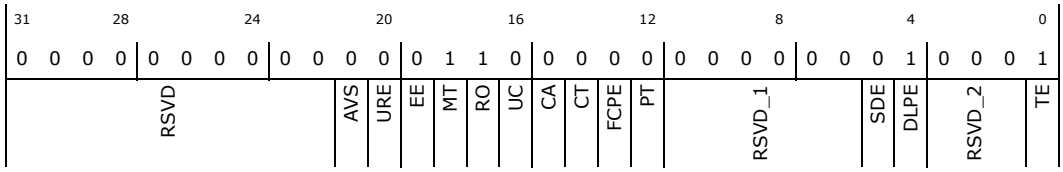
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 10Ch

Power Well: Core

Default: 00060011h



Bit Range	Default & Access	Field Name (ID): Description
31:22	000h RO	**Reserved (RSVD):** Reserved.
21	0b RO	**ACS Violation Severity (AVS):** Reserved. Access Control Services are not supported.
20	0b RW/P	**Unsupported Request Error Severity (URE):** Severity for unsupported request reception.
19	0b RO	**ECRC Error Severity (EE):** ECRC is not supported.
18	1b RW/P	**Malformed TLP Severity (MT):** Severity for malformed TLP reception.
17	1b RW/P	**Receiver Overflow Severity (RO):** Severity for receiver overflow occurrences.
16	0b RW/P	**Unexpected Completion Severity (UC):** Severity for unexpected completion reception.

Bit Range	Default & Access	Field Name (ID): Description
15	0b RW/P	**Completer Abort Severity (CA):** Severity for completer abort.
14	0b RW/P	**Completion Timeout Severity (CT):** Severity for completion timeout.
13	0b RO	**Flow Control Protocol Error Severity (FCPE):** Not supported.
12	0b RW/P	**Poisoned TLP Severity (PT):** Severity for poisoned TLP reception.
11:6	00h RO	**Reserved (RSVD_1):** Reserved.
5	0b RO	**Surprise Down Error Severity (SDE):** Surprise Down is not supported.
4	1b RW/P	**Data Link Protocol Error Severity (DLPE):** Severity for data link protocol errors.
3:1	000b RO	**Reserved (RSVD_2):** Reserved.
0	1b RO	**Training Error Severity (TE):** TE not supported. This bit is left as RO='1' for ease of implementation.

14.5.44 Correctable Error Status (CES)—Offset 110h

This register is only reset by a loss of core power.

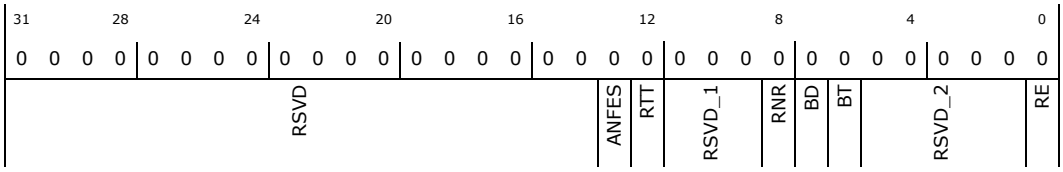
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 110h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	**Reserved (RSVD):** Reserved.
13	0b RWC/P	**Advisory Non-Fatal Error Status (ANFES):** When set, indicates that a Advisory Non-Fatal Error occurred.
12	0b RWC/P	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.
11:9	000b RO	**Reserved (RSVD_1):** Reserved.
8	0b RWC/P	**Replay Number Rollover Status (RNR):** Indicates the replay number rolled over.

Bit Range	Default & Access	Field Name (ID): Description
7	0b RWC/P	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0b RWC/P	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	00h RO	**Reserved (RSVD_2):** Reserved.
0	0b RWC/P	**Receiver Error Status (RE):** Indicates a receiver error occurred.

14.5.45 Correctable Error Mask (CEM)—Offset 114h

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

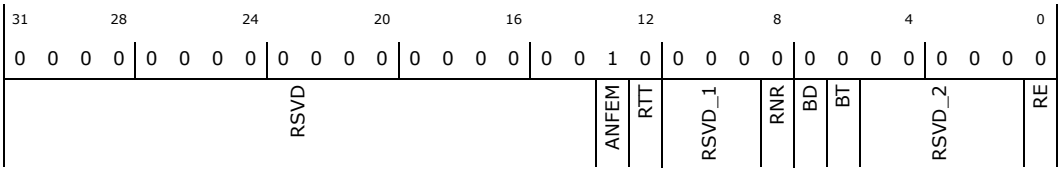
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 114h

Power Well: Core

Default: 00002000h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	**Reserved (RSVD):** Reserved.
13	1b RW/P	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0b RW/P	**Replay Timer Timeout Mask (RTT):** Mask for replay timer timeout.
11:9	000b RO	**Reserved (RSVD_1):** Reserved.
8	0b RW/P	**Replay Number Rollover Mask (RNR):** Mask for replay number rollover.
7	0b RW/P	**Bad DLLP Mask (BD):** Mask for bad DLLP reception.
6	0b RW/P	**Bad TLP Mask (BT):** Mask for bad TLP reception.
5:1	00h RO	**Reserved (RSVD_2):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
0	0b RW/P	**Receiver Error Mask (RE):** Mask for receiver errors.

14.5.46 Advanced Error Capabilities and Control (AECC)—Offset 118h

This register is only reset by a loss of core power.

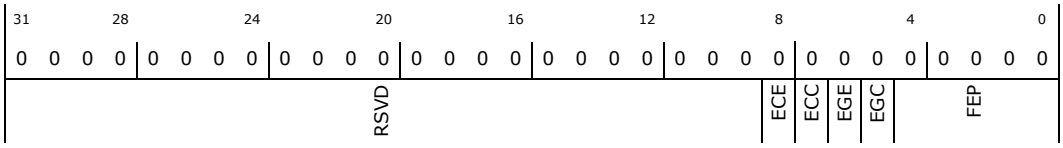
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 118h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:9	000000h RO	**Reserved (RSVD):** Reserved.
8	0b RO	**ECRC Check Enable (ECE):** ECRC is not supported.
7	0b RO	**ECRC Check Capable (ECC):** ECRC is not supported.
6	0b RO	**ECRC Generation Enable (EGE):** ECRC is not supported.
5	0b RO	**ECRC Generation Capable (EGC):** ECRC is not supported.
4:0	00000b RO/V/P	**First Error Pointer (FEP):** Identifies the bit position of the first error reported in the Uncorrectable Error Status Register.

14.5.47 Header Log (HL_DW1)—Offset 11Ch

These registers report the header for the TLP corresponding to a detected error. This register is only reset by a loss of core power.

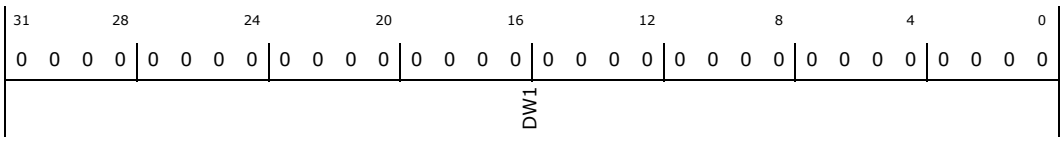
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 11Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO/V/P	**4th DWord of TLP (DW1):** Byte12 and Byte13 and Byte14 and Byte15

14.5.48 Header Log (HL_DW2)—Offset 120h

These registers report the header for the TLP corresponding to a detected error. This register is only reset by a loss of core power.

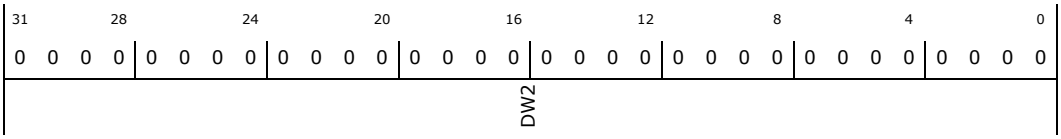
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 120h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO/V/P	**3rd DWord of TLP (DW2):** Byte8 and Byte9 and Byte10 and Byte11

14.5.49 Header Log (HL_DW3)—Offset 124h

These registers report the header for the TLP corresponding to a detected error. This register is only reset by a loss of core power.

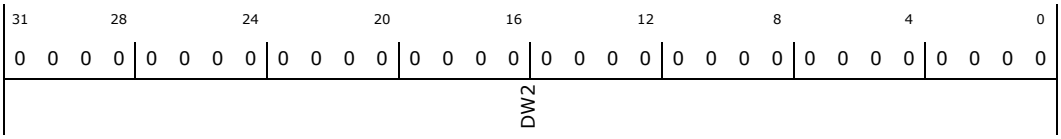
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 124h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO/V/P	**2nd DWord of TLP (DW2):** Byte4 and Byte5 and Byte6 and Byte7

14.5.50 Header Log (HL_DW4)—Offset 128h

These registers report the header for the TLP corresponding to a detected error. This register is only reset by a loss of core power.

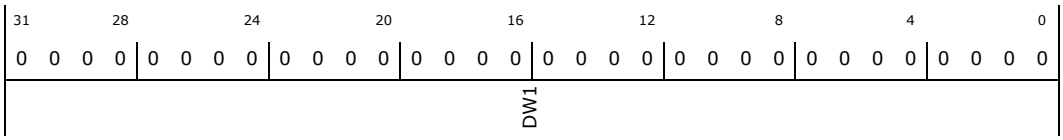
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 128h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO/V/P	**1st DWord of TLP (DW1):** Byte0 and Byte1 and Byte2 and Byte3

14.5.51 Root Error Command (REC)—Offset 12Ch

This register allows errors to generate interrupts.

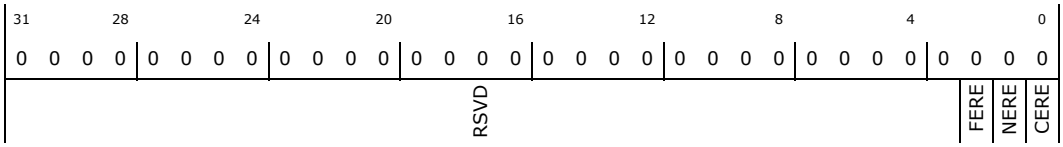
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 12Ch

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:3	00000000h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Fatal Error Reporting Enable (FERE):** When set, the root port will generate an interrupt when a fatal error is reported by the attached device.
1	0b RW	**Non-fatal Error Reporting Enable (NERE):** When set, the root port will generate an interrupt when a non-fatal error is reported by the attached device.
0	0b RW	**Correctable Error Reporting Enable (CERE):** When set, the root port will generate an interrupt when a correctable error is reported by the attached device.

14.5.52 Root Error Status (RES)—Offset 130h

This register can track more than one error and set the multiple bits if a second or subsequent error occurs and the first has not been serviced. This register is only reset by a loss of core power.

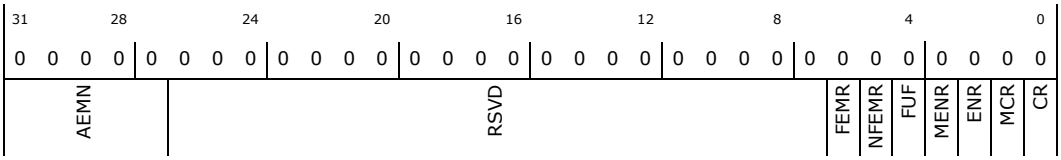
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 130h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:27	00h RO	**Advanced Error Interrupt Message Number (AEMN):** Reserved. There is only one error interrupt allocated.
26:7	00000h RO	**Reserved (RSVD):** Reserved.
6	0b RWC/P	**Fatal Error Message Received (FEMR):** Set when one or more Fatal Uncorrectable Error Messages have been received.
5	0b RWC/P	**Non-Fatal Error Messages Received (NFEMR):** Set when one or more Non-Fatal Uncorrectable error messages have been received.
4	0b RWC/P	**First Uncorrectable Fatal (FUF):** Set when the first Uncorrectable Error message received is for a fatal error.
3	0b RWC/P	**Multiple ERR_FATAL/NONFATAL Received (MENR):** Set when either a fatal or a non-fatal error is received and the ENR bit is already set.
2	0b RWC/P	**ERR_FATAL/NONFATAL Received (ENR):** Set when either a fatal or a non-fatal error message is received.
1	0b RWC/P	**Multiple ERR_COR Received (MCR):** Set when a correctable error message is received and the CR bit is already set.
0	0b RWC/P	**ERR_COR Received (CR):** Set when a correctable error message is received.

14.5.53 Error Source Identification (ESID)—Offset 134h

Identifies the source (Requester ID) of the first correctable and uncorrectable (Non-Fatal / Fatal) errors reported in the Root Error Status register. This register is only reset by a loss of core power.

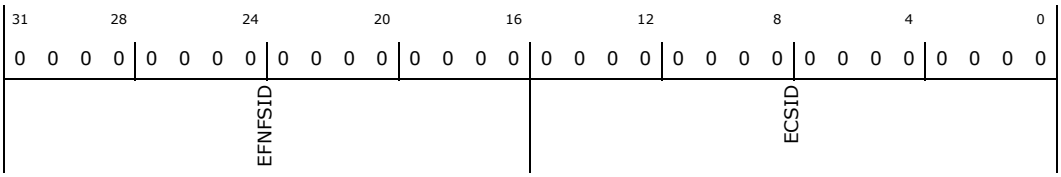
Access Method

Type: PCI Configuration Register (Size: 32 bits)

Offset: [B:0, D:23, F:0] + 134h

Power Well: Core

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO/V/P	**ERR_FATAL/NONFATAL Source Identification (EFNFSID):** Loaded with the Requester ID indicated in the received ERR_FATAL or ERR_NONFATAL Message with the ERR_FATAL/NONFATAL Received register is not already set.
15:0	0000h RO/V/P	**ERR_COR Source Identification (ECSID):** Loaded with the Requester ID indicated in the received ERR_COR Message with the ERR_COR Received register is not already set.

15.0 10/100 Mbps Ethernet

The Intel® Quark SoC X1000 provides two 10/100 Mbps Ethernet controllers. Each controller includes a MAC but not a PHY. The integrated controller is compatible with an industry standard, RMII based Ethernet PHY.

15.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 82. 10/100 Ethernet Interface Signals

Signal Name	Direction/ Type	Description
RMII_REF_CLK	I	50 MHz reference clock for the RMII interface
MAC[0/1]_TXDATA[1:0]	O	RMII Transmit data
MAC[0/1]_TXEN	O	RMII Transmit data enable
MAC[0/1]_RXDATA[1:0]	I	RMII Receive data
MAC[0/1]_RXDV	I	RMII Receive data valid
MAC[0/1]_MDC	O	Management data clock
MAC[0/1]_MDIO	I/O	Management data

15.2 Features:

- 10 and 100 Mbps data transfer rates with RMII interface to communicate with an external Fast Ethernet PHY
- Compliant with RMII specification version 1.2 from RMII consortium
- Full-duplex operation:
  - IEEE* 802.3x flow control automatic transmission of zero-quanta pause frame on flow control input de-assertion
  - Optional forwarding of received pause control frames to the user application
- Half-duplex operation:
  - CSMA/CD Protocol support
- Preamble and start-of-frame data (SFD) insertion in transmit path
- Preamble and SFD deletion in the receive path

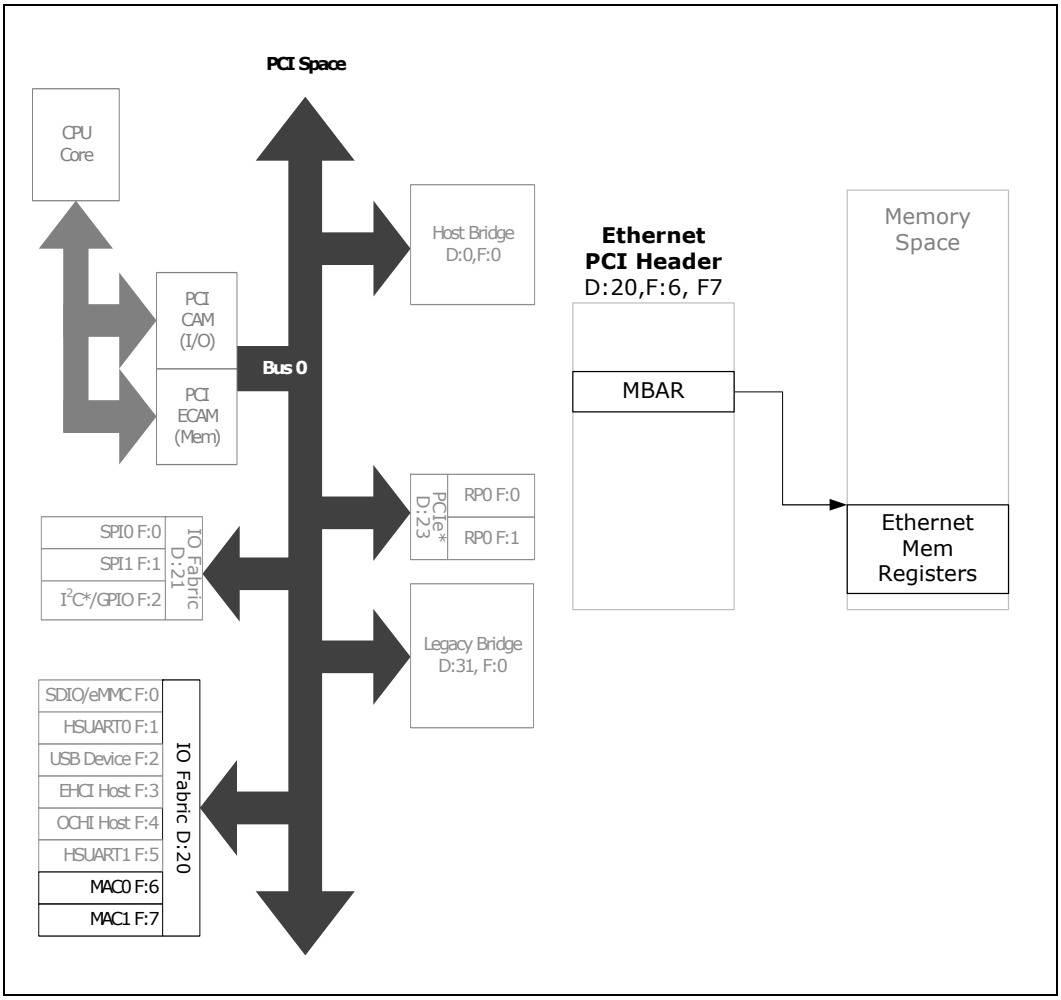
- Automatic CRC and pad generation controllable on a per-frame basis
- Automatic Pad and CRC Stripping options for receive frames
- Flexible address filtering modes:
  - 64-bit hash filter for multicast and unicast (DA) addresses
  - Option to pass all multicast addressed frames
  - Promiscuous mode to pass all frames without any filtering for network monitoring
  - Pass all incoming packets (as per filter) with a status report
- Programmable frame length.
- Programmable Interframe Gap (IFG) (40-96 bit times in steps of 8)
- Option to transmit frames with reduced preamble size
- Separate 32-bit status for transmit and receive packets
- IEEE 802.1Q VLAN tag detection for reception frames
- Additional frame filtering:
  - VLAN tag-based: hash-based filtering
- Separate transmission, reception, and control interfaces to the application
- Little-endian configuration for transmit and receive paths
- 32-bit data transfer interface on system-side
- Network statistics with RMON/MIB Counters (RFC2819/RFC2665)
- Enhanced receive module for checking IPv4 header checksum and TCP, UDP, or ICMP checksum encapsulated in IPv4 or IPv6 datagrams (Type 2)
- Support Ethernet frame time stamping as described in IEEE 1588-2002 and IEEE 1588-2008 The 64-bit timestamps are given in the transmit or receive status of each frame
- MDIO master interface for PHY device configuration and management
- CRC replacement, source address field insertion or replacement, and VLAN insertion, replacement, and deletion in transmitted frames with per-frame control

15.3 References

- IEEE 802.3TM Ethernet: http://standards.ieee.org/about/get/802/802.3.html
- Alert Standard Format Specification, Version 1.03: http://www.dmtf.org/standards/asf

15.4 Register Map

Figure 27. Ethernet Register Map



See Chapter 5.0, "Register Access Methods" for additional information.

15.5 PCI Configuration Registers

Registers listed are for Function 6 (MAC 0). Function 7 (MAC 1) contain the same registers. Differences between MACs are noted in individual registers.

Table 83. Summary of PCI Configuration Registers—0/20/6

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 310	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 311	0937h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 311	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 312	0010h

Table 83. Summary of PCI Configuration Registers—0/20/6 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 313	02000010h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 313	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 313	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 314	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 314	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 315	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 315	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 316	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 316	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 316	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 317	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 317	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 318	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 318	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 318	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 319	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 319	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 319	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 320	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 321	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 321	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 322	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 322	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 322	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 323	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 323	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 324	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 324	00000000h

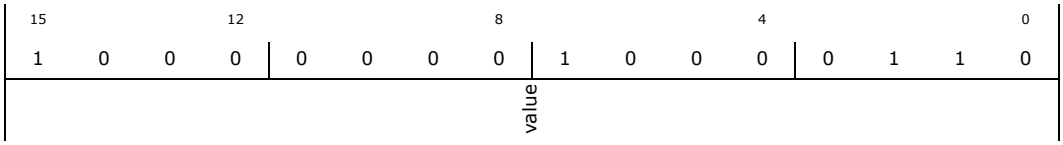
15.5.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:6] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

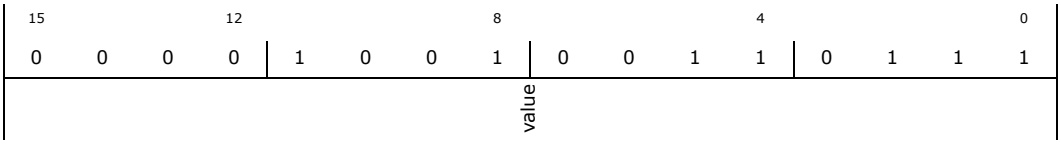
15.5.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:6] + 2h

Default: 0937h



Bit Range	Default & Access	Description
15: 0	0937h RO	**Device ID (value):** PCI Device ID

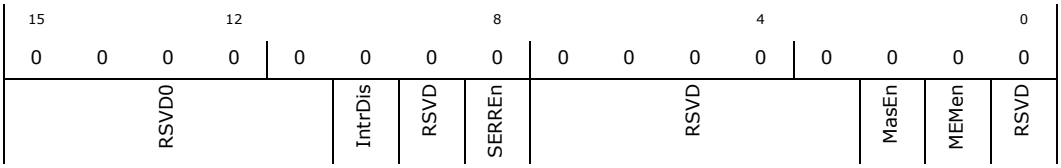
15.5.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:6] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.

Bit Range	Default & Access	Description
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

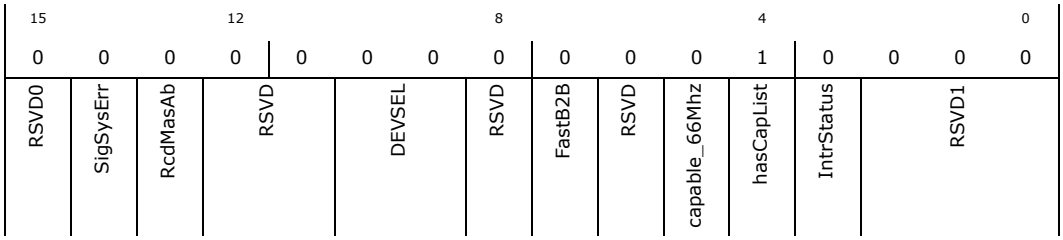
15.5.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:6] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

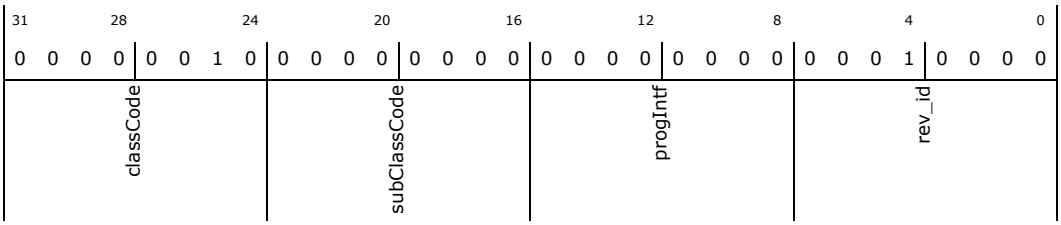
15.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:6] + 8h

Default: 02000010h



Bit Range	Default & Access	Description
31: 24	02h RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	00h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	00h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

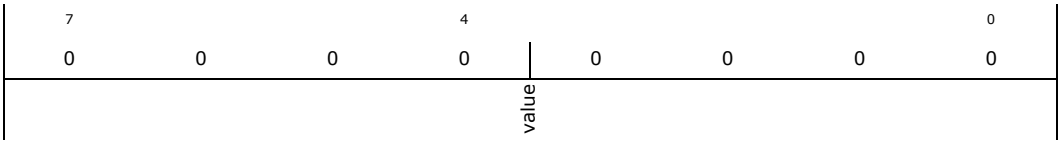
15.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:6] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

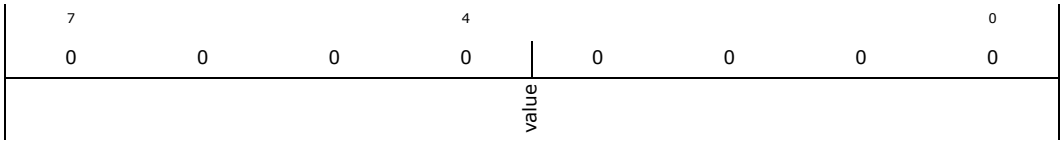
15.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:6] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

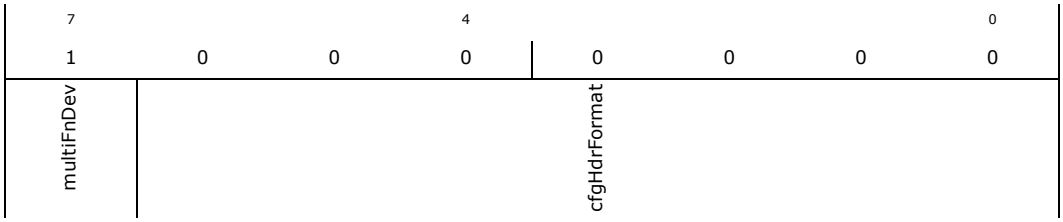
15.5.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:6] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

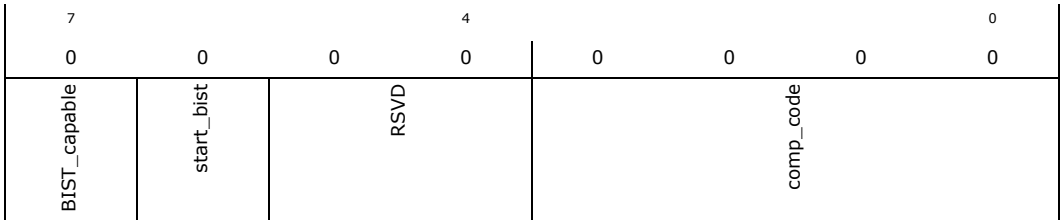
15.5.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:6] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)

Bit Range	Default & Access	Description
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

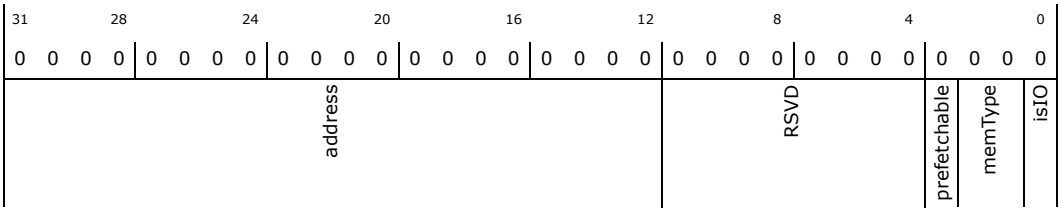
15.5.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

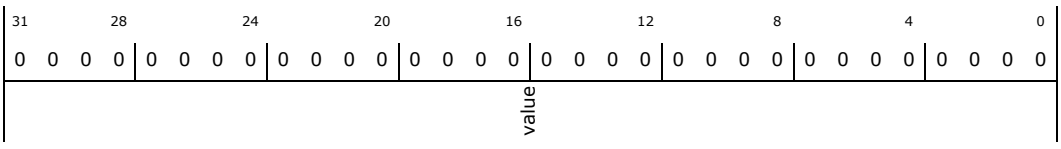
15.5.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:6] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

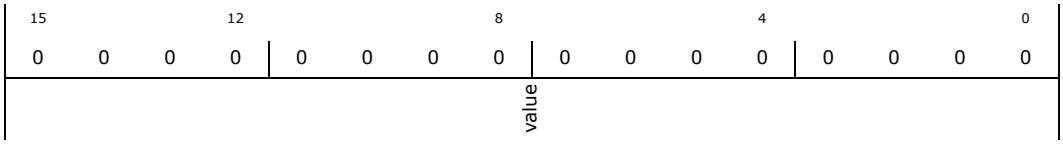
15.5.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:6] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

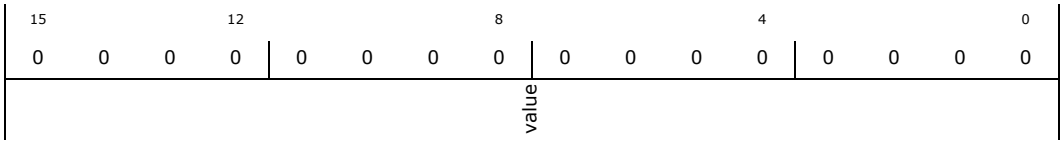
15.5.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:6] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

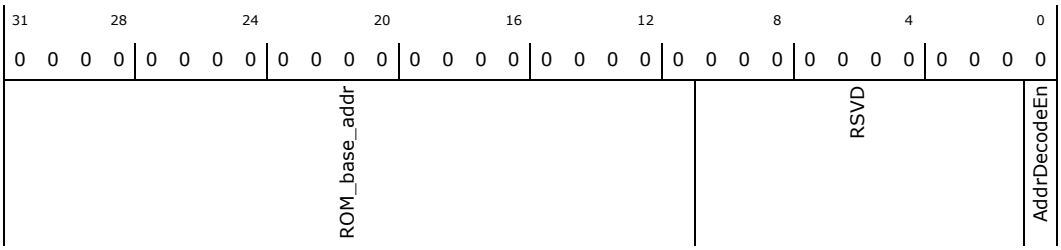
15.5.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:6] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

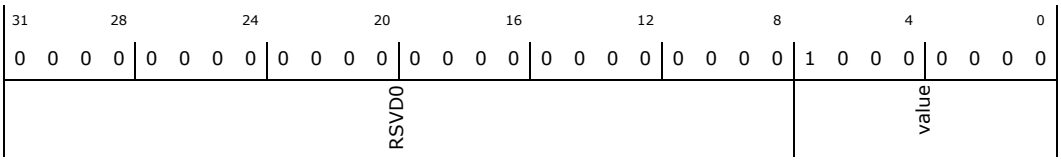
15.5.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:6] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

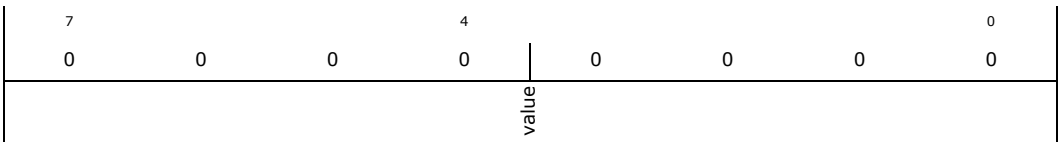
15.5.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:20, F:6] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

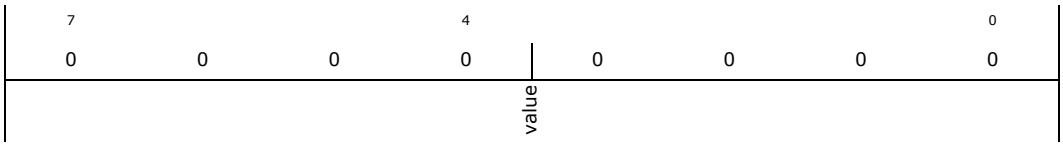
15.5.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:20, F:6] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

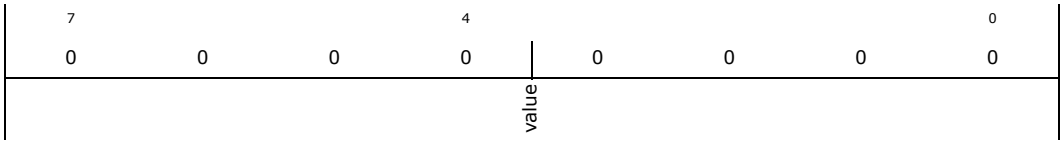
15.5.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:6] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

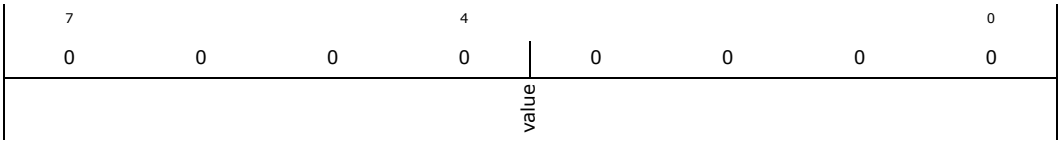
15.5.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:6] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

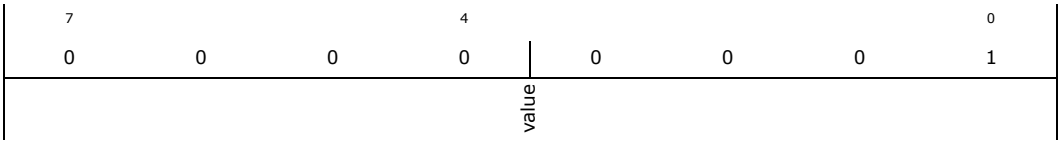
15.5.20 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:6] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

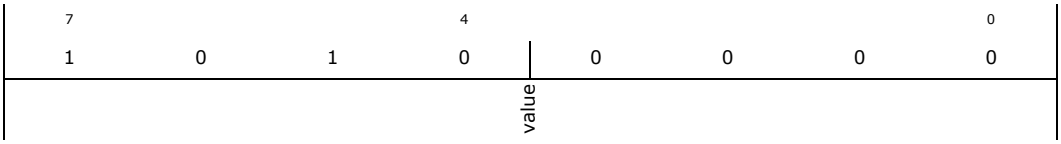
15.5.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:6] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

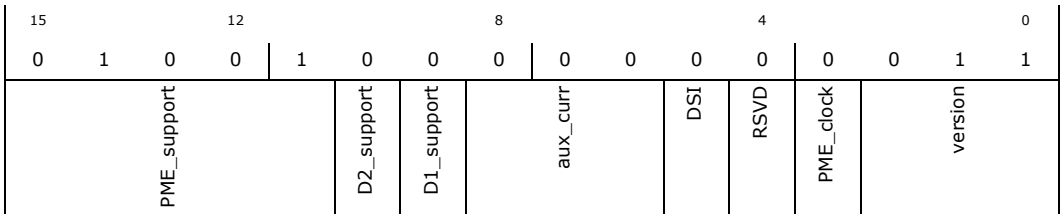
15.5.22 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:6] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

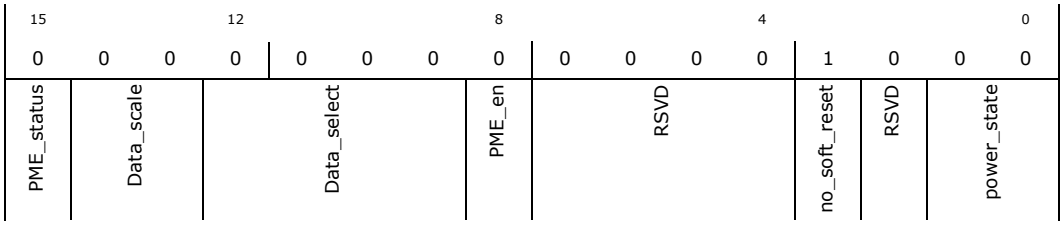
15.5.23 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:6] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported

Bit Range	Default & Access	Description
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

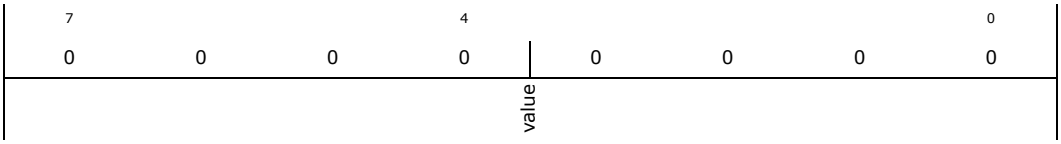
15.5.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:6] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

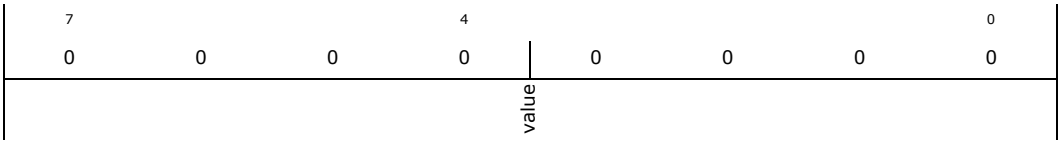
15.5.25 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:6] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

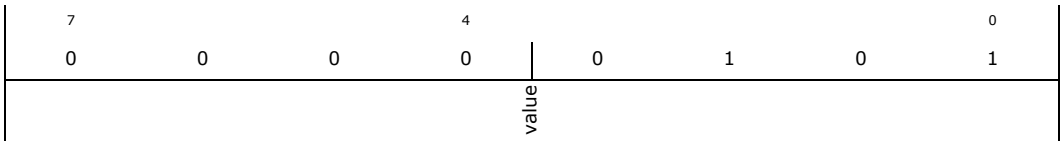
15.5.26 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:6] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

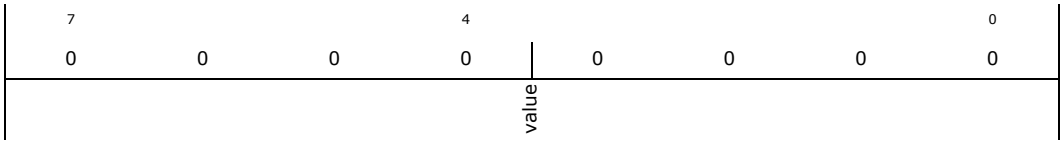
15.5.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:6] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

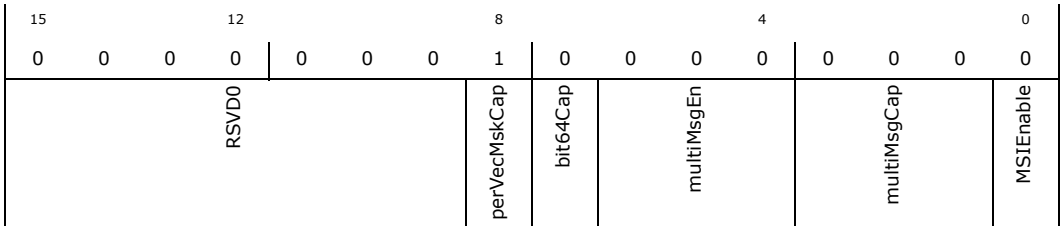
15.5.28 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:6] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

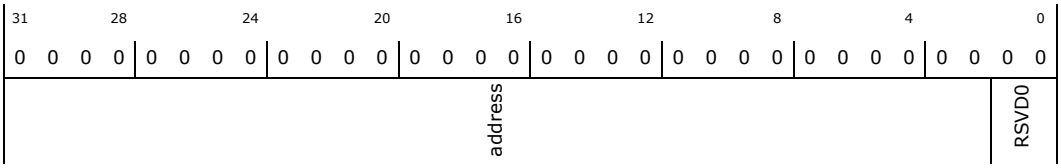
15.5.29 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:6] + A4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

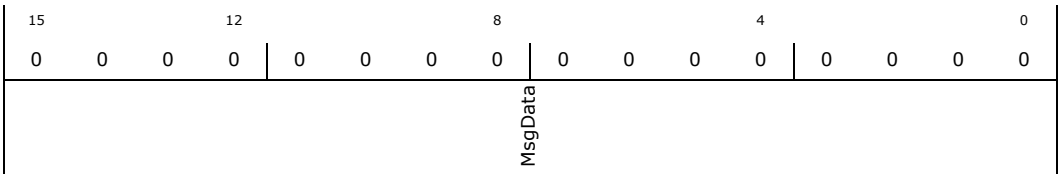
15.5.30 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:6] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

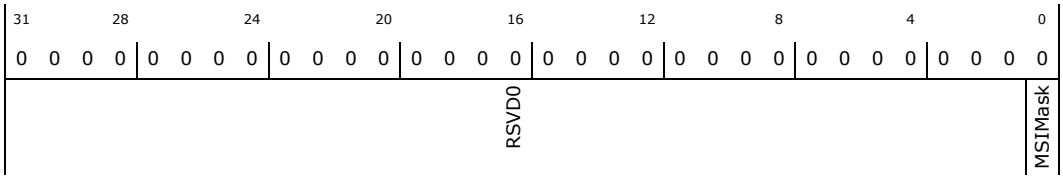
15.5.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:6] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

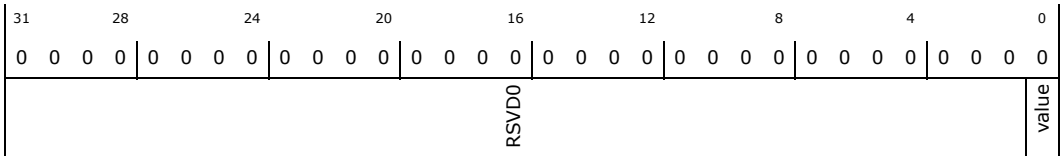
15.5.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:6] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

15.6 Memory Mapped Registers

Table 84. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"MAC Configuration Register (Register 0) (GMAC_REG_0)—Offset 0h" on page 329	00008000h
4h	7h	"MAC Frame Filter (Register 1) (GMAC_REG_1)—Offset 4h" on page 332	00000000h
8h	Bh	"Hash Table High Register (Register 2) (GMAC_REG_2)—Offset 8h" on page 334	00000000h
Ch	Fh	"Hash Table Low Register (Register 3) (GMAC_REG_3)—Offset Ch" on page 334	00000000h
10h	13h	"GMII Address Register (Register 4) (GMAC_REG_4)—Offset 10h" on page 335	00000000h
14h	17h	"GMII Data Register (Register 5) (GMAC_REG_5)—Offset 14h" on page 336	00000000h
18h	1Bh	"Flow Control Register (Register 6) (GMAC_REG_6)—Offset 18h" on page 337	00000000h
1Ch	1Fh	"VLAN Tag Register (Register 7) (GMAC_REG_7)—Offset 1Ch" on page 338	00000000h
20h	23h	"Version Register (Register 8) (GMAC_REG_8)—Offset 20h" on page 339	00001037h
24h	27h	"Debug Register (Register 9) (GMAC_REG_9)—Offset 24h" on page 340	00000000h
38h	3Bh	"Interrupt Register (Register 14) (GMAC_REG_14)—Offset 38h" on page 341	00000000h
3Ch	3Fh	"Interrupt Mask Register (Register 15) (GMAC_REG_15)—Offset 3Ch" on page 342	00000000h
40h	43h	"MAC Address0 High Register (Register 16) (GMAC_REG_16)—Offset 40h" on page 343	8000FFFFh
44h	47h	"MAC Address0 Low Register (Register 17) (GMAC_REG_17)—Offset 44h" on page 343	FFFFFFFFh
100h	103h	"MMC Control Register (Register 64) (GMAC_REG_64)—Offset 100h" on page 344	00000000h
104h	107h	"MMC Receive Interrupt Register (MMC_INTR_RX)—Offset 104h" on page 345	00000000h
108h	10Bh	"MMC Transmit Interrupt Register (MMC_INTR_TX)—Offset 108h" on page 347	00000000h
10Ch	10Fh	"MMC Receive Interrupt Mask Register (MMC_INTR_MASK_RX)—Offset 10Ch" on page 349	00000000h
110h	113h	"MMC Transmit Interrupt Mask Register (MMC_INTR_MASK_TX)—Offset 110h" on page 351	00000000h
114h	117h	"MMC Transmit Good Bad Octet Counter Register (TXOCTETCOUNT_GB)—Offset 114h" on page 353	00000000h
118h	11Bh	"MMC Transmit Good Bad Frame Counter Register (TXFRAMECOUNT_GB)—Offset 118h" on page 353	00000000h
11Ch	11Fh	"MMC Transmit Broadcast Good Frame Counter Register (TXBROADCASTFRAMES_G)—Offset 11Ch" on page 354	00000000h
120h	123h	"MMC Transmit Multicast Good Frame Counter Register (TXMULTICASTFRAMES_G)—Offset 120h" on page 354	00000000h
124h	127h	"MMC Transmit 64 Octet Good Bad Frame Counter Register (TX64OCTETS_GB)—Offset 124h" on page 355	00000000h
128h	12Bh	"MMC Transmit 65 to 127 Octet Good Bad Frame Counter Register (TX65TO127OCTETS_GB)—Offset 128h" on page 355	00000000h
12Ch	12Fh	"MMC Transmit 128 to 255 Octet Good Bad Frame Counter Register (TX128TO255OCTETS_GB)—Offset 12Ch" on page 356	00000000h
130h	133h	"MMC Transmit 256 to 511 Octet Good Bad Frame Counter Register (TX256TO511OCTETS_GB)—Offset 130h" on page 356	00000000h
134h	137h	"MMC Transmit 512 to 1023 Octet Good Bad Frame Counter Register (TX512TO1023OCTETS_GB)—Offset 134h" on page 356	00000000h
138h	13Bh	"MMC Transmit 1024 to Maximum Octet Good Bad Frame Counter Register (TX1024TOMAXOCTETS_GB)—Offset 138h" on page 357	00000000h
13Ch	13Fh	"MMC Transmit Unicast Good Bad Frame Counter Register (TXUNICASTFRAMES_GB)—Offset 13Ch" on page 357	00000000h

Table 84. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
140h	143h	"MMC Transmit Multicast Good Bad Frame Counter Register (TXMULTICASTFRAMES_GB)—Offset 140h" on page 358	00000000h
144h	147h	"MMC Transmit Broadcast Good Bad Frame Counter Register (TXBROADCASTFRAMES_GB)—Offset 144h" on page 358	00000000h
148h	14Bh	"MMC Transmit Underflow Error Frame Counter Register (TXUNDERFLOWERROR)—Offset 148h" on page 359	00000000h
14Ch	14Fh	"MMC Transmit Single Collision Good Frame Counter Register (TXSINGLECOL_G)—Offset 14Ch" on page 359	00000000h
150h	153h	"MMC Transmit Multiple Collision Good Frame Counter Register (TXMULTICOL_G)—Offset 150h" on page 360	00000000h
154h	157h	"MMC Transmit Deferred Frame Counter Register (TXDEFERRED)—Offset 154h" on page 360	00000000h
158h	15Bh	"MMC Transmit Late Collision Frame Counter Register (TXLATECOL)—Offset 158h" on page 360	00000000h
15Ch	15Fh	"MMC Transmit Excessive Collision Frame Counter Register (TXEXESSCOL)—Offset 15Ch" on page 361	00000000h
160h	163h	"MMC Transmit Carrier Error Frame Counter Register (TXCARRIERERROR)—Offset 160h" on page 361	00000000h
164h	167h	"MMC Transmit Good Bad Octet Counter Register (TXOCTETCOUNT_GB)—Offset 114h" on page 353	00000000h
168h	16Bh	"MMC Transmit Good Bad Frame Counter Register (TXFRAMECOUNT_GB)—Offset 118h" on page 353	00000000h
16Ch	16Fh	"MMC Transmit Excessive Deferral Frame Counter Register (TXEXCESSDEF)—Offset 16Ch" on page 363	00000000h
170h	173h	"MMC Transmit Pause Frame Counter Register (TXPAUSEFRAMES)—Offset 170h" on page 363	00000000h
174h	177h	"MMC Transmit VLAN Good Frame Counter Register (TXVLANFRAMES_G)—Offset 174h" on page 364	00000000h
178h	17Bh	"MMC Transmit Oversize Good Frame Counter Register (TXOVERSIZE_G)—Offset 178h" on page 364	00000000h
180h	183h	"MMC Receive Good Bad Frame Counter Register (RXFRAMECOUNT_GB)—Offset 180h" on page 364	00000000h
184h	187h	"MMC Receive Good Bad Octet Counter Register (RXOCTETCOUNT_GB)—Offset 184h" on page 365	00000000h
188h	18Bh	"MMC Receive Good Bad Octet Counter Register (RXOCTETCOUNT_GB)—Offset 184h" on page 365	00000000h
18Ch	18Fh	"MMC Receive Broadcast Good Frame Counter Register (RXBROADCASTFRAMES_G)—Offset 18Ch" on page 366	00000000h
190h	193h	"MMC Receive Multicast Good Frame Counter Register (RXMULTICASTFRAMES_G)—Offset 190h" on page 366	00000000h
194h	197h	"MMC Receive CRC Error Frame Counter Register (RXCRCERROR)—Offset 194h" on page 367	00000000h
198h	19Bh	"MMC Receive Alignment Error Frame Counter Register (RXALIGNMENTERROR)—Offset 198h" on page 367	00000000h
19Ch	19Fh	"MMC Receive Runt Frame Counter Register (RXRUNTERROR)—Offset 19Ch" on page 368	00000000h
1A0h	1A3h	"MMC Receive Jabber Error Frame Counter Register (RXJABBERERROR)—Offset 1A0h" on page 368	00000000h
1A4h	1A7h	"MMC Receive Undersize Good Frame Counter Register (RXUNDERSIZE_G)—Offset 1A4h" on page 368	00000000h

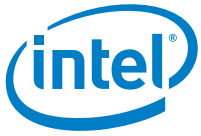


Table 84. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
1A8h	1ABh	"MMC Receive Oversize Good Frame Counter Register (RXOVERSIZE_G)—Offset 1A8h" on page 369	00000000h
1ACh	1AFh	"MMC Receive 64 Octet Good Bad Frame Counter Register (RX64OCTETS_GB)—Offset 1ACh" on page 369	00000000h
1B0h	1B3h	"MMC Receive 65 to 127 Octet Good Bad Frame Counter Register (RX65TO127OCTETS_GB)—Offset 1B0h" on page 370	00000000h
1B4h	1B7h	"MMC Receive 128 to 255 Octet Good Bad Frame Counter Register (RX128TO255OCTETS_GB)—Offset 1B4h" on page 370	00000000h
1B8h	1BBh	"MMC Receive 256 to 511 Octet Good Bad Frame Counter Register (RX256TO511OCTETS_GB)—Offset 1B8h" on page 371	00000000h
1BCh	1BFh	"MMC Receive 512 to 1023 Octet Good Bad Frame Counter Register (RX512TO1023OCTETS_GB)—Offset 1BCh" on page 371	00000000h
1C0h	1C3h	"MMC Receive 1024 to Maximum Octet Good Bad Frame Counter Register (RX1024TOMAXOCTETS_GB)—Offset 1C0h" on page 372	00000000h
1C4h	1C7h	"MMC Receive Unicast Good Frame Counter Register (RXUNICASTFRAMES_G)—Offset 1C4h" on page 372	00000000h
1C8h	1CBh	"MMC Receive Length Error Frame Counter Register (RXLENGTHERROR)—Offset 1C8h" on page 372	00000000h
1CCh	1CFh	"MMC Receive Out Of Range Error Frame Counter Register (RXOUTOFRANGETYPE)—Offset 1CCh" on page 373	00000000h
1D0h	1D3h	"MMC Receive Pause Frame Counter Register (RXPAUSEFRAMES)—Offset 1D0h" on page 373	00000000h
1D4h	1D7h	"MMC Receive FIFO Overflow Frame Counter Register (RXFIFOOVERFLOW)—Offset 1D4h" on page 374	00000000h
1D8h	1DBh	"MMC Receive VLAN Good Bad Frame Counter Register (RXVLANFRAMES_GB)—Offset 1D8h" on page 374	00000000h
1DCh	1DFh	"MMC Receive Watchdog Error Frame Counter Register (RXWATCHDOGERROR)—Offset 1DCh" on page 375	00000000h
1E0h	1E3h	"MMC Receive Error Frame Counter Register (RXRCVERROR)—Offset 1E0h" on page 375	00000000h
1E4h	1E7h	"MMC Receive Control Frame Counter Register (RXCTRLFRAMES_G)—Offset 1E4h" on page 376	00000000h
200h	203h	"MMC IPC Receive Checksum Offload Interrupt Mask Register (MMC_IPC_INTR_MASK_RX)—Offset 200h" on page 376	00000000h
208h	20Bh	"MMC Receive Checksum Offload Interrupt Register (MMC_IPC_INTR_RX)—Offset 208h" on page 378	00000000h
210h	213h	"MMC Receive IPV4 Good Frame Counter Register (RXIPV4_GD_FRMS)—Offset 210h" on page 380	00000000h
214h	217h	"MMC Receive IPV4 Header Error Frame Counter Register (RXIPV4_HDRERR_FRMS)—Offset 214h" on page 381	00000000h
218h	21Bh	"MMC Receive IPV4 No Payload Frame Counter Register (RXIPV4_NOPAY_FRMS)—Offset 218h" on page 381	00000000h
21Ch	21Fh	"MMC Receive IPV4 Fragmented Frame Counter Register (RXIPV4_FRAG_FRMS)—Offset 21Ch" on page 382	00000000h
220h	223h	"MMC Receive IPV4 UDP Checksum Disabled Frame Counter Register (RXIPV4_UDSBL_FRMS)—Offset 220h" on page 382	00000000h
224h	227h	"MMC Receive IPV6 Good Frame Counter Register (RXIPV6_GD_FRMS)—Offset 224h" on page 382	00000000h
228h	22Bh	"MMC Receive IPV6 Header Error Frame Counter Register (RXIPV6_HDRERR_FRMS)—Offset 228h" on page 383	00000000h
22Ch	22Fh	"MMC Receive IPV6 No Payload Frame Counter Register (RXIPV6_NOPAY_FRMS)—Offset 22Ch" on page 383	00000000h

Table 84. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
230h	233h	"MMC Receive UDP Good Frame Counter Register (RXUDP_GD_FRMS)—Offset 230h" on page 384	00000000h
234h	237h	"MMC Receive UDP Error Frame Counter Register (RXUDP_ERR_FRMS)—Offset 234h" on page 384	00000000h
238h	23Bh	"MMC Receive TCP Good Frame Counter Register (RXTCP_GD_FRMS)—Offset 238h" on page 385	00000000h
23Ch	23Fh	"MMC Receive TCP Error Frame Counter Register (RXTCP_ERR_FRMS)—Offset 23Ch" on page 385	00000000h
240h	243h	"MMC Receive ICMP Good Frame Counter Register (RXICMP_GD_FRMS)—Offset 240h" on page 386	00000000h
244h	247h	"MMC Receive ICMP Error Frame Counter Register (RXICMP_ERR_FRMS)—Offset 244h" on page 386	00000000h
250h	253h	"MMC Receive IPV4 Good Octet Counter Register (RXIPV4_GD_OCTETS)—Offset 250h" on page 386	00000000h
254h	257h	"MMC Receive IPV4 Header Error Octet Counter Register (RXIPV4_HDRERR_OCTETS)—Offset 254h" on page 387	00000000h
258h	25Bh	"MMC Receive IPV4 No Payload Octet Counter Register (RXIPV4_NOPAY_OCTETS)—Offset 258h" on page 387	00000000h
25Ch	25Fh	"MMC Receive IPV4 Fragmented Octet Counter Register (RXIPV4_FRAG_OCTETS)—Offset 25Ch" on page 388	00000000h
260h	263h	"MMC Receive IPV4 UDP Checksum Disabled Octet Counter Register (RXIPV4_UDSBL_OCTETS)—Offset 260h" on page 388	00000000h
264h	267h	"MMC Receive IPV6 Good Octet Counter Register (RXIPV6_GD_OCTETS)—Offset 264h" on page 389	00000000h
268h	26Bh	"MMC Receive IPV6 Good Octet Counter Register (RXIPV6_HDRERR_OCTETS)—Offset 268h" on page 389	00000000h
26Ch	26Fh	"MMC Receive IPV6 Header Error Octet Counter Register (RXIPV6_NOPAY_OCTETS)—Offset 26Ch" on page 390	00000000h
270h	273h	"MMC Receive IPV6 No Payload Octet Counter Register (RXUDP_GD_OCTETS)—Offset 270h" on page 390	00000000h
274h	277h	"MMC Receive UDP Good Octet Counter Register (RXUDP_ERR_OCTETS)—Offset 274h" on page 390	00000000h
278h	27Bh	"MMC Receive TCP Good Octet Counter Register (RXTCP_GD_OCTETS)—Offset 278h" on page 391	00000000h
27Ch	27Fh	"MMC Receive TCP Error Octet Counter Register (RXTCP_ERR_OCTETS)—Offset 27Ch" on page 391	00000000h
280h	283h	"MMC Receive ICMP Good Octet Counter Register (RXICMP_GD_OCTETS)—Offset 280h" on page 392	00000000h
284h	287h	"MMC Receive ICMP Error Octet Counter Register (RXICMP_ERR_OCTETS)—Offset 284h" on page 392	00000000h
584h	587h	"VLAN Tag Inclusion or Replacement Register (Register 353) (GMAC_REG_353)—Offset 584h" on page 393	00000000h
588h	58Bh	"VLAN Hash Table Register (Register 354) (GMAC_REG_354)—Offset 588h" on page 394	00000000h
700h	703h	"Timestamp Control Register (Register 448) (GMAC_REG_448)—Offset 700h" on page 394	00002000h
704h	707h	"Sub-Second Increment Register (Register 449) (GMAC_REG_449)—Offset 704h" on page 396	00000000h
708h	70Bh	"System Time - Seconds Register (Register 450) (GMAC_REG_450)—Offset 708h" on page 396	00000000h

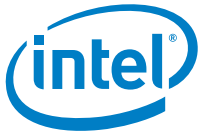


Table 84. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
70Ch	70Fh	"System Time - Nanoseconds Register (Register 451) (GMAC_REG_451)—Offset 70Ch" on page 397	00000000h
710h	713h	"System Time - Seconds Update Register (Register 452) (GMAC_REG_452)—Offset 710h" on page 397	00000000h
714h	717h	"System Time - Nanoseconds Update Register (Register 453) (GMAC_REG_453)—Offset 714h" on page 398	00000000h
718h	71Bh	"Timestamp Addend Register (Register 454) (GMAC_REG_454)—Offset 718h" on page 398	00000000h
71Ch	71Fh	"Target Time Seconds Register (Register 455) (GMAC_REG_455)—Offset 71Ch" on page 399	00000000h
720h	723h	"Target Time Nanoseconds Register (Register 456) (GMAC_REG_456)—Offset 720h" on page 399	00000000h
724h	727h	"System Time - Higher Word Seconds Register (Register 457) (GMAC_REG_457)—Offset 724h" on page 400	00000000h
728h	72Bh	"Timestamp Status Register (Register 458) (GMAC_REG_458)—Offset 728h" on page 401	00000000h
1000h	1003h	"Bus Mode Register (Register 0) (DMA_REG_0)—Offset 1000h" on page 402	00020101h
1004h	1007h	"Transmit Poll Demand Register (Register 1) (DMA_REG_1)—Offset 1004h" on page 404	00000000h
1008h	100Bh	"Receive Poll Demand Register (Register 2) (DMA_REG_2)—Offset 1008h" on page 404	00000000h
100Ch	100Fh	"Receive Descriptor List Address Register (Register 3) (DMA_REG_3)—Offset 100Ch" on page 405	00000000h
1010h	1013h	"Transmit Descriptor List Address Register (Register 4) (DMA_REG_4)—Offset 1010h" on page 405	00000000h
1014h	1017h	"Status Register (Register 5) (DMA_REG_5)—Offset 1014h" on page 406	00000000h
1018h	101Bh	"Operation Mode Register (Register 6) (DMA_REG_6)—Offset 1018h" on page 409	00000000h
101Ch	101Fh	"Interrupt Enable Register (Register 7) (DMA_REG_7)—Offset 101Ch" on page 412	00000000h
1020h	1023h	"Missed Frame and Buffer Overflow Counter Register (Register 8) (DMA_REG_8)—Offset 1020h" on page 413	00000000h
1024h	1027h	"Receive Interrupt Watchdog Timer Register (Register 9) (DMA_REG_9)—Offset 1024h" on page 414	00000000h
102Ch	102Fh	"AHB Status Register (Register 11) (DMA_REG_11)—Offset 102Ch" on page 414	00000000h
1048h	104Bh	"Current Host Transmit Descriptor Register (Register 18) (DMA_REG_18)—Offset 1048h" on page 415	00000000h
104Ch	104Fh	"Current Host Receive Descriptor Register (Register 19) (DMA_REG_19)—Offset 104Ch" on page 415	00000000h
1050h	1053h	"Current Host Transmit Buffer Address Register (Register 20) (DMA_REG_20)—Offset 1050h" on page 416	00000000h
1054h	1057h	"Current Host Receive Buffer Address Register (Register 21) (DMA_REG_21)—Offset 1054h" on page 416	00000000h
1058h	105Bh	"HW Feature Register (Register 22) (DMA_REG_22)—Offset 1058h" on page 417	4B0F3915h

15.6.1 MAC Configuration Register (Register 0) (GMAC_REG_0)—Offset 0h

The MAC Configuration register establishes receive and transmit operating modes.

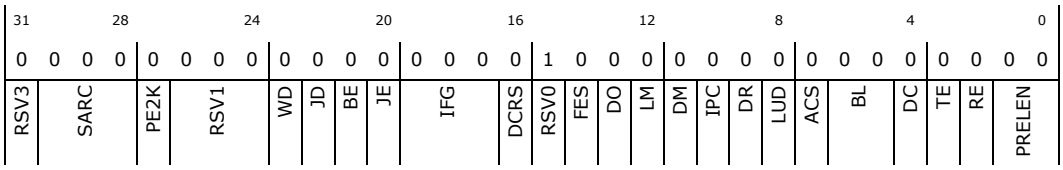
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00008000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Reserved (RSV3):** Reserved.
30:28	000b RW	**Source Address Insertion or Replacement Control (SARC):** This field controls the source address insertion or replacement for all transmitted frames. When Bit 30 is set to 0, the MAC replaces the content of the MAC Address 0 registers (registers 16 and 17) in the SA field of all transmitted frames based on the values of Bits [29:28]: - 2b0x: SA insertion is controlled by the internal signal from the MTL layer. - 2b10: if Bit 30 is set to 0, the MAC inserts the content of the MAC Address 0 registers (registers 16 and 17) in the SA field of all transmitted frames. - 2b11: if Bit 30 is set to 0, the MAC replaces the content of the MAC Address 0 registers (registers 16 and 17) in the SA field of all transmitted frames. NOTE:Changes to this field take effect only on the start of a frame. If you write this register field when a frame is being transmitted, only the subsequent frame can use the updated value, that is, the current frame does not use the updated value.
27	0b RW	**IEEE 802.3as support for 2K packets Enable (PE2K):** When set, the MAC considers all frames, with up to 2,000 bytes length, as normal packets. When Bit 20 (Jumbo Enable) is not set, the MAC considers all received frames of size more than 2K bytes as Giant frames. When this bit is reset and Bit 20 (Jumbo Enable) is not set, the MAC considers all received frames of size more than 1,518 bytes (1,522 bytes for tagged) as Giant frames. When Bit 20 (Jumbo Enable) is set, setting this bit has no effect on Giant Frame status.
26:24	000b RO	**Reserved (RSV1):** Reserved.
23	0b RW	**Watchdog Disable (WD):** When this bit is set, the MAC disables the watchdog timer on the receiver. The MAC can receive frames of up to 16,384 bytes. When this bit is reset, the MAC does not allow more than 2,048 bytes (10,240 if JE is set high) of the frame being received. The MAC cuts off any bytes received after 2,048 bytes.
22	0b RW	**Jabber Disable (JD):** When this bit is set, the MAC disables the jabber timer on the transmitter. The MAC can transfer frames of up to 16,384 bytes. When this bit is reset, the MAC cuts off the transmitter if the application sends out more than 2,048 bytes of data (10,240 if JE is set high) during transmission.
21	0b RW	**Reserved (BE):** Reserved.
20	0b RW	**Jumbo Frame Enable (JE):** When this bit is set, the MAC allows Jumbo frames of 9,018 bytes (9,022 bytes for VLAN tagged frames) without reporting a giant frame error in the receive frame status.
19:17	000b RW	**Inter-Frame Gap (IFG):** These bits control the minimum IFG between frames during transmission. 000: 96 bit times 001: 88 bit times 010: 80 bit times 111: 40 bit times In the half-duplex mode, the minimum IFG can be configured only for 64 bit times (IFG = 100). Lower values are not considered.

Bit Range	Default & Access	Field Name (ID): Description
16	0b RW	**Disable Carrier Sense During Transmission (DCRS):** When set high, this bit makes the MAC transmitter ignore the MII CRS signal during frame transmission in the half-duplex mode. This request results in no errors generated because of Loss of Carrier or No Carrier during such transmission. When this bit is low, the MAC transmitter generates such errors because of Carrier Sense and can even abort the transmissions.
15	1b RO	**GMII/MII configuration (RSV0):** This bit identifies the supported interface: 0: GMII (1000 Mbps) 1: MII (10/100 Mbps)
14	0b RW	**RMII Speed (FES):** This bit selects the speed in the RMII interface: 0: 10 Mbps 1: 100 Mbps
13	0b RW	**Disable Receive Own (DO):** When this bit is set, the MAC disables the reception of frames when the gmii_txen_o is asserted in the half-duplex mode. When this bit is reset, the MAC receives all packets that are given by the PHY while transmitting. This bit is not applicable if the MAC is operating in the full-duplex mode.
12	0b RW	**Loopback Mode (LM):** When this bit is set, the MAC operates in the loopback mode at MII.
11	0b RW	**Duplex Mode (DM):** When this bit is set, the MAC operates in the full-duplex mode where it can transmit and receive simultaneously.
10	0b RW	**Checksum Offload (IPC):** When this bit is set, the MAC calculates the 16-bit ones complement of the ones complement sum of all received Ethernet frame payloads. It also checks whether the IPv4 Header checksum (assumed to be bytes 2526 or 2930 (VLAN-tagged) of the received Ethernet frame) is correct for the received frame and gives the status in the receive status word. The MAC also appends the 16-bit checksum calculated for the IP header datagram payload (bytes after the IPv4 header) and appends it to the Ethernet frame transferred to the application (when Type 2 COE is deselected). When this bit is reset, this function is disabled. As Type 2 COE (Checksum Offload Engine) is supported, this bit, when set, enables the IPv4 header checksum checking and IPv4 or IPv6 TCP, UDP, or ICMP payload checksum checking. When this bit is reset, the COE function in the receiver is disabled and the corresponding PCE and IP HCE status bits are always cleared.
9	0b RW	**Disable Retry (DR):** When this bit is set, the MAC attempts only one transmission. When a collision occurs on the MII interface, the MAC ignores the current frame transmission and reports a Frame Abort with excessive collision error in the transmit frame status. When this bit is reset, the MAC attempts retries based on the settings of the BL field (Bits [6:5]). This bit is applicable only in the half-duplex mode.
8	0b RO	**Reserved (LUD):** Reserved.
7	0b RW	**Automatic Pad or CRC Stripping (ACS):** When this bit is set, the MAC strips the Pad or FCS (Frame Check Sequence) field on the incoming frames only if the value of the length field is less than 1,536 bytes. All received frames with length field greater than or equal to 1,536 bytes are passed to the application without stripping the Pad or FCS field. When this bit is reset, the MAC passes all incoming frames, without modifying them, to the Host.
6:5	00b RW	**Back-Off Limit (BL):** The Back-Off limit determines the random integer number (r) of slot time delays (512 bit times for 10/100 Mbps) for which the MAC waits before rescheduling a transmission attempt during retries after a collision. This bit is applicable only in the half-duplex mode. 00: k = min (n, 10) 01: k = min (n, 8) 10: k = min (n, 4) 11: k = min (n, 1) where n = retransmission attempt. The random integer r takes the value in the range 0 (= r (kth power of 2

Bit Range	Default & Access	Field Name (ID): Description
4	0b RW	**Deferral Check (DC):** When this bit is set, the deferral check function is enabled in the MAC. The MAC issues a Frame Abort status, along with the excessive deferral error bit set in the transmit frame status, when the transmit state machine is deferred for more than 24,288 bit times in the 10 or 100 Mbps mode. If the Jumbo frame mode is enabled in the 10 or 100 Mbps mode, the threshold for deferral is 155,680 bits times. Deferral begins when the transmitter is ready to transmit, but is prevented because of an active carrier sense signal (CRS) on MII. Defer time is not cumulative. When the transmitter defers for 10,000 bit times, it transmits, collides, backs off, and then defers again after completion of back-off. The deferral timer resets to 0 and restarts. When this bit is reset, the deferral check function is disabled and the MAC defers until the CRS signal goes inactive. This bit is applicable only in the half-duplex mode.
3	0b RW	**Transmitter Enable (TE):** When this bit is set, the transmit state machine of the MAC is enabled for transmission on the MII. When this bit is reset, the MAC transmit state machine is disabled after the completion of the transmission of the current frame, and does not transmit any further frames.
2	0b RW	**Receiver Enable (RE):** When this bit is set, the receiver state machine of the MAC is enabled for receiving frames from the MII. When this bit is reset, the MAC receive state machine is disabled after the completion of the reception of the current frame, and does not receive any further frames from the MII.
1:0	00b RW	**Preamble Length for Transmit Frames (PRELEN):** These bits control the number of preamble bytes that are added to the beginning of every Transmit frame. The preamble reduction occurs only when the MAC is operating in the full-duplex mode. 2'b00: 7 bytes of preamble 2'b01: 5 byte of preamble 2'b10: 3 bytes of preamble 2'b11: reserved

15.6.2 MAC Frame Filter (Register 1) (GMAC_REG_1)—Offset 4h

The MAC Frame Filter register contains the filter controls for receiving frames. Some of the controls from this register go to the address check block of the MAC, which performs the first level of address filtering. The second level of filtering is performed on the incoming frame, based on other controls such as Pass Bad Frames and Pass Control Frames.

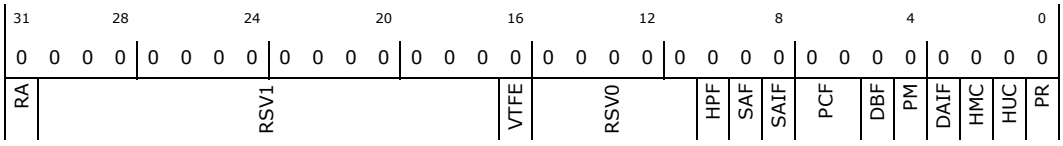
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Receive All (RA):** When this bit is set, the MAC Receiver module passes all received frames, irrespective of whether they pass the address filter or not, to the Application. The result of the SA or DA filtering is updated (pass or fail) in the corresponding bits in the Receive Status Word. When this bit is reset, the Receiver module passes only those frames to the Application that pass the SA or DA address filter.
30:17	0b RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
16	0b RW	**VLAN Tag Filter Enable (VTFE):** When set, this bit enables the MAC to drop VLAN tagged frames that do not match the VLAN Tag comparison. When reset, the MAC forwards all frames irrespective of the match status of the VLAN Tag.
15:11	00000b RO	**Reserved (RSV0):** Reserved.
10	0b RW	**Hash or Perfect Filter (HPF):** When this bit is set, it configures the address filter to pass a frame if it matches either the perfect filtering or the hash filtering as set by the HMC or HUC bits. When this bit is low and the HUC or HMC bit is set, the frame is passed only if it matches the Hash filter.
9	0b RW	**Source Address Filter Enable (SAF):** When this bit is set, the MAC compares the SA field of the received frames with the values programmed in the enabled SA registers. If the comparison matches, then the SA Match bit of RxStatus Word is set high. When this bit is set high and the SA filter fails, the MAC drops the frame. When this bit is reset, the MAC forwards the received frame to the application and with the updated SA Match bit of the RxStatus depending on the SA address comparison.
8	0b RW	**SA Inverse Filtering (SAIF):** When this bit is set, the Address Check block operates in inverse filtering mode for the SA address comparison. The frames whose SA matches the SA registers are marked as failing the SA Address filter. When this bit is reset, frames whose SA does not match the SA registers are marked as failing the SA Address filter.
7:6	00b RW	**Pass Control Frames (PCF):** These bits control the forwarding of all control frames (including unicast and multicast PAUSE frames). 00: MAC filters all control frames from reaching the application. 01: MAC forwards all control frames except PAUSE control frames to application even if they fail the Address filter. 10: MAC forwards all control frames to application even if they fail the Address Filter. 11: MAC forwards control frames that pass the Address Filter. The following conditions should be true for the PAUSE control frames processing: Condition 1: The MAC is in the full-duplex mode and flow control is enabled by setting Bit 2 (RFE) of Register 6 (Flow Control Register) to 1. Condition 2: The destination address (DA) of the received frame matches the special multicast address or the MAC Address 0 when Bit 3 (UP) of the Register 6 (Flow Control Register) is set. Condition 3: The Type field of the received frame is 0x8808 and the OPCODE field is 0x0001. NOTE: This field should be set to 01 only when the Condition 1 is true, that is, the MAC is programmed to operate in the full-duplex mode and the RFE bit is enabled. Otherwise, the PAUSE frame filtering may be inconsistent. When Condition 1 is false, the PAUSE frames are considered as generic control frames. Therefore, to pass all control frames (including PAUSE control frames) when the full-duplex mode and flow control is not enabled, you should set the PCF field to 10 or 11 (as required by the application).
5	0b RW	**Disable Broadcast Frames (DBF):** When this bit is set, the AFM module filters all incoming broadcast frames. In addition, it overrides all other filter settings. When this bit is reset, the AFM module passes all received broadcast frames.
4	0b RW	**Pass All Multicast (PM):** When set, this bit indicates that all received frames with a multicast destination address (first bit in the destination address field is '1') are passed. When reset, filtering of multicast frame depends on HMC bit.
3	0b RW	**DA Inverse Filtering (DAIF):** When this bit is set, the Address Check block operates in inverse filtering mode for the DA address comparison for both unicast and multicast frames. When reset, normal filtering of frames is performed.
2	0b RW	**Hash Multicast (HMC):** When set, MAC performs destination address filtering of received multicast frames according to the hash table. When reset, the MAC performs a perfect destination address filtering for multicast frames, that is, it compares the DA field with the values programmed in DA registers.
1	0b RW	**Hash Unicast (HUC):** When set, MAC performs destination address filtering of unicast frames according to the hash table. When reset, the MAC performs a perfect destination address filtering for unicast frames, that is, it compares the DA field with the values programmed in DA registers.

Bit Range	Default & Access	Field Name (ID): Description
0	0b RW	**Promiscuous Mode (PR):** When this bit is set, the Address Filter module passes all incoming frames regardless of its destination or source address. The SA or DA Filter Fails status bits of the Receive Status Word are always cleared when PR is set.

15.6.3 Hash Table High Register (Register 2) (GMAC_REG_2)—Offset 8h

The Hash Table High register contains the higher 32 bits of the Hash table. The 64-bit Hash table is used for group address filtering. For hash filtering, the contents of the destination address in the incoming frame is passed through the CRC logic, and the upper 6 bits of the CRC register are used to index the contents of the Hash table. The most significant bit determines the register to be used (Hash Table High or Hash Table Low), and the other 5 bits determine which bit within the register. A hash value of 5b'00000 selects Bit 0 of the selected register, and a value of 5b'11111 selects Bit 31 of the selected register. The hash value of the destination address is calculated in the following way: 1. Calculate the 32-bit CRC for the DA (See IEEE 802.3, Section 3.2.8 for the steps to calculate CRC32). 2. Perform bitwise reversal for the value obtained in Step 1. 3. Take the upper 6 bits from the value obtained in Step 2. For example, if the DA of the incoming frame is received as 0x1F52419CB6AF (0x1F is the first byte received on MII interface), then the internally calculated 6-bit Hash value is 0x2C and Bit 12 of Hash Table High register is checked for filtering. If the DA of the incoming frame is received as 0xA00A98000045, then the calculated 6-bit Hash value is 0x07 and Bit 7 of Hash Table Low register is checked for filtering. If the corresponding bit value of the register is 1'b1, the frame is accepted. Otherwise, it is rejected. If the PM (Pass All Multicast) bit is set in Register 1, then all multicast frames are accepted regardless of the multicast hash values.

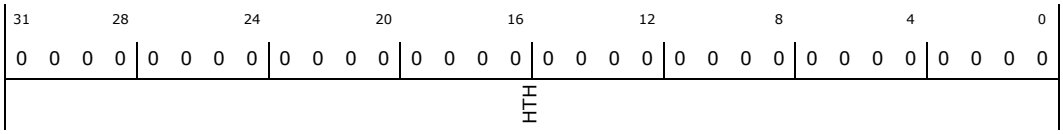
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Hash Table High (HTH):** This field contains the upper 32 bits of the Hash table.

15.6.4 Hash Table Low Register (Register 3) (GMAC_REG_3)—Offset Ch

The Hash Table Low register contains the lower 32 bits of the Hash table.

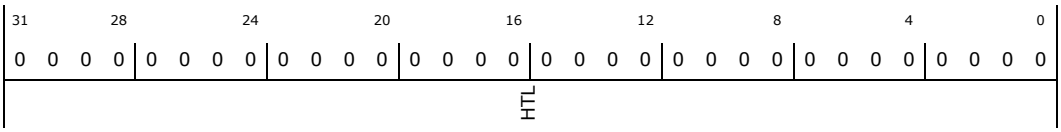
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Hash Table Low (HTL):** This field contains the lower 32 bits of the Hash table.

15.6.5 GMII Address Register (Register 4) (GMAC_REG_4)—Offset 10h

The GMII Address register controls the management cycles to the external PHY through the management interface.

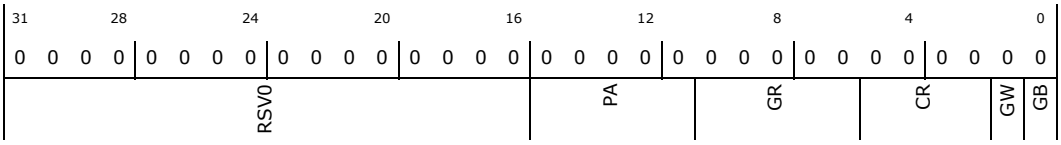
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSV0):** Reserved.
15:11	00000b RW	**Physical Layer Address (PA):** This field indicates which of the 32 possible PHY devices are being accessed.
10:6	00000b RW	**GMII Register (GR):** These bits select the desired GMII register in the selected PHY device.

Bit Range	Default & Access	Field Name (ID): Description
5:2	0000b RW	**CSR Clock Range (CR):** The CSR Clock Range selection determines the frequency of the serial management clock (MDC) according to the system clock (clk_csr_i) frequency used in your design, which is 133MHz. When Bit[5] = 0 allowed values are: - 0001: The frequency of the clk_csr_i clock is 100-150 MHz and the MDC clock is clk_csr_i/62. - 0010: The frequency of the clk_csr_i clock is 20-35 MHz and the MDC clock is clk_csr_i/16. - 0011: The frequency of the clk_csr_i clock is 35-60 MHz and the MDC clock is clk_csr_i/26. - 0100: The frequency of the clk_csr_i clock is 150-250 MHz and the MDC clock is clk_csr_i/102. - 0100: The frequency of the clk_csr_i clock is 250-300 MHz and the MDC clock is clk_csr_i/124. - 0110 and 0111: Reserved Based on a system clock of 133MHz, the CR value that ensures the MDC clock is approximately between the frequency range 1.0 MHz - 2.5 MHz is 0010 When Bit 5 is set, you can achieve MDC clock of frequency higher than the IEEE 802.3 specified frequency limit of 2.5 MHz and program a clock divider of lower value. Program the following values only if the interfacing chips support faster MDC clocks: - 1000: clk_csr_i/4 - 1001: clk_csr_i/6 - 1010: clk_csr_i/8 - 1011: clk_csr_i/10 - 1100: clk_csr_i/12 - 1101: clk_csr_i/14 - 1110: clk_csr_i/16 - 1111: clk_csr_i/18
1	0b RW	**GMII Write (GW):** When set, this bit indicates to the PHY that this is a Write operation using the GMII Data register. If this bit is not set, it indicates that this is a Read operation, that is, placing the data in the GMII Data register.
0	0b RW	**GMII Busy (GB):** This bit should read logic 0 before writing to Register 4 and Register 5. During a PHY register access, the software sets this bit to 1'b1 to indicate that a Read or Write access is in progress. The Register 5 is invalid until this bit is cleared by the MAC. Therefore, Register 5 (GMII Data) should be kept valid until the MAC clears this bit during a PHY Write operation. Similarly for a read operation, the contents of Register 5 are not valid until this bit is cleared. The subsequent read or write operation should happen only after the previous operation is complete. Because there is no acknowledgment from the PHY to MAC after a read or write operation is completed, there is no change in the functionality of this bit even when the PHY is not present.

15.6.6 GMII Data Register (Register 5) (GMAC_REG_5)—Offset 14h

The GMII Data register stores Write data to be written to the PHY register located at the address specified in Register 4 (GMII Address Register). This register also stores the Read data from the PHY register located at the address specified by Register 4.

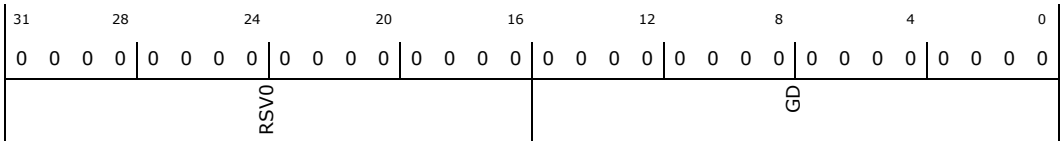
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSV0):** Reserved.
15:0	0000h RW	**GMII Data (GD):** This field contains the 16-bit data value read from the PHY after a Management Read operation or the 16-bit data value to be written to the PHY before a Management Write operation.

15.6.7 Flow Control Register (Register 6) (GMAC_REG_6)—Offset 18h

The Flow Control register controls the generation and reception of the Control (Pause Command) frames by the MAC's Flow control module. A Write to a register with the Busy bit set to '1' triggers the Flow Control block to generate a Pause Control frame. The fields of the control frame are selected as specified in the 802.3x specification, and the Pause Time value from this register is used in the Pause Time field of the control frame. The Busy bit remains set until the control frame is transferred onto the cable. The Host must make sure that the Busy bit is cleared before writing to the register.

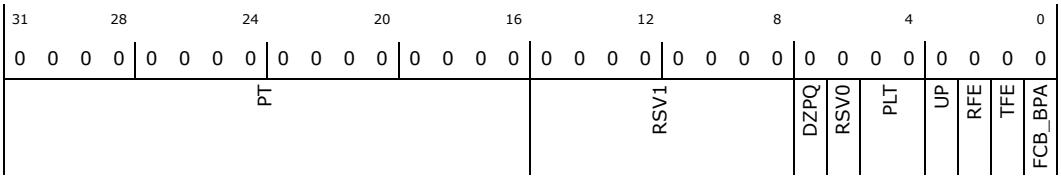
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RW	**Pause Time (PT):** This field holds the value to be used in the Pause Time field in the transmit control frame. If the Pause Time bits is configured to be double-synchronized to the MII clock domain, then consecutive writes to this register should be performed only after at least four clock cycles in the destination clock domain.
15:8	00h RO	**Reserved (RSV1):** Reserved.
7	0b RW	**Disable Zero-Quanta Pause (DZPQ):** When this bit is set, it disables the automatic generation of the Zero-Quanta Pause Control frames on the de-assertion of the flow-control signal from the FIFO layer. When this bit is reset, normal operation with automatic Zero-Quanta Pause Control frame generation is enabled.
6	0b RO	**Reserved (RSV0):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
5:4	00b RW	**Pause Low Threshold (PLT):** This field configures the threshold of the PAUSE timer at which the flow-control signal from the FIFO layer is checked for automatic retransmission of PAUSE Frame. The threshold values should be always less than the Pause Time configured in Bits[31:16]. For example, if PT = 100H (256 slot-times), and PLT = 01, then a second PAUSE frame is automatically transmitted if the flow-control signal from the FIFO layer is asserted at 228 (256 - 28) slot times after the first PAUSE frame is transmitted. The following list provides the threshold values for different values: 00: The threshold is Pause time minus 4 slot times (PT - 4 slot times). 01: The threshold is Pause time minus 28 slot times (PT - 28 slot times). 10: The threshold is Pause time minus 144 slot times (PT - 144 slot times). 11: The threshold is Pause time minus 256 slot times (PT - 256 slot times). The slot time is defined as the time taken to transmit 512 bits (64 bytes) on the MII interface.
3	0b RW	**Unicast Pause Frame Detect (UP):** When this bit is set, then in addition to the detecting Pause frames with the unique multicast address, the MAC detects the Pause frames with the station's unicast address specified in the MAC Address0 High Register and MAC Address0 Low Register. When this bit is reset, the MAC detects only a Pause frame with the unique multicast address specified in the 802.3x standard.
2	0b RW	**Receive Flow Control Enable (RFE):** When this bit is set, the MAC decodes the received Pause frame and disables its transmitter for a specified (Pause) time. When this bit is reset, the decode function of the Pause frame is disabled.
1	0b RW	**Transmit Flow Control Enable (TFE):** In the full-duplex mode, when this bit is set, the MAC enables the flow control operation to transmit Pause frames. When this bit is reset, the flow control operation in the MAC is disabled, and the MAC does not transmit any Pause frames. In half-duplex mode, when this bit is set, the MAC enables the back-pressure operation. When this bit is reset, the back-pressure feature is disabled.
0	0b RW	**Flow Control Busy or Backpressure Activate (FCB_BPA):** This bit initiates a Pause Control frame in the full-duplex mode and activates the backpressure function in the half-duplex mode if the TFE bit is set. In the full-duplex mode, this bit should be read as 1'b0 before writing to the Flow Control register. To initiate a Pause control frame, the Application must set this bit to 1'b1. During a transfer of the Control Frame, this bit continues to be set to signify that a frame transmission is in progress. After the completion of Pause control frame transmission, the MAC resets this bit to 1'b0. The Flow Control register should not be written to until this bit is cleared. In the half-duplex mode, when this bit is set (and TFE is set), then backpressure is asserted by the MAC. During backpressure, when the MAC receives a new frame, the transmitter starts sending a JAM pattern resulting in a collision. This control register bit is logically ORed with the flow-control signal from the FIFO layer for the backpressure function. When the MAC is configured for the full-duplex mode, the BPA is automatically disabled.

15.6.8 VLAN Tag Register (Register 7) (GMAC_REG_7)—Offset 1Ch

The VLAN Tag register contains the IEEE 802.1Q VLAN Tag to identify the VLAN frames. The MAC compares the 13th and 14th bytes of the receiving frame (Length/Type) with 16'h8100, and the following two bytes are compared with the VLAN tag. If a match occurs, the MAC sets the received VLAN bit in the receive frame status. The legal length of the frame is increased from 1,518 bytes to 1,522 Bytes.

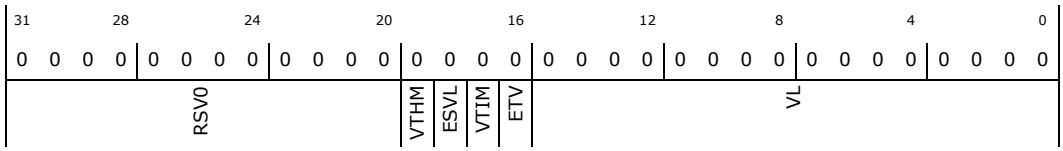
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	000h RO	**Reserved (RSV0):** Reserved.
19	0b RW	**VLAN Tag Hash Table Match Enable (VTHM):** When set, the most significant four bits of the VLAN tags CRC are used to index the content of Register 354 (VLAN Hash Table Register). A value of 1 in the VLAN Hash Table register, corresponding to the index, indicates that the frame matched the VLAN hash table. When Bit 16 (ETV) is set, the CRC of the 12-bit VLAN Identifier (VID) is used for comparison whereas when ETV is reset, the CRC of the 16-bit VLAN tag is used for comparison. When reset, the VLAN Hash Match operation is not performed.
18	0b RW	**Enable S-VLAN (ESVL):** When this bit is set, the MAC transmitter and receiver also consider the S-VLAN (Type = 0x88A8) frames as valid VLAN tagged frames.
17	0b RW	**VLAN Tag Inverse Match Enable (VTIM):** When set, this bit enables the VLAN Tag inverse matching. The frames that do not have matching VLAN Tag are marked as matched. When reset, this bit enables the VLAN Tag perfect matching. The frames with matched VLAN Tag are marked as matched.
16	0b RW	**Enable 12-Bit VLAN Tag Comparison (ETV):** When this bit is set, a 12-bit VLAN identifier is used for comparing and filtering instead of the complete 16-bit VLAN tag. Bits [11:0] of VLAN tag are compared with the corresponding field in the received VLAN-tagged frame. Similarly, when enabled, only 12 bits of the VLAN tag in the received frame are used for hash-based VLAN filtering. When this bit is reset, all 16 bits of the 15th and 16th bytes of the received VLAN frame are used for comparison and VLAN hash filtering.
15:0	0000h RW	**VLAN Tag Identifier for Receive Frames (VL):** This field contains the 802.1Q VLAN tag to identify the VLAN frames and is compared to the 15th and 16th bytes of the frames being received for VLAN frames. The following list describes the bits of this field: Bits [15:13]: User Priority Bit 12: Canonical Format Indicator (CFI) or Drop Eligible Indicator (DEI) Bits[11:0]: VLAN tag's VLAN Identifier (VID) field When the ETV bit is set, only the VID (Bits[11:0]) is used for comparison. If VL (VL[11:0] if ETV is set) is all zeros, the MAC does not check the fifteenth and 16th bytes for VLAN tag comparison, and declares all frames with a Type field value of 0x8100 or 0x88a8 as VLAN frames.

15.6.9 Version Register (Register 8) (GMAC_REG_8)—Offset 20h

The Version registers identifies the version of the MAC. This register contains two bytes: one identifies the core IP release number, and the other that identifies the user release.

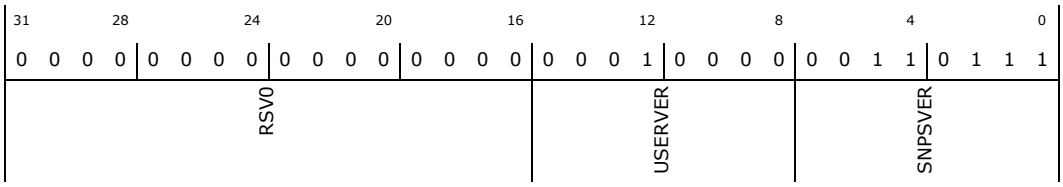
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00001037h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**RSV0:** Reserved
15:8	10h RO	**User-defined Version (1.0) (USERVER):** Reserved.
7:0	37h RO	**Synopsys-defined Version (3.7) (SNPSVER):** Reserved.

15.6.10 Debug Register (Register 9) (GMAC_REG_9)—Offset 24h

The Debug register gives the status of all main modules of the transmit and receive data-paths and the FIFOs. An all-zero status indicates that the MAC is in idle state (and FIFOs are empty) and no activity is going on in the data-paths.

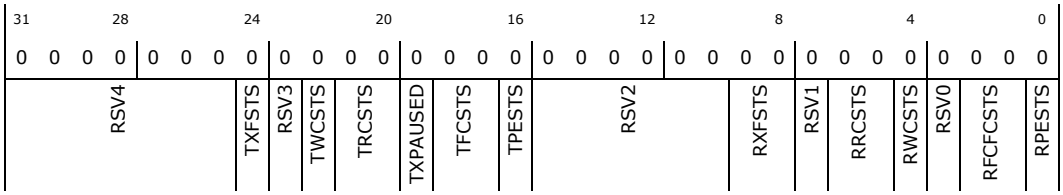
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 24h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0000000b RO	**Reserved (RSV4):** Reserved.
24	0b RO	**MTL Tx FIFO Not Empty Status (TXFSTS):** When high, this bit indicates that the MTL Tx FIFO is not empty and some data is left for transmission.
23	0b RO	**Reserved (RSV3):** Reserved.
22	0b RO	**MTL Tx FIFO Write Controller Active Status (TWCSTS):** When high, this bit indicates that the MTL Tx FIFO Write Controller is active and transferring data to the Tx FIFO.
21:20	00b RO	**MTL Tx FIFO Read Controller Status (TRCSTS):** This field indicates the state of the Tx FIFO Read Controller: 00: IDLE state 01: READ state (transferring data to MAC transmitter) 10: Waiting for TxStatus from MAC transmitter 11: Writing the received TxStatus or flushing the Tx FIFO

Bit Range	Default & Access	Field Name (ID): Description
19	0b RO	**MAC transmitter in PAUSE (TXPAUSED):** When high, this bit indicates that the MAC transmitter is in the PAUSE condition (in the full-duplex only mode) and hence does not schedule any frame for transmission.
18:17	00b RO	**MAC Transmit Frame Controller Status (TFCSTS):** This field indicates the state of the MAC Transmit Frame Controller module: 00: IDLE state 01: Waiting for Status of previous frame or IFG or backoff period to be over 10: Generating and transmitting a PAUSE control frame (in the full-duplex mode) 11: Transferring input frame for transmission
16	0b RO	**MAC MII Transmit Protocol Engine Status (TPESTS):** When high, this bit indicates that the MAC MII transmit protocol engine is actively transmitting data and is not in the IDLE state.
15:10	000000b RO	**Reserved (RSV2):** Reserved.
9:8	00b RO	**MTL Rx FIFO Fill-level Status (RXFSTS):** This field gives the status of the fill-level of the Rx FIFO: 00: Rx FIFO Empty 01: Rx FIFO fill level is below the flow-control deactivate threshold 10: Rx FIFO fill level is above the flow-control activate threshold 11: Rx FIFO Full
7	0b RO	**Reserved (RSV1):** Reserved.
6:5	00b RO	**MTL Rx FIFO Read Controller State (RRCSTS):** This field gives the state of the Rx FIFO read Controller: 00: IDLE state 01: Reading frame data 10: Reading frame status (or timestamp) 11: Flushing the frame data and status
4	0b RO	**MTL Rx FIFO Write Controller Active Status (RWCSTS):** When high, this bit indicates that the MTL Rx FIFO Write Controller is active and is transferring a received frame to the FIFO.
3	0b RO	**Reserved (RSV0):** Reserved.
2:1	00b RO	**MAC Receive Frame Controller FIFO Status (RFCFCSTS):** When high, this field indicates the active state of the small FIFO Read and Write controllers of the MAC Receive Frame Controller Module.
0	0b RO	**MAC MII Receive Protocol Engine Status (RPESTS):** When high, this bit indicates that the MAC MII receive protocol engine is actively receiving data and not in IDLE state.

15.6.11 Interrupt Register (Register 14) (GMAC_REG_14)—Offset 38h

The Interrupt Status register identifies the events in the MAC that can generate interrupt.

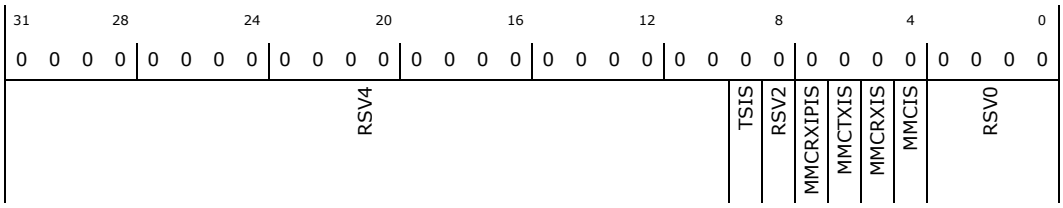
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 38h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	000000h RO	**Reserved (RSV4):** Reserved.
9	0b RO	**Timestamp Interrupt Status (TSIS):** When the Advanced Timestamp feature is enabled, this bit is set when any of the following conditions is true: The system time value equals or exceeds the value specified in the Target Time High and Low registers. There is an overflow in the seconds register. The Auxiliary snapshot trigger is asserted. This bit is cleared on reading Bit 0 of the Register 458 (Timestamp Status Register). If default Timestamping is enabled, when set, this bit indicates that the system time value is equal to or exceeds the value specified in the Target Time registers. In this mode, this bit is cleared after the completion of the read of this bit. In all other modes, this bit is reserved.
8	0b RO	**Reserved (RSV2):** Reserved.
7	0b RO	**MMC Receive Checksum Offload Interrupt Status (MMCRXIPIS):** This bit is set high when an interrupt is generated in the MMC Receive Checksum Offload Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared.
6	0b RO	**MMC Transmit Interrupt Status (MMCTXIS):** This bit is set high when an interrupt is generated in the MMC Transmit Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared.
5	0b RO	**MMC Receive Interrupt Status (MMCRXIS):** This bit is set high when an interrupt is generated in the MMC Receive Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared.
4	0b RO	**MMC Interrupt Status (MMCIS):** This bit is set high when any of the Bits [7:5] is set high and cleared only when all of these bits are low.
3:0	0000b RO	**Reserved (RSV0):** Reserved.

15.6.12 Interrupt Mask Register (Register 15) (GMAC_REG_15)—Offset 3Ch

The Interrupt Mask Register bits enable you to mask the interrupt signal because of the corresponding event in the Interrupt Status Register.

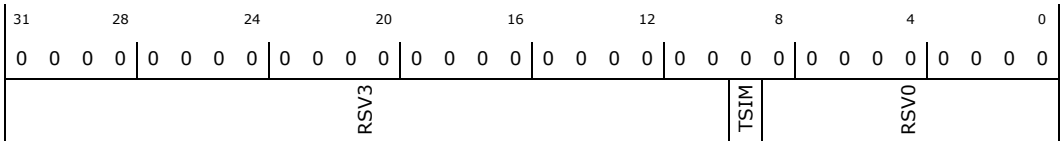
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 3Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	000000h RO	**Reserved (RSV3):** Reserved.
9	0b RW	**Timestamp Interrupt Mask (TSIM):** When set, this bit disables the assertion of the interrupt signal because of the setting of Timestamp Interrupt Status bit in Register 14 (Interrupt Status Register). This bit is valid only when IEEE1588 timestamping is enabled. In all other modes, this bit is reserved.
8:0	0b RO	**Reserved (RSV0):** Reserved.

15.6.13 MAC Address0 High Register (Register 16) (GMAC_REG_16)—Offset 40h

The MAC Address0 High register holds the upper 16 bits of the first 6-byte MAC address of the station. The first DA byte that is received on the MII interface corresponds to the LS byte (Bits [7:0]) of the MAC Address Low register. For example, if 0x112233445566 is received (0x11 in lane 0 of the first column) on the MII as the destination address, then the MacAddress0 Register [47:0] is compared with 0x665544332211. Using the standard IEEE 802 format for printing MAC-48 addresses this corresponds to 11:22:33:44:55:66 where 0x11 is the LS byte (Bits [7:0]) of the MAC Address Low register.

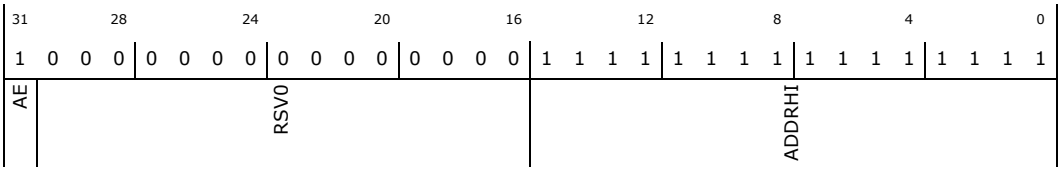
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 40h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 8000FFFFh



Bit Range	Default & Access	Field Name (ID): Description
31	1b RO	**Address Enable (AE):** This bit is always set to 1.
30:16	0000h RO	**Reserved (RSV0):** Reserved.
15:0	ffffh RW	**MAC Address0 High (ADDRHI):** This field contains the upper 16 bits (47:32) of the first 6-byte MAC address. The MAC uses this field for filtering the received frames and inserting the MAC address in the Transmit Flow Control (PAUSE) Frames.

15.6.14 MAC Address0 Low Register (Register 17) (GMAC_REG_17)—Offset 44h

The MAC Address0 Low register holds the lower 32 bits of the first 6-byte MAC address of the station.

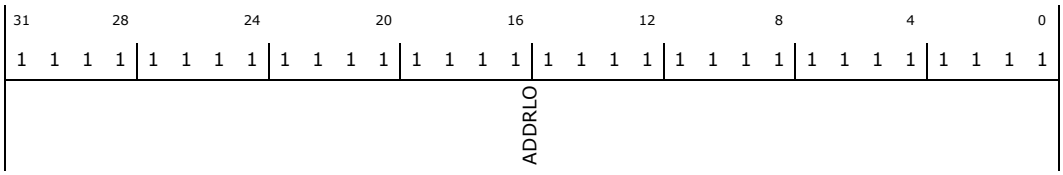
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 44h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:0	ffffffffh RW	**MAC Address0 Low (ADDRLO):** This field contains the lower 32 bits of the first 6-byte MAC address. This is used by the MAC for filtering the received frames and inserting the MAC address in the Transmit Flow Control (PAUSE) Frames.

15.6.15 MMC Control Register (Register 64) (GMAC_REG_64)—Offset 100h

The MMC Control register establishes the operating mode of the management counters. NOTE: The bit 0 (Counters Reset) has higher priority than bit 4 (Counter Preset). Therefore, when the Software tries to set both bits in the same write cycle, all counters are cleared and the bit 4 is not set.

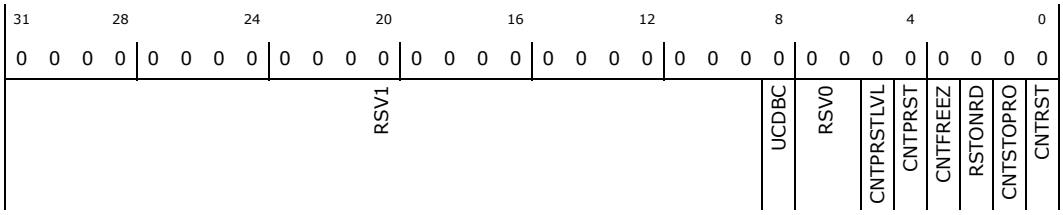
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 100h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:9	000000h RO	**Reserved (RSV1):** Reserved.
8	0b RW	**Update MMC Counters for Dropped Broadcast Frames (UCDBC):** When set, this bit enables MAC to update all the related MMC Counters for Broadcast frames dropped due to setting of DBF bit (Disable Broadcast Frames) of MAC Filter Register at offset 0x0004. When reset, MMC Counters are not updated for dropped Broadcast frames.
7:6	00b RO	**Reserved (RSV0):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
5	0b RW	**Full-Half Preset (CNTPRSTLVL):** When low and bit 4 is set, all MMC counters get preset to almost-half value. All octet counters get preset to 0x7FFF_F800 (half - 2KBytes) and all frame-counters gets preset to 0x7FFF_FFF0 (half - 16). When this bit is high and bit 4 is set, all MMC counters get preset to almost-full value. All octet counters get preset to 0xFFFF_F800 (full - 2KBytes) and all frame-counters gets preset to 0xFFFF_FFF0 (full - 16). For 16-bit counters, the almost-half preset values are 0x7800 and 0x7FF0 for the respective octet and frame counters. Similarly, the almost-full preset values for the 16-bit counters are 0xF800 and 0xFFF0.
4	0b RW	**Counters Preset (CNTPRST):** When this bit is set, all counters are initialized or preset to almost full or almost half according to bit 5. This bit is cleared automatically after 1 clock cycle. This bit, along with bit 5, is useful for debugging and testing the assertion of interrupts because of MMC counter becoming half-full or full.
3	0b RW	**MMC Counter Freeze (CNTFREEZ):** When this bit is set, it freezes all MMC counters to their current value. Until this bit is reset to 0, no MMC counter is updated because of any transmitted or received frame. If any MMC counter is read with the Reset on Read bit set, then that counter is also cleared in this mode
2	0b RW	**Reset on Read (RSTONRD):** When this bit is set, the MMC counters are reset to zero after Read (self-clearing after reset). The counters are cleared when the least significant byte lane (bits[7:0]) is read.
1	0b RW	**Counters Stop Rollover (CNTSTOPRO):** When this bit is set, after reaching maximum value, the counter does not roll over to zero.
0	0b RW	**Counters Reset (CNTRST):** When this bit is set, all counters are reset. This bit is cleared automatically after one clock cycle.

15.6.16 MMC Receive Interrupt Register (MMC_INTR_RX)—Offset 104h

The MMC Receive Interrupt Register maintains the interrupt generated from all of the receive statistic counters. An interrupt bit is cleared when the respective MMC counter that caused the interrupt is read.

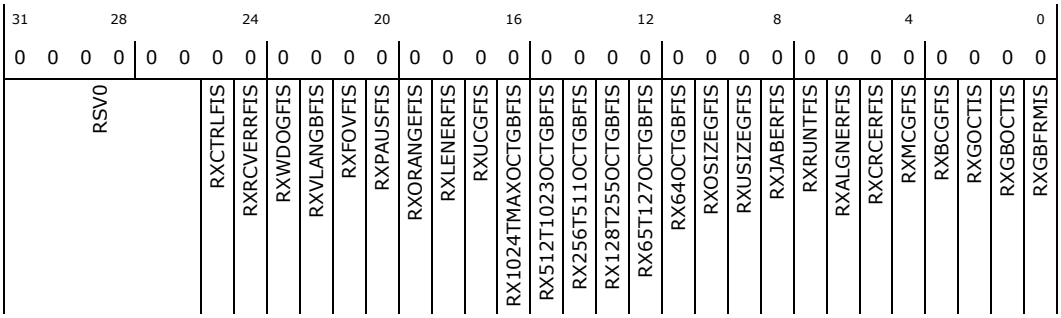
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 104h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:26	000000b RO	**Reserved (RSV0):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
25	0b RO	**MMC Receive Control Frame Counter Interrupt Status (RXCTRLFIS):** This bit is set when the rxctrlframes_g counter reaches half of the maximum value or the maximum value.
24	0b RO	**MMC Receive Error Frame Counter Interrupt Status (RXRCVERRFIS):** This bit is set when the rxrcverror counter reaches half of the maximum value or the maximum value.
23	0b RO	**MMC Receive Watchdog Error Frame Counter Interrupt Status (RXWDOGFIS):** This bit is set when the rxwatchdog error counter reaches half of the maximum value or the maximum value.
22	0b RO	**MMC Receive VLAN Good Bad Frame Counter Interrupt Status (RXVLANGBFIS):** This bit is set when the rxvlanframes_gb counter reaches half of the maximum value or the maximum value.
21	0b RO	**MMC Receive FIFO Overflow Frame Counter Interrupt Status (RXFOVFIS):** This bit is set when the rxfifooverflow counter reaches half of the maximum value or the maximum value.
20	0b RO	**MMC Receive Pause Frame Counter Interrupt Status (RXPAUSFIS):** This bit is set when the rxpauseframes counter reaches half of the maximum value or the maximum value.
19	0b RO	**MMC Receive Out Of Range Error Frame Counter Interrupt Status (RXORANGEFIS):** This bit is set when the rxoutofrangetype counter reaches half of the maximum value or the maximum value.
18	0b RO	**MMC Receive Length Error Frame Counter Interrupt Status (RXLENERFIS):** This bit is set when the rxlengtherror counter reaches half of the maximum value or the maximum value.
17	0b RO	**MMC Receive Unicast Good Frame Counter Interrupt Status (RXUCGFIS):** This bit is set when the rxunicastframes_g counter reaches half of the maximum value or the maximum value.
16	0b RO	**MMC Receive 1024 to Maximum Octet Good Bad Frame Counter Interrupt Status (RX1024TMAXOCTGBFIS):** This bit is set when the rx1024tomaxoctets_gb counter reaches half of the maximum value or the maximum value.
15	0b RO	**MMC Receive 512 to 1023 Octet Good Bad Frame Counter Interrupt Status (RX512T1023OCTGBFIS):** This bit is set when the rx512to1023octets_gb counter reaches half of the maximum value or the maximum value.
14	0b RO	**MMC Receive 256 to 511 Octet Good Bad Frame Counter Interrupt Status (RX256T511OCTGBFIS):** This bit is set when the rx256to511octets_gb counter reaches half of the maximum value or the maximum value.
13	0b RO	**MMC Receive 128 to 255 Octet Good Bad Frame Counter Interrupt Status (RX128T255OCTGBFIS):** This bit is set when the rx128to255octets_gb counter reaches half of the maximum value or the maximum value.
12	0b RO	**MMC Receive 65 to 127 Octet Good Bad Frame Counter Interrupt Status (RX65T127OCTGBFIS):** This bit is set when the rx65to127octets_gb counter reaches half of the maximum value or the maximum value.
11	0b RO	**MMC Receive 64 Octet Good Bad Frame Counter Interrupt Status (RX64OCTGBFIS):** This bit is set when the rx64octets_gb counter reaches half of the maximum value or the maximum value.
10	0b RO	**MMC Receive Oversize Good Frame Counter Interrupt Status (RXOSIZEGFIS):** This bit is set when the rxoversize_g counter reaches half of the maximum value or the maximum value.
9	0b RO	**MMC Receive Undersize Good Frame Counter Interrupt Status (RXUSIZEGFIS):** This bit is set when the rxundersize_g counter reaches half of the maximum value or the maximum value.
8	0b RO	**MMC Receive Jabber Error Frame Counter Interrupt Status (RXJABERFIS):** This bit is set when the rxjabbererror counter reaches half of the maximum value or the maximum value.
7	0b RO	**MMC Receive Runt Frame Counter Interrupt Status (RXRUNTFIS):** This bit is set when the rxrunterror counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
6	0b RO	**MMC Receive Alignment Error Frame Counter Interrupt Status (RXALGNERFIS):** This bit is set when the rxalignmenterror counter reaches half of the maximum value or the maximum value.
5	0b RO	**MMC Receive CRC Error Frame Counter Interrupt Status (RXCRCERFIS):** This bit is set when the rxcrcerror counter reaches half of the maximum value or the maximum value.
4	0b RO	**MMC Receive Multicast Good Frame Counter Interrupt Status (RXMCGFIS):** This bit is set when the rxmulticastframes_g counter reaches half of the maximum value or the maximum value.
3	0b RO	**MMC Receive Broadcast Good Frame Counter Interrupt Status (RXBCGFIS):** This bit is set when the rxbroadcastframes_g counter reaches half of the maximum value or the maximum value.
2	0b RO	**MMC Receive Good Octet Counter Interrupt Status (RXGOCTIS):** This bit is set when the rxoctetcount_g counter reaches half of the maximum value or the maximum value.
1	0b RO	**MMC Receive Good Bad Octet Counter Interrupt Status (RXGBOCTIS):** This bit is set when the rxoctetcount_gb counter reaches half of the maximum value or the maximum value.
0	0b RO	**MMC Receive Good Bad Frame Counter Interrupt Status (RXGBFRMIS):** This bit is set when the rxframecount_gb counter reaches half of the maximum value or the maximum value.

15.6.17 MMC Transmit Interrupt Register (MMC_INTR_TX)—Offset 108h

The maintains the interrupt generated from all of the transmit statistic counters. An interrupt bit is cleared when the respective MMC counter that caused the interrupt is read.

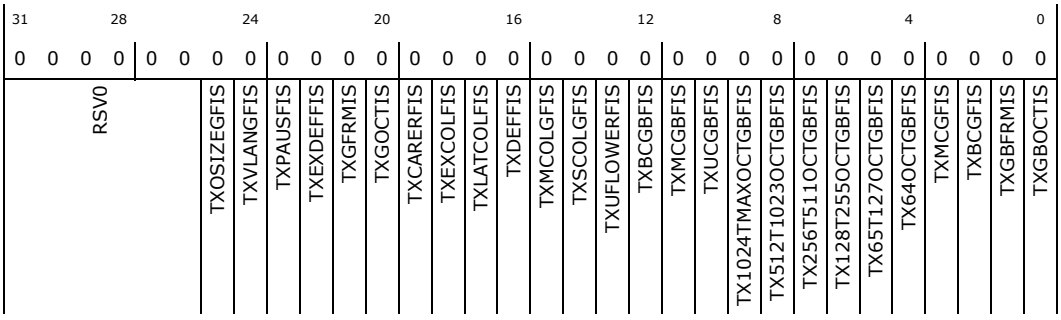
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 108h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:26	000000b RO	**Reserved (RSV0):** Reserved.
25	0b RO	**MMC Transmit Oversize Good Frame Counter Interrupt Status (TXOSIZEGFIS):** This bit is set when the txoversize_g counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
24	0b RO	**MMC Transmit VLAN Good Frame Counter Interrupt Status (TXVLANGFIS):** This bit is set when the txvlanframes_g counter reaches half of the maximum value or the maximum value.
23	0b RO	**MMC Transmit Pause Frame Counter Interrupt Status (TXPAUSFIS):** This bit is set when the txpauseframeserror counter reaches half of the maximum value or the maximum value.
22	0b RO	**MMC Transmit Excessive Deferral Frame Counter Interrupt Status (TXEXDEFFIS):** This bit is set when the txexcessdef counter reaches half of the maximum value or the maximum value.
21	0b RO	**MMC Transmit Good Frame Counter Interrupt Status (TXGFRMIS):** This bit is set when the txframecount_g counter reaches half of the maximum value or the maximum value.
20	0b RO	**MMC Transmit Good Octet Counter Interrupt Status (TXGOCTIS):** This bit is set when the txoctetcount_g counter reaches half of the maximum value or the maximum value.
19	0b RO	**MMC Transmit Carrier Error Frame Counter Interrupt Status (TXCARERFIS):** This bit is set when the txcarriererror counter reaches half of the maximum value or the maximum value.
18	0b RO	**MMC Transmit Excessive Collision Frame Counter Interrupt Status (TXEXCOLFIS):** This bit is set when the txexesscol counter reaches half of the maximum value or the maximum value.
17	0b RO	**MMC Transmit Late Collision Frame Counter Interrupt Status (TXLATCOLFIS):** This bit is set when the txlatecol counter reaches half of the maximum value or the maximum value.
16	0b RO	**MMC Transmit Deferred Frame Counter Interrupt Status (TXDEFFIS):** This bit is set when the txdeferred counter reaches half of the maximum value or the maximum value.
15	0b RO	**MMC Transmit Multiple Collision Good Frame Counter Interrupt Status (TXMCOLGFIS):** This bit is set when the txmulticol_g counter reaches half of the maximum value or the maximum value.
14	0b RO	**MMC Transmit Single Collision Good Frame Counter Interrupt Status (TXSCOLGFIS):** This bit is set when the txsinglecol_g counter reaches half of the maximum value or the maximum value.
13	0b RO	**MMC Transmit Underflow Error Frame Counter Interrupt Status (TXUFLOWERFIS):** This bit is set when the txunderflowerror counter reaches half of the maximum value or the maximum value.
12	0b RO	**MMC Transmit Broadcast Good Bad Frame Counter Interrupt Status (TXBCGBFIS):** This bit is set when the txbroadcastframes_gb counter reaches half of the maximum value or the maximum value.
11	0b RO	**MMC Transmit Multicast Good Bad Frame Counter Interrupt Status (TXMCGBFIS):** The bit is set when the txmulticastframes_gb counter reaches half of the maximum value or the maximum value.
10	0b RO	**MMC Transmit Unicast Good Bad Frame Counter Interrupt Status (TXUCGBFIS):** This bit is set when the txunicastframes_gb counter reaches half of the maximum value or the maximum value.
9	0b RO	**MMC Transmit 1024 to Maximum Octet Good Bad Frame Counter Interrupt Status (TX1024TMAXOCTGBFIS):** This bit is set when the tx1024tomaxoctets_gb counter reaches half of the maximum value or the maximum value.
8	0b RO	**MMC Transmit 512 to 1023 Octet Good Bad Frame Counter Interrupt Status (TX512T1023OCTGBFIS):** This bit is set when the tx512to1023octets_gb counter reaches half of the maximum value or the maximum value.
7	0b RO	**MMC Transmit 256 to 511 Octet Good Bad Frame Counter Interrupt Status (TX256T511OCTGBFIS):** This bit is set when the tx256to511octets_gb counter reaches half of the maximum value or the maximum value.
6	0b RO	**MMC Transmit 128 to 255 Octet Good Bad Frame Counter Interrupt Status (TX128T255OCTGBFIS):** This bit is set when the tx128to255octets_gb counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
5	0b RO	**MMC Transmit 65 to 127 Octet Good Bad Frame Counter Interrupt Status (TX65T127OCTGBFIS):** This bit is set when the tx65to127octets_gb counter reaches half the maximum value, and also when it reaches the maximum value.
4	0b RO	**MMC Transmit 64 Octet Good Bad Frame Counter Interrupt Status (TX64OCTGBFIS):** This bit is set when the tx64octets_gb counter reaches half of the maximum value or the maximum value.
3	0b RO	**MMC Transmit Multicast Good Frame Counter Interrupt Status (TXMCGFIS):** This bit is set when the txmulticastframes_g counter reaches half of the maximum value or the maximum value.
2	0b RO	**MMC Transmit Broadcast Good Frame Counter Interrupt Status (TXBCGFIS):** This bit is set when the txbroadcastframes_g counter reaches half of the maximum value or the maximum value.
1	0b RO	**MMC Transmit Good Bad Frame Counter Interrupt Status (TXGBFRMIS):** This bit is set when the txframecount_gb counter reaches half of the maximum value or the maximum value.
0	0b RO	**MMC Transmit Good Bad Octet Counter Interrupt Status (TXGBOCTIS):** This bit is set when the txoctetcount_gb counter reaches half of the maximum value or the maximum value.

15.6.18 MMC Receive Interrupt Mask Register (MMC_INTR_MASK_RX)—Offset 10Ch

The MMC Receive Interrupt Mask Register maintains the mask for the interrupt generated from all of the receive statistic counters.

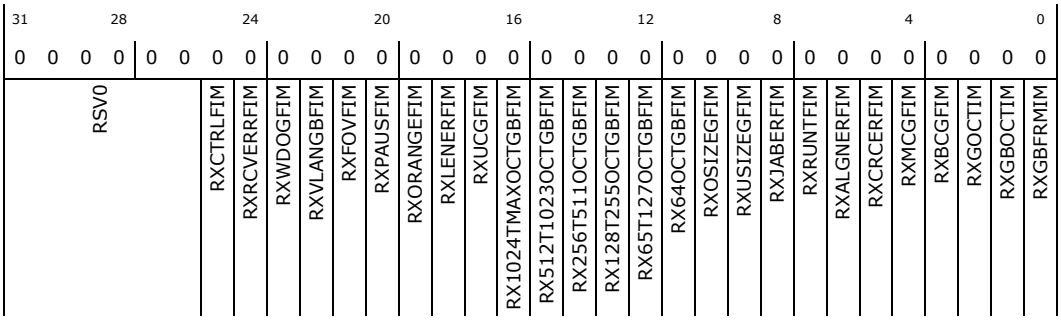
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 10Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:26	000000b RO	**Reserved (RSV0):** Reserved.
25	0b RW	**MMC Receive Control Frame Counter Interrupt Mask (RXCTRLFIM):** Setting this bit masks the interrupt when the rxctrlframes_g counter reaches half of the maximum value or the maximum value.
24	0b RW	**MMC Receive Error Frame Counter Interrupt Mask (RXRCVERRFIM):** Setting this bit masks the interrupt when the rxrcverror counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
23	0b RW	**MMC Receive Watchdog Error Frame Counter Interrupt Mask (RXWDOGFIM):** Setting this bit masks the interrupt when the rxwatchdog counter reaches half of the maximum value or the maximum value.
22	0b RW	**MMC Receive VLAN Good Bad Frame Counter Interrupt Mask (RXVLANGBFIM):** Setting this bit masks the interrupt when the rxvlanframes_gb counter reaches half of the maximum value or the maximum value.
21	0b RW	**MMC Receive FIFO Overflow Frame Counter Interrupt Mask (RXFOVFIM):** Setting this bit masks the interrupt when the rxfifooverflow counter reaches half of the maximum value or the maximum value.
20	0b RW	**MMC Receive Pause Frame Counter Interrupt Mask (RXPAUSFIM):** Setting this bit masks the interrupt when the rxpauseframes counter reaches half of the maximum value or the maximum value.
19	0b RW	**MMC Receive Out Of Range Error Frame Counter Interrupt Mask (RXORANGEFIM):** Setting this bit masks the interrupt when the rxoutofrangetype counter reaches half of the maximum value or the maximum value.
18	0b RW	**MMC Receive Length Error Frame Counter Interrupt Mask (RXLENERFIM):** Setting this bit masks the interrupt when the rxlengtherror counter reaches half of the maximum value or the maximum value.
17	0b RW	**MMC Receive Unicast Good Frame Counter Interrupt Mask (RXUCGFIM):** Setting this bit masks the interrupt when the rxunicastframes_g counter reaches half of the maximum value or the maximum value.
16	0b RW	**MMC Receive 1024 to Maximum Octet Good Bad Frame Counter Interrupt Mask (RX1024TMAXOCTGBFIM):** Setting this bit masks the interrupt when the rx1024tomaxoctets_gb counter reaches half of the maximum value or the maximum value.
15	0b RW	**MMC Receive 512 to 1023 Octet Good Bad Frame Counter Interrupt Mask (RX512T1023OCTGBFIM):** Setting this bit masks the interrupt when the rx512to1023octets_gb counter reaches half of the maximum value or the maximum value.
14	0b RW	**MMC Receive 256 to 511 Octet Good Bad Frame Counter Interrupt Mask (RX256T511OCTGBFIM):** Setting this bit masks the interrupt when the rx256to511octets_gb counter reaches half of the maximum value or the maximum value.
13	0b RW	**MMC Receive 128 to 255 Octet Good Bad Frame Counter Interrupt Mask (RX128T255OCTGBFIM):** Setting this bit masks the interrupt when the rx128to255octets_gb counter reaches half of the maximum value or the maximum value.
12	0b RW	**MMC Receive 65 to 127 Octet Good Bad Frame Counter Interrupt Mask (RX65T127OCTGBFIM):** Setting this bit masks the interrupt when the rx65to127octets_gb counter reaches half of the maximum value or the maximum value.
11	0b RW	**MMC Receive 64 Octet Good Bad Frame Counter Interrupt Mask (RX64OCTGBFIM):** Setting this bit masks the interrupt when the rx64octets_gb counter reaches half of the maximum value or the maximum value.
10	0b RW	**MMC Receive Oversize Good Frame Counter Interrupt Mask (RXOSIZEGFIM):** Setting this bit masks the interrupt when the rxoversize_g counter reaches half of the maximum value or the maximum value.
9	0b RW	**MMC Receive Undersize Good Frame Counter Interrupt Mask (RXUSIZEGFIM):** Setting this bit masks the interrupt when the rxundersize_g counter reaches half of the maximum value or the maximum value.
8	0b RW	**MMC Receive Jabber Error Frame Counter Interrupt Mask (RXJABERFIM):** Setting this bit masks the interrupt when the rxjabbererror counter reaches half of the maximum value or the maximum value.
7	0b RW	**MMC Receive Runt Frame Counter Interrupt Mask (RXRUNTFIM):** Setting this bit masks the interrupt when the rxrunterror counter reaches half of the maximum value or the maximum value.
6	0b RW	**MMC Receive Alignment Error Frame Counter Interrupt Mask (RXALGNERFIM):** Setting this bit masks the interrupt when the rxalignmenterror counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
5	0b RW	**MMC Receive CRC Error Frame Counter Interrupt Mask (RXCRCERFIM):** Setting this bit masks the interrupt when the rxcrcerror counter reaches half of the maximum value or the maximum value.
4	0b RW	**MMC Receive Multicast Good Frame Counter Interrupt Mask (RXMCGFIM):** Setting this bit masks the interrupt when the rxmulticastframes_g counter reaches half of the maximum value or the maximum value.
3	0b RW	**MMC Receive Broadcast Good Frame Counter Interrupt Mask (RXBCGFIM):** Setting this bit masks the interrupt when the rxbroadcastframes_g counter reaches half of the maximum value or the maximum value.
2	0b RW	**MMC Receive Good Octet Counter Interrupt Mask (RXGOCTIM):** Setting this bit masks the interrupt when the rxoctetcount_g counter reaches half of the maximum value or the maximum value.
1	0b RW	**MMC Receive Good Bad Octet Counter Interrupt Mask (RXGBOCTIM):** Setting this bit masks the interrupt when the rxoctetcount_gb counter reaches half of the maximum value or the maximum value.
0	0b RW	**MMC Receive Good Bad Frame Counter Interrupt Mask (RXGBFRMIM):** Setting this bit masks the interrupt when the rxframecount_gb counter reaches half of the maximum value or the maximum value.

15.6.19 MMC Transmit Interrupt Mask Register (MMC_INTR_MASK_TX)—Offset 110h

The MMC Transmit Interrupt Mask Register maintains the mask for the interrupt generated from all of the transmit statistic counters.

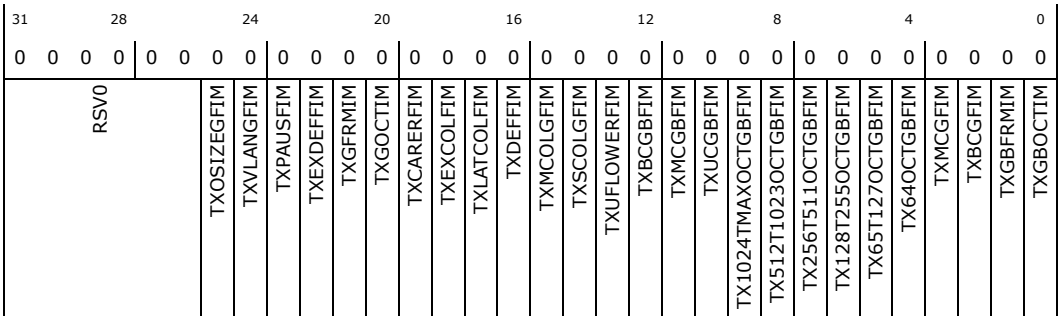
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 110h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:26	000000b RO	**Reserved (RSV0):** Reserved.
25	0b RW	**MMC Transmit Oversize Good Frame Counter Interrupt Mask (TXOSIZEGFIM):** Setting this bit masks the interrupt when the txoversize_g counter reaches half of the maximum value or the maximum value.
24	0b RW	**MMC Transmit VLAN Good Frame Counter Interrupt Mask (TXVLANGFIM):** Setting this bit masks the interrupt when the txvlanframes_g counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
23	0b RW	**MMC Transmit Pause Frame Counter Interrupt Mask (TXPAUSFIM):** Setting this bit masks the interrupt when the txpauseframes counter reaches half of the maximum value or the maximum value.
22	0b RW	**MMC Transmit Excessive Deferral Frame Counter Interrupt Mask (TXEXDEFFIM):** Setting this bit masks the interrupt when the txexcessdef counter reaches half of the maximum value or the maximum value.
21	0b RW	**MMC Transmit Good Frame Counter Interrupt Mask (TXGFRMIM):** Setting this bit masks the interrupt when the txframecount_g counter reaches half of the maximum value or the maximum value.
20	0b RW	**MMC Transmit Good Octet Counter Interrupt Mask (TXGOCTIM):** Setting this bit masks the interrupt when the txoctetcount_g counter reaches half of the maximum value or the maximum value.
19	0b RW	**MMC Transmit Carrier Error Frame Counter Interrupt Mask (TXCARERFIM):** Setting this bit masks the interrupt when the txcarriererror counter reaches half of the maximum value or the maximum value.
18	0b RW	**MMC Transmit Excessive Collision Frame Counter Interrupt Mask (TXEXCOLFIM):** Setting this bit masks the interrupt when the txexcesscol counter reaches half of the maximum value or the maximum value.
17	0b RW	**MMC Transmit Late Collision Frame Counter Interrupt Mask (TXLATCOLFIM):** Setting this bit masks the interrupt when the txlatecol counter reaches half of the maximum value or the maximum value.
16	0b RW	**MMC Transmit Deferred Frame Counter Interrupt Mask (TXDEFFIM):** Setting this bit masks the interrupt when the txdeferred counter reaches half of the maximum value or the maximum value.
15	0b RW	**MMC Transmit Multiple Collision Good Frame Counter Interrupt Mask (TXMCOLGFIM):** Setting this bit masks the interrupt when the txmulticol_g counter reaches half of the maximum value or the maximum value.
14	0b RW	**MMC Transmit Single Collision Good Frame Counter Interrupt Mask (TXSCOLGFIM):** Setting this bit masks the interrupt when the txsinglecol_g counter reaches half of the maximum value or the maximum value.
13	0b RW	**MMC Transmit Underflow Error Frame Counter Interrupt Mask (TXUFLOWERFIM):** Setting this bit masks the interrupt when the txunderflowerror counter reaches half of the maximum value or the maximum value.
12	0b RW	**MMC Transmit Broadcast Good Bad Frame Counter Interrupt Mask (TXBCGBFIM):** Setting this bit masks the interrupt when the txbroadcastframes_gb counter reaches half of the maximum value or the maximum value.
11	0b RW	**MMC Transmit Multicast Good Bad Frame Counter Interrupt Mask (TXMCGBFIM):** Setting this bit masks the interrupt when the txmulticastframes_gb counter reaches half of the maximum value or the maximum value.
10	0b RW	**MMC Transmit Unicast Good Bad Frame Counter Interrupt Mask (TXUCGBFIM):** Setting this bit masks the interrupt when the txunicastframes_gb counter reaches half of the maximum value or the maximum value.
9	0b RW	**MMC Transmit 1024 to Maximum Octet Good Bad Frame Counter Interrupt Mask (TX1024TMAXOCTGBFIM):** Setting this bit masks the interrupt when the tx1024tomaxoctets_gb counter reaches half of the maximum value or the maximum value.
8	0b RW	**MMC Transmit 512 to 1023 Octet Good Bad Frame Counter Interrupt Mask (TX512T1023OCTGBFIM):** Setting this bit masks the interrupt when the tx512to1023octets_gb counter reaches half of the maximum value or the maximum value.
7	0b RW	**MMC Transmit 256 to 511 Octet Good Bad Frame Counter Interrupt Mask (TX256T511OCTGBFIM):** Setting this bit masks the interrupt when the tx256to511octets_gb counter reaches half of the maximum value or the maximum value.
6	0b RW	**MMC Transmit 128 to 255 Octet Good Bad Frame Counter Interrupt Mask (TX128T255OCTGBFIM):** Setting this bit masks the interrupt when the tx128to255octets_gb counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
5	0b RW	**MMC Transmit 65 to 127 Octet Good Bad Frame Counter Interrupt Mask (TX65T127OCTGBFIM):** Setting this bit masks the interrupt when the tx65to127octets_gb counter reaches half of the maximum value or the maximum value.
4	0b RW	**MMC Transmit 64 Octet Good Bad Frame Counter Interrupt Mask (TX64OCTGBFIM):** Setting this bit masks the interrupt when the tx64octets_gb counter reaches half of the maximum value or the maximum value.
3	0b RW	**MMC Transmit Multicast Good Frame Counter Interrupt Mask (TXMCGFIM):** Setting this bit masks the interrupt when the txmulticastframes_g counter reaches half of the maximum value or the maximum value.
2	0b RW	**MMC Transmit Broadcast Good Frame Counter Interrupt Mask (TXBCGFIM):** Setting this bit masks the interrupt when the txbroadcastframes_g counter reaches half of the maximum value or the maximum value.
1	0b RW	**MMC Transmit Good Bad Frame Counter Interrupt Mask (TXGBFRMIM):** Setting this bit masks the interrupt when the txframecount_gb counter reaches half of the maximum value or the maximum value.
0	0b RW	**MMC Transmit Good Bad Octet Counter Interrupt Mask (TXGBOCTIM):** Setting this bit masks the interrupt when the txoctetcount_gb counter reaches half of the maximum value or the maximum value.

15.6.20 MMC Transmit Good Bad Octet Counter Register (TXOCTETCOUNT_GB)—Offset 114h

Number of bytes transmitted, exclusive of preamble and retried bytes, in good and bad frames.

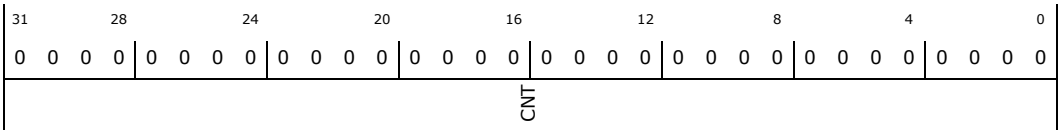
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 114h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.21 MMC Transmit Good Bad Frame Counter Register (TXFRAMECOUNT_GB)—Offset 118h

Number of good and bad frames transmitted, exclusive of retried frames.

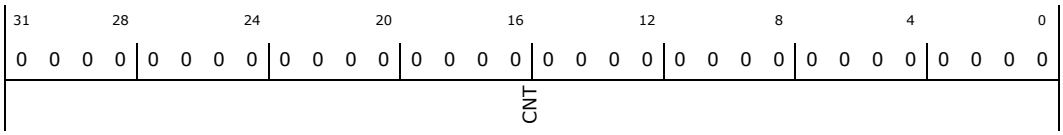
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 118h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.22 MMC Transmit Broadcast Good Frame Counter Register (TXBROADCASTFRAMES_G)—Offset 11Ch

Number of good broadcast frames transmitted.

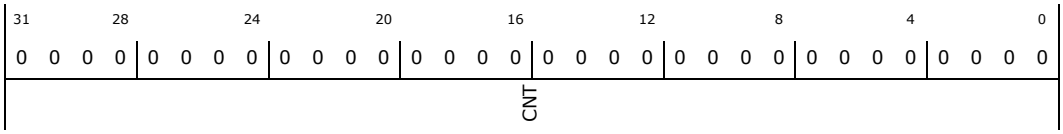
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 11Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.23 MMC Transmit Multicast Good Frame Counter Register (TXMULTICASTFRAMES_G)—Offset 120h

Number of good multicast frames transmitted.

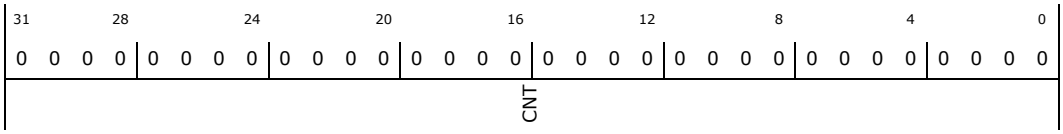
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 120h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.24 MMC Transmit 64 Octet Good Bad Frame Counter Register (TX64OCTETS_GB)—Offset 124h

Number of good and bad frames transmitted with length 64 bytes, exclusive of preamble and retried frames.

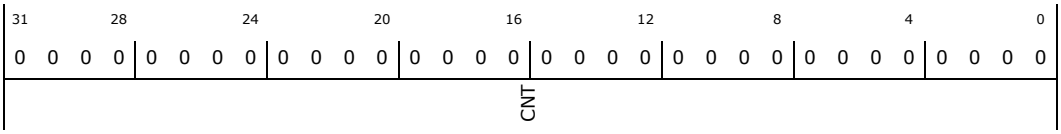
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 124h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.25 MMC Transmit 65 to 127 Octet Good Bad Frame Counter Register (TX65TO127OCTETS_GB)—Offset 128h

Number of good and bad frames transmitted with length between 65 and 127 (inclusive) bytes, exclusive of preamble and retried frames.

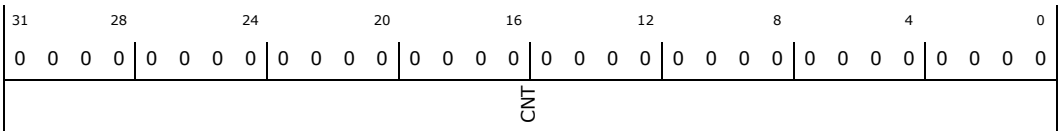
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 128h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.26 MMC Transmit 128 to 255 Octet Good Bad Frame Counter Register (TX128TO255OCTETS_GB)—Offset 12Ch

Number of good and bad frames transmitted with length between 128 and 255 (inclusive) bytes, exclusive of preamble and retried frames.

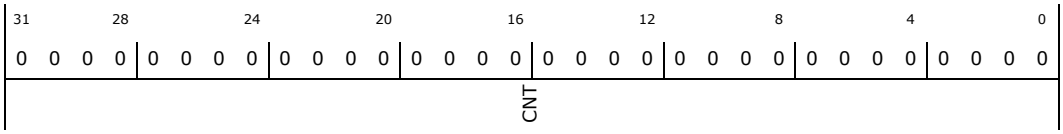
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 12Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.27 MMC Transmit 256 to 511 Octet Good Bad Frame Counter Register (TX256TO511OCTETS_GB)—Offset 130h

Number of good and bad frames transmitted with length between 256 and 511 (inclusive) bytes, exclusive of preamble and retried frames.

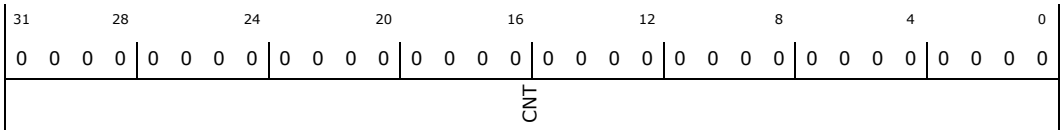
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 130h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.28 MMC Transmit 512 to 1023 Octet Good Bad Frame Counter Register (TX512TO1023OCTETS_GB)—Offset 134h

Number of good and bad frames transmitted with length between 512 and 1,023 (inclusive) bytes, exclusive of preamble and retried frames.

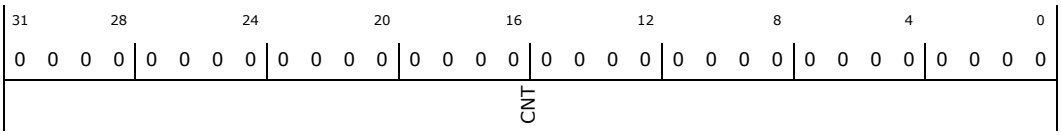
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 134h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.29 MMC Transmit 1024 to Maximum Octet Good Bad Frame Counter Register (TX1024TOMAXOCTETS_GB)—Offset 138h

Number of good and bad frames transmitted with length between 1,024 and maxsize (inclusive) bytes, exclusive of preamble and retried frames.

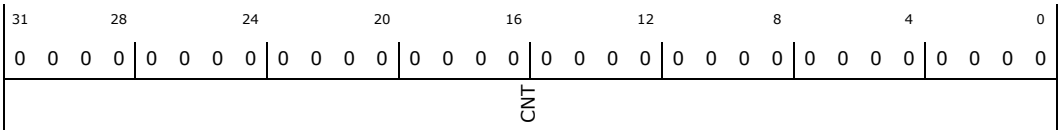
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 138h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.30 MMC Transmit Unicast Good Bad Frame Counter Register (TXUNICASTFRAMES_GB)—Offset 13Ch

Number of good and bad unicast frames transmitted.

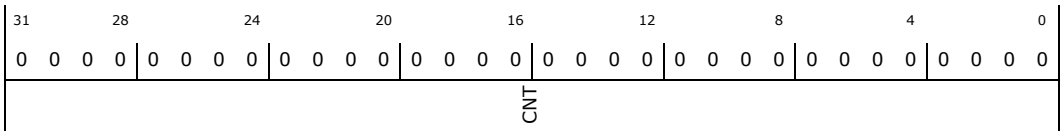
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 13Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.31 MMC Transmit Multicast Good Bad Frame Counter Register (TXMULTICASTFRAMES_GB)—Offset 140h

Number of good and bad multicast frames transmitted.

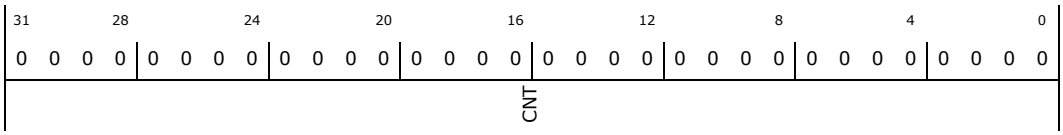
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 140h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.32 MMC Transmit Broadcast Good Bad Frame Counter Register (TXBROADCASTFRAMES_GB)—Offset 144h

Number of good and bad broadcast frames transmitted.

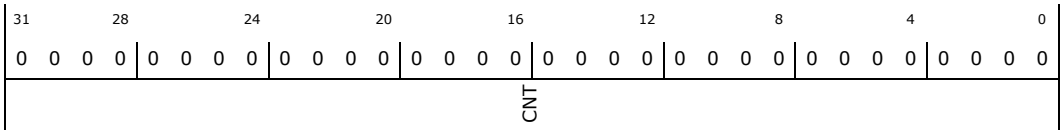
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 144h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.33 MMC Transmit Underflow Error Frame Counter Register (TXUNDERFLOWERROR)—Offset 148h

Number of frames aborted because of frame underflow error.

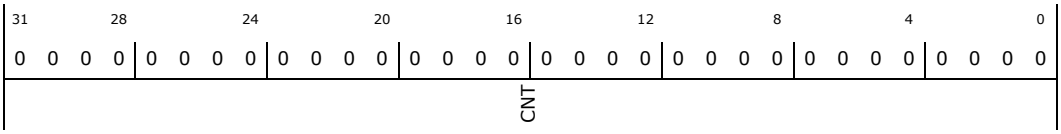
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 148h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.34 MMC Transmit Single Collision Good Frame Counter Register (TXSINGLECOL_G)—Offset 14Ch

Number of successfully transmitted frames after a single collision in the half-duplex mode.

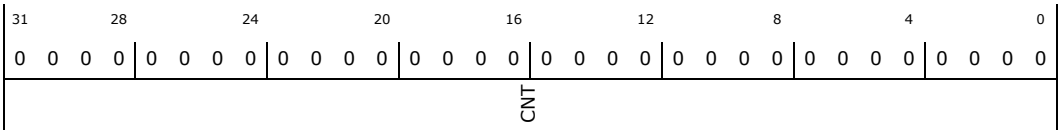
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.35 MMC Transmit Multiple Collision Good Frame Counter Register (TXMULTICOL_G)—Offset 150h

Number of successfully transmitted frames after multiple collisions in the half-duplex mode.

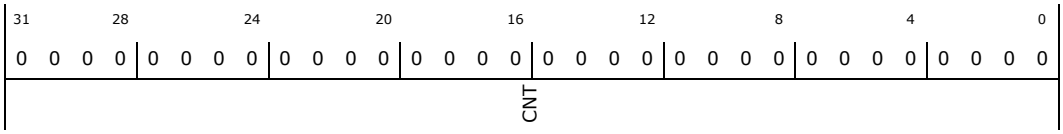
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 150h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.36 MMC Transmit Deferred Frame Counter Register (TXDEFERRED)—Offset 154h

Number of successfully transmitted frames after a deferral in the half-duplex mode.

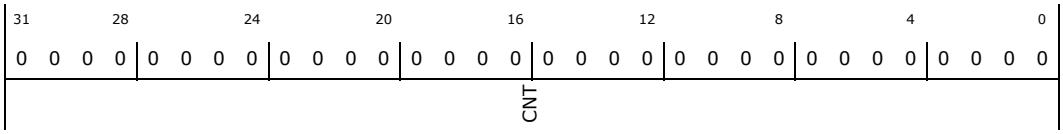
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 154h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.37 MMC Transmit Late Collision Frame Counter Register (TXLATECOL)—Offset 158h

Number of frames aborted because of late collision error.

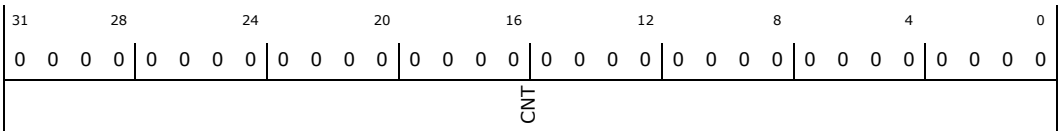
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 158h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.38 MMC Transmit Excessive Collision Frame Counter Register (TXEXESSCOL)—Offset 15Ch

Number of frames aborted because of excessive (16) collision errors.

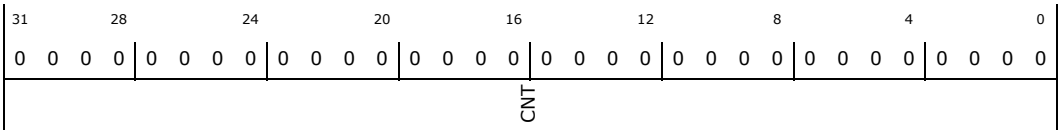
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 15Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.39 MMC Transmit Carrier Error Frame Counter Register (TXCARRIERERROR)—Offset 160h

Number of frames aborted because of carrier sense error (no carrier or loss of carrier).

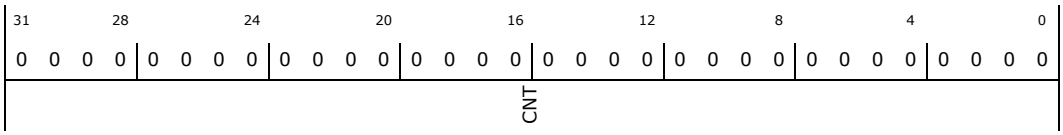
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 160h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.40 MMC Transmit Good Octet Counter Register (TXOCTETCOUNT_G)—Offset 164h

Number of bytes transmitted, exclusive of preamble, in good frames only.

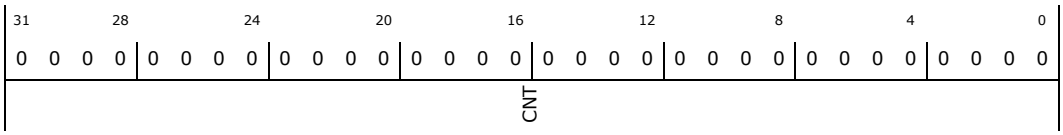
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 164h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.41 MMC Transmit Good Frame Counter Register (TXFRAMECOUNT_G)—Offset 168h

Number of good frames transmitted.

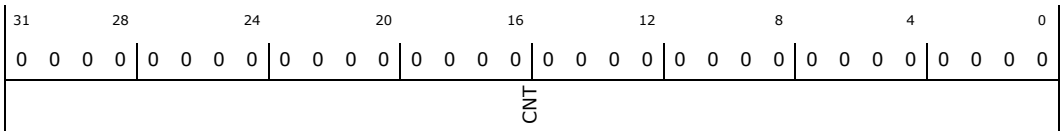
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 168h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.42 MMC Transmit Excessive Deferral Frame Counter Register (TXEXCESSDEF)—Offset 16Ch

Number of frames aborted because of excessive deferral error (deferred for more than two max-sized frame times).

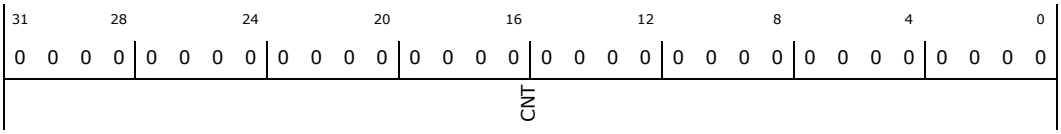
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 16Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.43 MMC Transmit Pause Frame Counter Register (TXPAUSEFRAMES)—Offset 170h

Number of good PAUSE frames transmitted.

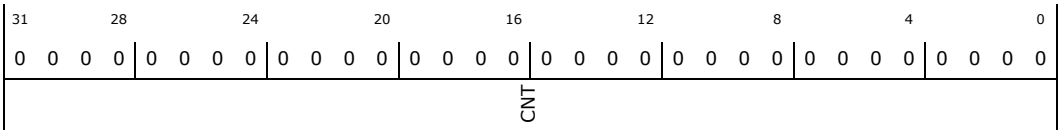
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 170h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.44 MMC Transmit VLAN Good Frame Counter Register (TXVLANFRAMES_G)—Offset 174h

Number of good VLAN frames transmitted, exclusive of retried frames.

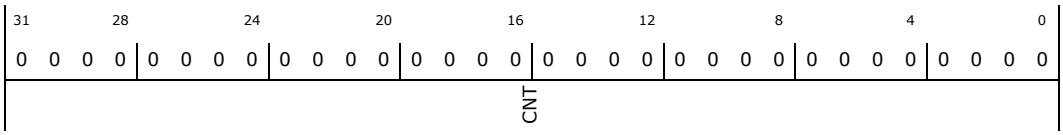
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 174h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.45 MMC Transmit Oversize Good Frame Counter Register (TXOVERSIZE_G)—Offset 178h

Number of frames transmitted without errors and with length greater than the maxsize (1,518 or 1,522 bytes for VLAN tagged frames; 2000 bytes if enabled in Bit 27 of Register 0 (MAC Configuration Register)).

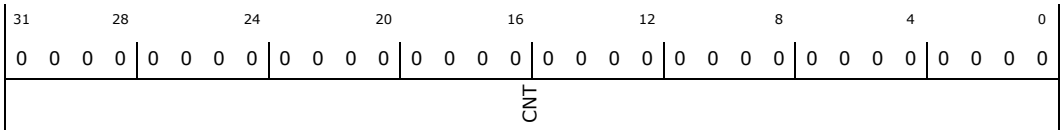
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 178h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.46 MMC Receive Good Bad Frame Counter Register (RXFRAMECOUNT_GB)—Offset 180h

Number of good and bad frames received.

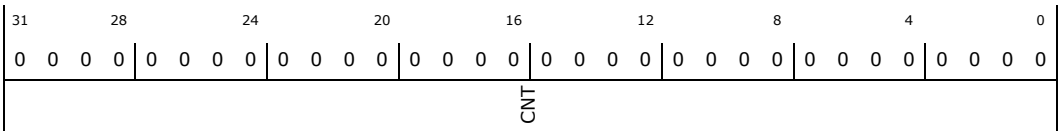
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 180h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.47 MMC Receive Good Bad Octet Counter Register (RXOCTETCOUNT_GB)—Offset 184h

Number of bytes received, exclusive of preamble, in good and bad frames.

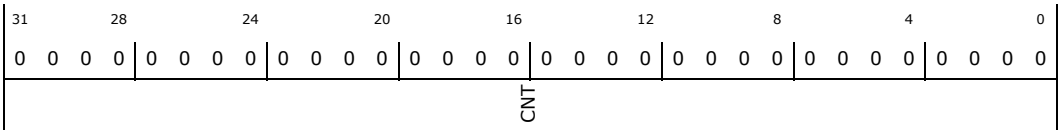
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 184h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.48 MMC Receive Good Octet Counter Register (RXOCTETCOUNT_G)—Offset 188h

Number of bytes received, exclusive of preamble, only in good frames.

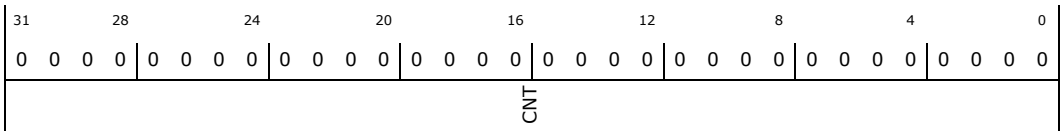
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 188h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.49 MMC Receive Broadcast Good Frame Counter Register (RXBROADCASTFRAMES_G)—Offset 18Ch

Number of good broadcast frames received.

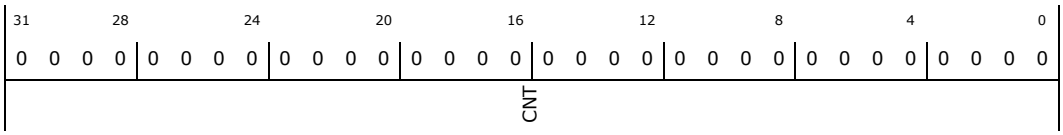
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 18Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.50 MMC Receive Multicast Good Frame Counter Register (RXMULTICASTFRAMES_G)—Offset 190h

Number of good multicast frames received.

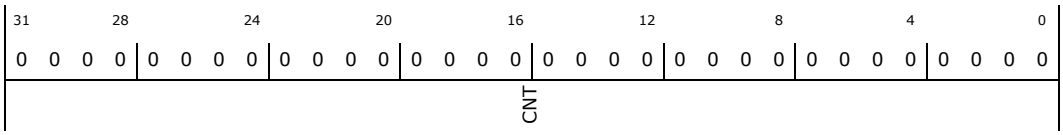
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 190h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.51 MMC Receive CRC Error Frame Counter Register (RXCRCERROR)—Offset 194h

Number of frames received with CRC error.

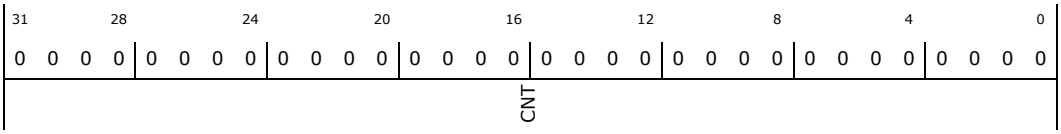
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 194h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.52 MMC Receive Alignment Error Frame Counter Register (RXALIGNMENTERROR)—Offset 198h

Number of frames received with alignment (dribble) error.

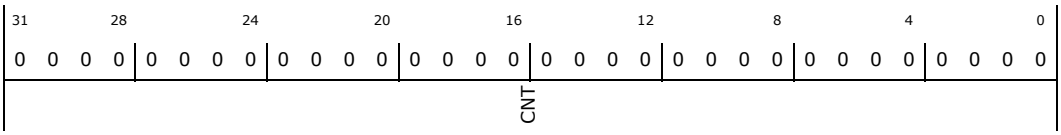
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 198h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.53 MMC Receive Runt Frame Counter Register (RXRUNTERROR)—Offset 19Ch

Number of frames received with runt ((64 bytes and CRC error) error.

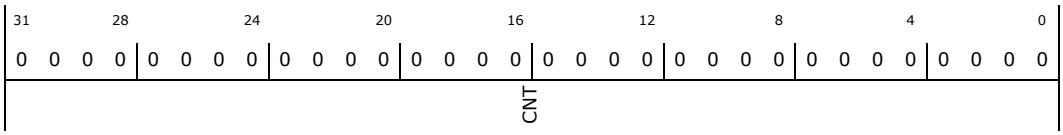
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 19Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.54 MMC Receive Jabber Error Frame Counter Register (RXJABBERERROR)—Offset 1A0h

Number of giant frames received with length (including CRC) greater than 1,518 bytes (1,522 bytes for VLAN tagged) and with CRC error. If Jumbo Frame mode is enabled, then frames of length greater than 9,018 bytes (9,022 for VLAN tagged) are considered as giant frames.

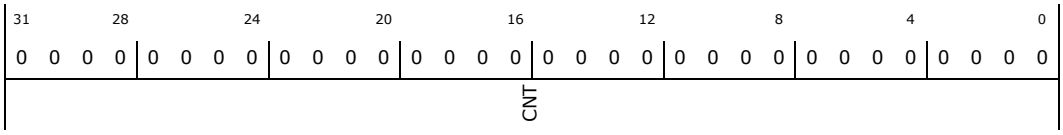
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1A0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.55 MMC Receive Undersize Good Frame Counter Register (RXUNDERSIZE_G)—Offset 1A4h

Number of frames received with length less than 64 bytes, without any errors.

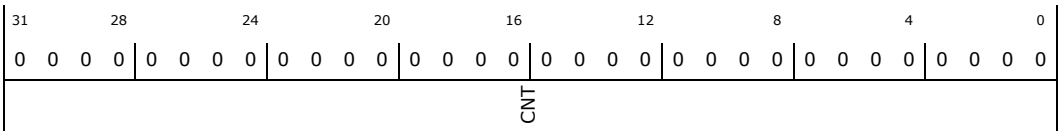
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1A4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.56 MMC Receive Oversize Good Frame Counter Register (RXOVERSIZE_G)—Offset 1A8h

Number of frames received without errors, with length greater than the maxsize (1,518 or 1,522 for VLAN tagged frames; 2,000 bytes if enabled in Bit 27 of Register 0 (MAC Configuration Register)).

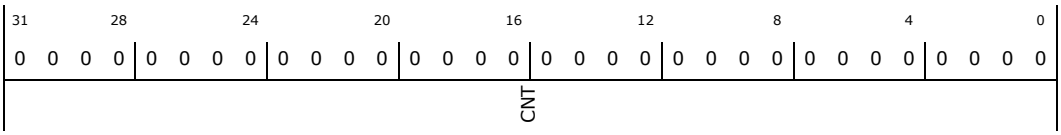
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1A8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.57 MMC Receive 64 Octet Good Bad Frame Counter Register (RX64OCTETS_GB)—Offset 1ACh

Number of good and bad frames received with length 64 bytes, exclusive of preamble.

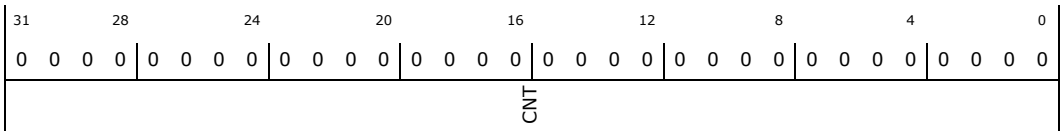
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1ACh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.58 MMC Receive 65 to 127 Octet Good Bad Frame Counter Register (RX65TO127OCTETS_GB)—Offset 1B0h

Number of good and bad frames received with length between 65 and 127 (inclusive) bytes, exclusive of preamble.

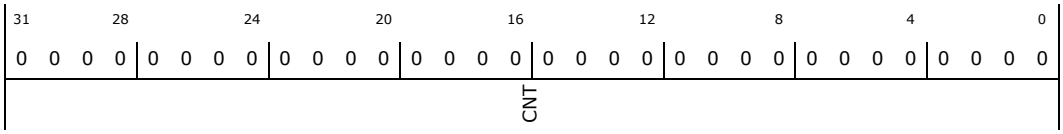
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1B0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.59 MMC Receive 128 to 255 Octet Good Bad Frame Counter Register (RX128TO255OCTETS_GB)—Offset 1B4h

Number of good and bad frames received with length between 128 and 255 (inclusive) bytes, exclusive of preamble.

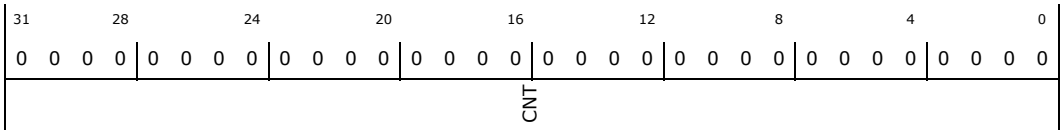
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1B4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.60 MMC Receive 256 to 511 Octet Good Bad Frame Counter Register (RX256TO511OCTETS_GB)—Offset 1B8h

Number of good and bad frames received with length between 256 and 511 (inclusive) bytes, exclusive of preamble.

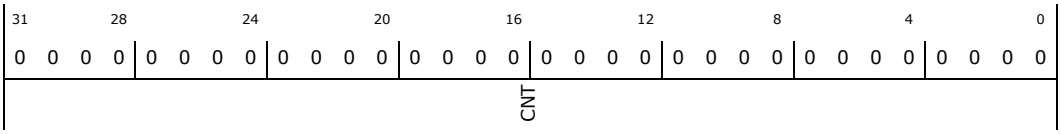
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1B8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.61 MMC Receive 512 to 1023 Octet Good Bad Frame Counter Register (RX512TO1023OCTETS_GB)—Offset 1BCh

Number of good and bad frames received with length between 512 and 1,023 (inclusive) bytes, exclusive of preamble.

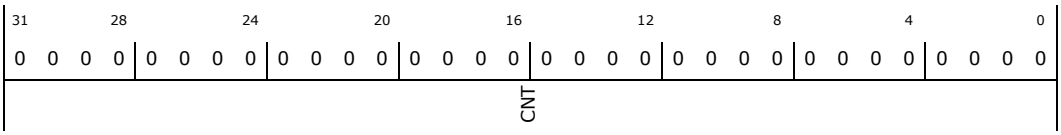
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1BCh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.62 MMC Receive 1024 to Maximum Octet Good Bad Frame Counter Register (RX1024TOMAXOCTETS_GB)—Offset 1C0h

Number of good and bad frames received with length between 1,024 and maxsize (inclusive) bytes, exclusive of preamble and retried frames.

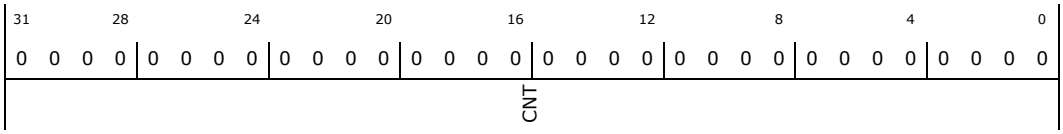
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1C0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.63 MMC Receive Unicast Good Frame Counter Register (RXUNICASTFRAMES_G)—Offset 1C4h

Number of received good unicast frames.

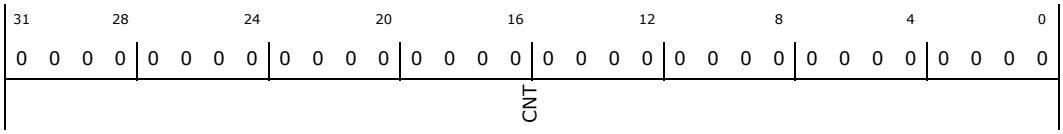
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1C4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.64 MMC Receive Length Error Frame Counter Register (RXLENGTHERROR)—Offset 1C8h

Number of frames received with length error (Length type field doesn't match frame size), for all frames with valid length field.

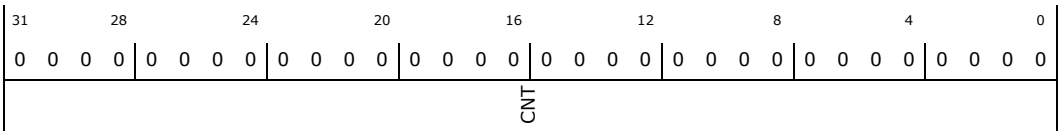
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1C8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.65 MMC Receive Out Of Range Error Frame Counter Register (RXOUTOFRANGETYPE)—Offset 1CCh

Number of frames received with length field not equal to the valid frame size (greater than 1,500 but less than 1,536).

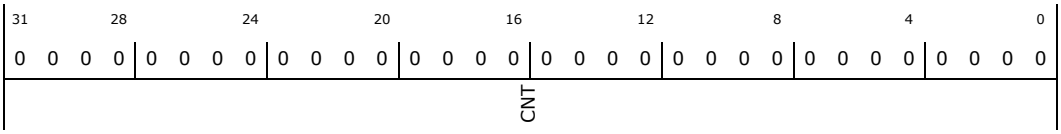
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1CCh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.66 MMC Receive Pause Frame Counter Register (RXPAUSEFRAMES)—Offset 1D0h

Number of good and valid PAUSE frames received.

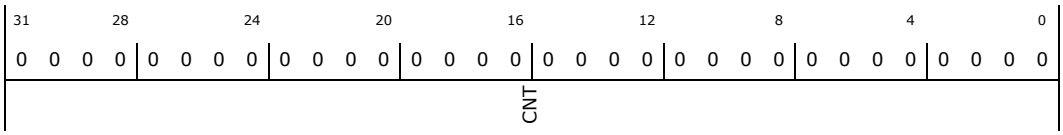
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1D0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.67 MMC Receive FIFO Overflow Frame Counter Register (RXFIFOOVERFLOW)—Offset 1D4h

Number of missed received frames because of FIFO overflow.

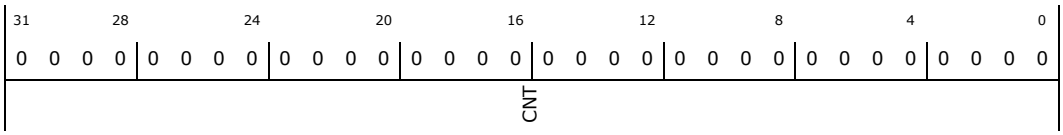
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1D4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.68 MMC Receive VLAN Good Bad Frame Counter Register (RXVLANFRAMES_GB)—Offset 1D8h

Number of good and bad VLAN frames received.

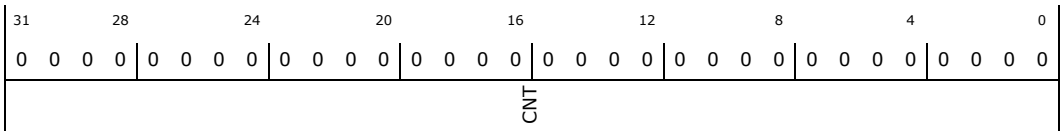
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1D8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.69 MMC Receive Watchdog Error Frame Counter Register (RXWATCHDOGERROR)—Offset 1DCh

Number of frames received with error because of watchdog timeout error (frames with a data load larger than 2,048 bytes).

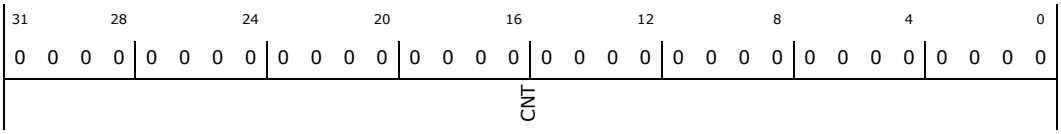
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1DCh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.70 MMC Receive Error Frame Counter Register (RXRCVERROR)—Offset 1E0h

Number of frames received with Receive error or Frame Extension error on the MII interface.

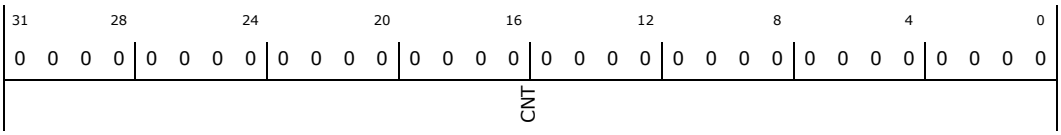
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1E0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.71 MMC Receive Control Frame Counter Register (RXCTRLFRAMES_G)—Offset 1E4h

Number of received good control frames.

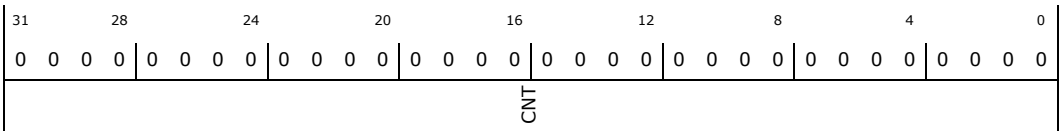
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1E4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.72 MMC IPC Receive Checksum Offload Interrupt Mask Register (MMC_IPC_INTR_MASK_RX)—Offset 200h

The MMC IPC Receive Checksum Offload Interrupt Mask maintains the mask for the interrupt generated from the receive IPC statistic counters.

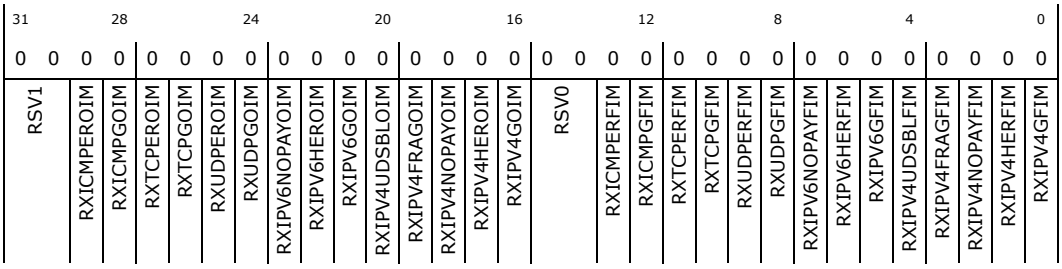
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

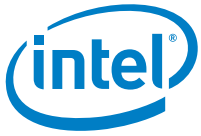
Offset: [BAR0] + 200h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0b RO	**Reserved (RSV1):** Reserved.
29	0b RW	**MMC Receive ICMP Error Octet Counter Interrupt Mask (RXICMPEROIM):** Setting this bit masks the interrupt when the rxicmp_err_octets counter reaches half of the maximum value or the maximum value.
28	0b RW	**MMC Receive ICMP Good Octet Counter Interrupt Mask (RXICMPGOIM):** Setting this bit masks the interrupt when the rxicmp_gd_octets counter reaches half of the maximum value or the maximum value.



Bit Range	Default & Access	Field Name (ID): Description
27	0b RW	**MMC Receive TCP Error Octet Counter Interrupt Mask (RXTCPEROIM):** Setting this bit masks the interrupt when the rxtcp_err_octets counter reaches half of the maximum value or the maximum value.
26	0b RW	**MMC Receive TCP Good Octet Counter Interrupt Mask (RXTCPGOIM):** Setting this bit masks the interrupt when the rxtcp_gd_octets counter reaches half of the maximum value or the maximum value.
25	0b RW	**MMC Receive UDP Good Octet Counter Interrupt Mask (RXUDPEROIM):** Setting this bit masks the interrupt when the rxudp_err_octets counter reaches half of the maximum value or the maximum value.
24	0b RW	**MMC Receive IPV6 No Payload Octet Counter Interrupt Mask (RXUDPGOIM):** Setting this bit masks the interrupt when the rxudp_gd_octets counter reaches half of the maximum value or the maximum value.
23	0b RW	**MMC Receive IPV6 Header Error Octet Counter Interrupt Mask (RXIPV6NOPAYOIM):** Setting this bit masks the interrupt when the rxipv6_nopay_octets counter reaches half of the maximum value or the maximum value.
22	0b RW	**MMC Receive IPV6 Good Octet Counter Interrupt Mask (RXIPV6HEROIM):** Setting this bit masks the interrupt when the rxipv6_hdrerr_octets counter reaches half of the maximum value or the maximum value.
21	0b RW	**MMC Receive IPV6 Good Octet Counter Interrupt Mask (RXIPV6GOIM):** Setting this bit masks the interrupt when the rxipv6_gd_octets counter reaches half of the maximum value or the maximum value.
20	0b RW	**MMC Receive IPV4 UDP Checksum Disabled Octet Counter Interrupt Mask (RXIPV4UDSBLOIM):** Setting this bit masks the interrupt when the rxipv4_udsbl_octets counter reaches half of the maximum value or the maximum value.
19	0b RW	**MMC Receive IPV4 Fragmented Octet Counter Interrupt Mask (RXIPV4FRAGOIM):** Setting this bit masks the interrupt when the rxipv4_frag_octets counter reaches half of the maximum value or the maximum value.
18	0b RW	**MMC Receive IPV4 No Payload Octet Counter Interrupt Mask (RXIPV4NOPAYOIM):** Setting this bit masks the interrupt when the rxipv4_nopay_octets counter reaches half of the maximum value or the maximum value.
17	0b RW	**MMC Receive IPV4 Header Error Octet Counter Interrupt Mask (RXIPV4HEROIM):** Setting this bit masks the interrupt when the rxipv4_hdrerr_octets counter reaches half of the maximum value or the maximum value.
16	0b RW	**MMC Receive IPV4 Good Octet Counter Interrupt Mask (RXIPV4GOIM):** Setting this bit masks the interrupt when the rxipv4_gd_octets counter reaches half of the maximum value or the maximum value.
15:14	00b RO	**Reserved (RSV0):** Reserved.
13	0b RW	**MMC Receive ICMP Error Frame Counter Interrupt Mask (RXICMPERFIM):** Setting this bit masks the interrupt when the rxicmp_err_frms counter reaches half of the maximum value or the maximum value.
12	0b RW	**MMC Receive ICMP Good Frame Counter Interrupt Mask (RXICMPGFIM):** Setting this bit masks the interrupt when the rxicmp_gd_frms counter reaches half of the maximum value or the maximum value.
11	0b RW	**MMC Receive TCP Error Frame Counter Interrupt Mask (RXTCPERFIM):** Setting this bit masks the interrupt when the rxtcp_err_frms counter reaches half of the maximum value or the maximum value.
10	0b RW	**MMC Receive TCP Good Frame Counter Interrupt Mask (RXTCPGFIM):** Setting this bit masks the interrupt when the rxtcp_gd_frms counter reaches half of the maximum value or the maximum value.
9	0b RW	**MMC Receive UDP Error Frame Counter Interrupt Mask (RXUDPERFIM):** Setting this bit masks the interrupt when the rxudp_err_frms counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
8	0b RW	**MMC Receive UDP Good Frame Counter Interrupt Mask (RXUDPGFIM):** Setting this bit masks the interrupt when the rxudp_gd_frms counter reaches half of the maximum value or the maximum value.
7	0b RW	**MMC Receive IPV6 No Payload Frame Counter Interrupt Mask (RXIPV6NOPAYFIM):** Setting this bit masks the interrupt when the rxipv6_nopay_frms counter reaches half of the maximum value or the maximum value.
6	0b RW	**MMC Receive IPV6 Header Error Frame Counter Interrupt Mask (RXIPV6HERFIM):** Setting this bit masks the interrupt when the rxipv6_hdrerr_frms counter reaches half of the maximum value or the maximum value.
5	0b RW	**MMC Receive IPV6 Good Frame Counter Interrupt Mask (RXIPV6GFIM):** Setting this bit masks the interrupt when the rxipv6_gd_frms counter reaches half of the maximum value or the maximum value.
4	0b RW	**MMC Receive IPV4 UDP Checksum Disabled Frame Counter Interrupt Mask (RXIPV4UDSBLFIM):** Setting this bit masks the interrupt when the rxipv4_udsbl_frms counter reaches half of the maximum value or the maximum value.
3	0b RW	**MMC Receive IPV4 Fragmented Frame Counter Interrupt Mask (RXIPV4FRAGFIM):** Setting this bit masks the interrupt when the rxipv4_frag_frms counter reaches half of the maximum value or the maximum value.
2	0b RW	**MMC Receive IPV4 No Payload Frame Counter Interrupt Mask (RXIPV4NOPAYFIM):** Setting this bit masks the interrupt when the rxipv4_nopay_frms counter reaches half of the maximum value or the maximum value.
1	0b RW	**MMC Receive IPV4 Header Error Frame Counter Interrupt Mask (RXIPV4HERFIM):** Setting this bit masks the interrupt when the rxipv4_hdrerr_frms counter reaches half of the maximum value or the maximum value.
0	0b RW	**MMC Receive IPV4 Good Frame Counter Interrupt Mask (RXIPV4GFIM):** Setting this bit masks the interrupt when the rxipv4_gd_frms counter reaches half of the maximum value or the maximum value.

15.6.73 MMC Receive Checksum Offload Interrupt Register (MMC_IPC_INTR_RX)—Offset 208h

The MMC Receive Checksum Offload Interrupt register maintains the interrupts generated when receive IPC statistic counters reach half their maximum values, and when they cross their maximum values. When Counter Stop Rollover is set, then interrupts are set but the counter remains at all-ones. When the MMC IPC counter that caused the interrupt is read, its corresponding interrupt bit is cleared.

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 208h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSV1	RXICMPEROIS	RXICMPGOIS	RXTCPEROIS	RXTCPGOIS	RXUDPEROIS	RXUDPGOIS	RXIPV6NOPAYOIS	RXIPV6HEROIS	RXIPV6GOIS	RXIPV4UDSBLOIS	RXIPV4FRAGOIS	RXIPV4NOPAYOIS	RXIPV4HEROIS	RXIPV4GOIS	RSV0	RXICMPERFIS	RXICMPGFIS	RXTCPERFIS	RXTCPGFIS	RXUDPERFIS	RXUDPGFIS	RXIPV6NOPAYFIS	RXIPV6HERFIS	RXIPV6GFIS	RXIPV4UDSBLFIS	RXIPV4FRAGFIS	RXIPV4NOPAYFIS	RXIPV4HERFIS	RXIPV4GFIS		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0b RO	**Reserved (RSV1):** Reserved.
29	0b RO	**MMC Receive ICMP Error Octet Counter Interrupt Status (RXICMPEROIS):** This bit is set when the rxicmp_err_octets counter reaches half of the maximum value or the maximum value.
28	0b RO	**MMC Receive ICMP Good Octet Counter Interrupt Status (RXICMPGOIS):** This bit is set when the rxicmp_gd_octets counter reaches half of the maximum value or the maximum value.
27	0b RO	**MMC Receive TCP Error Octet Counter Interrupt Status (RXTCPEROIS):** This bit is set when the rxtcp_err_octets counter reaches half of the maximum value or the maximum value.
26	0b RO	**MMC Receive TCP Good Octet Counter Interrupt Status (RXTCPGOIS):** This bit is set when the rxtcp_gd_octets counter reaches half of the maximum value or the maximum value.
25	0b RO	**MMC Receive UDP Error Octet Counter Interrupt Status (RXUDPEROIS):** This bit is set when the rxudp_err_octets counter reaches half of the maximum value or the maximum value.
24	0b RO	**MMC Receive UDP Good Octet Counter Interrupt Status (RXUDPGOIS):** This bit is set when the rxudp_gd_octets counter reaches half of the maximum value or the maximum value.
23	0b RO	**MMC Receive IPV6 No Payload Octet Counter Interrupt Status (RXIPV6NOPAYOIS):** This bit is set when the rxipv6_nopay_octets counter reaches half of the maximum value or the maximum value.
22	0b RO	**MMC Receive IPV6 Header Error Octet Counter Interrupt Status (RXIPV6HEROIS):** This bit is set when the rxipv6_hdrerr_octets counter reaches half of the maximum value or the maximum value.
21	0b RO	**MMC Receive IPV6 Good Octet Counter Interrupt Status (RXIPV6GOIS):** This bit is set when the rxipv6_gd_octets counter reaches half of the maximum value or the maximum value.
20	0b RO	**MMC Receive IPV4 UDP Checksum Disabled Octet Counter Interrupt Status (RXIPV4UDSBLOIS):** This bit is set when the rxipv4_udsbl_octets counter reaches half of the maximum value or the maximum value.
19	0b RO	**MMC Receive IPV4 Fragmented Octet Counter Interrupt Status (RXIPV4FRAGOIS):** This bit is set when the rxipv4_frag_octets counter reaches half of the maximum value or the maximum value.
18	0b RO	**MMC Receive IPV4 No Payload Octet Counter Interrupt Status (RXIPV4NOPAYOIS):** This bit is set when the rxipv4_nopay_octets counter reaches half of the maximum value or the maximum value.
17	0b RO	**MMC Receive IPV4 Header Error Octet Counter Interrupt Status (RXIPV4HEROIS):** This bit is set when the rxipv4_hdrerr_octets counter reaches half of the maximum value or the maximum value.
16	0b RO	**MMC Receive IPV4 Good Octet Counter Interrupt Status (RXIPV4GOIS):** This bit is set when the rxipv4_gd_octets counter reaches half of the maximum value or the maximum value.
15:14	00b RO	**Reserved (RSV0):** Reserved.
13	0b RO	**MMC Receive ICMP Error Frame Counter Interrupt Status (RXICMPERFIS):** This bit is set when the rxicmp_err_frms counter reaches half of the maximum value or the maximum value.
12	0b RO	**MMC Receive ICMP Good Frame Counter Interrupt Status (RXICMPGFIS):** This bit is set when the rxicmp_gd_frms counter reaches half of the maximum value or the maximum value.
11	0b RO	**MMC Receive TCP Error Frame Counter Interrupt Status (RXTCPERFIS):** This bit is set when the rxtcp_err_frms counter reaches half of the maximum value or the maximum value.

Bit Range	Default & Access	Field Name (ID): Description
10	0b RO	**MMC Receive TCP Good Frame Counter Interrupt Status (RXTCPGFIS):** This bit is set when the rxtcp_gd_frms counter reaches half of the maximum value or the maximum value.
9	0b RO	**MMC Receive UDP Error Frame Counter Interrupt Status (RXUDPERFIS):** This bit is set when the rxudp_err_frms counter reaches half of the maximum value or the maximum value.
8	0b RO	**MMC Receive UDP Good Frame Counter Interrupt Status (RXUDPGFIS):** This bit is set when the rxudp_gd_frms counter reaches half of the maximum value or the maximum value.
7	0b RO	**MMC Receive IPV6 No Payload Frame Counter Interrupt Status (RXIPV6NOPAYFIS):** This bit is set when the rxipv6_nopay_frms counter reaches half of the maximum value or the maximum value.
6	0b RO	**MMC Receive IPV6 Header Error Frame Counter Interrupt Status (RXIPV6HERFIS):** This bit is set when the rxipv6_hdrerr_frms counter reaches half of the maximum value or the maximum value.
5	0b RO	**MMC Receive IPV6 Good Frame Counter Interrupt Status (RXIPV6GFIS):** This bit is set when the rxipv6_gd_frms counter reaches half of the maximum value or the maximum value.
4	0b RO	**MMC Receive IPV4 UDP Checksum Disabled Frame Counter Interrupt Status (RXIPV4UDSBLFIS):** This bit is set when the rxipv4_udsbl_frms counter reaches half of the maximum value or the maximum value.
3	0b RO	**MMC Receive IPV4 Fragmented Frame Counter Interrupt Status (RXIPV4FRAGFIS):** This bit is set when the rxipv4_frag_frms counter reaches half of the maximum value or the maximum value.
2	0b RO	**MMC Receive IPV4 No Payload Frame Counter Interrupt Status (RXIPV4NOPAYFIS):** This bit is set when the rxipv4_nopay_frms counter reaches half of the maximum value or the maximum value.
1	0b RO	**MMC Receive IPV4 Header Error Frame Counter Interrupt Status (RXIPV4HERFIS):** This bit is set when the rxipv4_hdrerr_frms counter reaches half of the maximum value or the maximum value.
0	0b RO	**MMC Receive IPV4 Good Frame Counter Interrupt Status (RXIPV4GFIS):** This bit is set when the rxipv4_gd_frms counter reaches half of the maximum value or the maximum value.

15.6.74 MMC Receive IPV4 Good Frame Counter Register (RXIPV4_GD_FRMS)—Offset 210h

Number of good IPv4 datagrams received with the TCP, UDP, or ICMP payload.

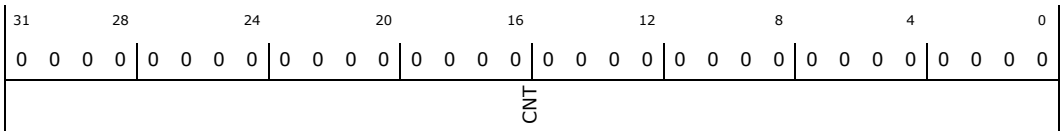
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 210h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.75 MMC Receive IPV4 Header Error Frame Counter Register (RXIPV4_HDRERR_FRMS)—Offset 214h

Number of IPv4 datagrams received with header (checksum, length, or version mismatch) errors.

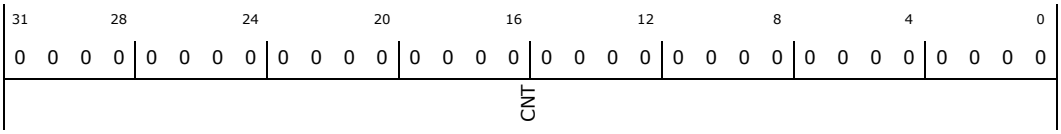
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 214h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.76 MMC Receive IPV4 No Payload Frame Counter Register (RXIPV4_NOPAY_FRMS)—Offset 218h

Number of IPv4 datagram frames received that did not have a TCP, UDP, or ICMP payload processed by the Checksum engine.

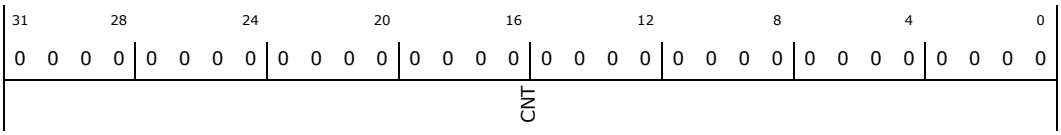
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 218h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.77 MMC Receive IPV4 Fragmented Frame Counter Register (RXIPV4_FRAG_FRMS)—Offset 21Ch

Number of good IPv4 datagrams with fragmentation.

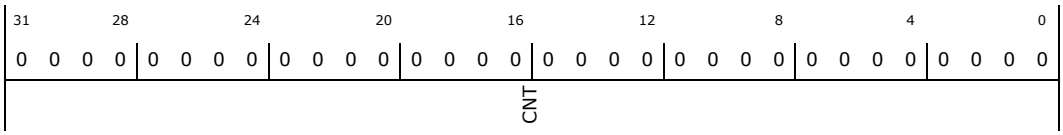
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 21Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.78 MMC Receive IPV4 UDP Checksum Disabled Frame Counter Register (RXIPV4_UDSBL_FRMS)—Offset 220h

Number of good IPv4 datagrams received that had a UDP payload with checksum disabled.

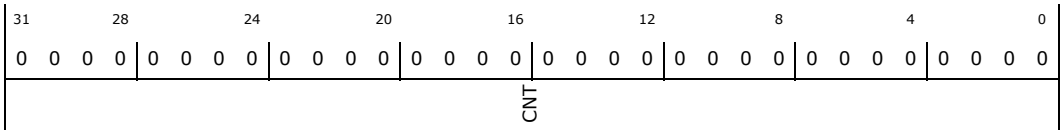
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 220h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.79 MMC Receive IPV6 Good Frame Counter Register (RXIPV6_GD_FRMS)—Offset 224h

Number of good IPv6 datagrams received with TCP, UDP, or ICMP payloads.

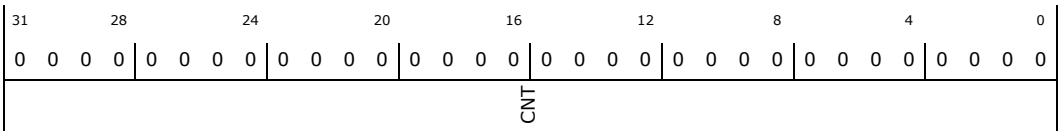
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 224h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.80 MMC Receive IPV6 Header Error Frame Counter Register (RXIPV6_HDRERR_FRMS)—Offset 228h

Number of IPv6 datagrams received with header errors (length or version mismatch).

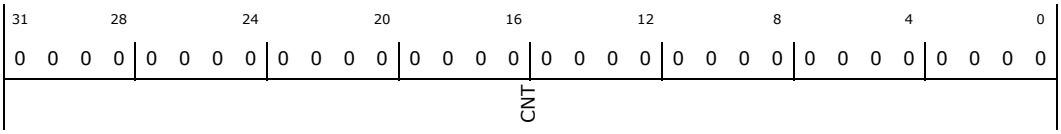
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 228h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.81 MMC Receive IPV6 No Payload Frame Counter Register (RXIPV6_NOPAY_FRMS)—Offset 22Ch

Number of IPv6 datagram frames received that did not have a TCP, UDP, or ICMP payload. This includes all IPv6 datagrams with fragmentation or security extension headers.

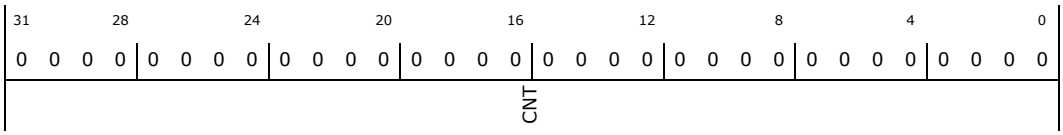
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 22Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.82 MMC Receive UDP Good Frame Counter Register (RXUDP_GD_FRMS)—Offset 230h

Number of good IP datagrams with a good UDP payload. This counter is not updated when the rxipv4_udsbl_frms counter is incremented.

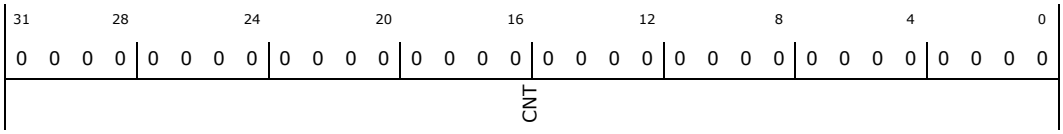
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 230h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.83 MMC Receive UDP Error Frame Counter Register (RXUDP_ERR_FRMS)—Offset 234h

Number of good IP datagrams whose UDP payload has a checksum error.

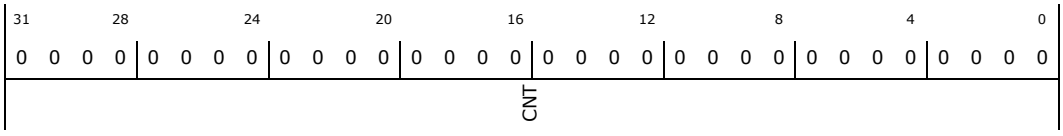
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 234h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.84 MMC Receive TCP Good Frame Counter Register (RXTCP_GD_FRMS)—Offset 238h

Number of good IP datagrams with a good TCP payload.

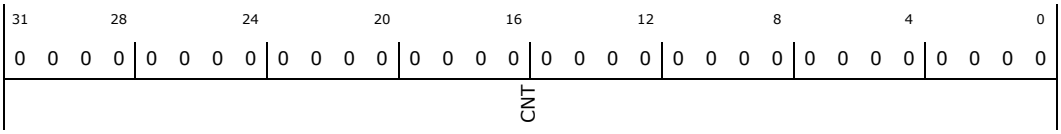
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 238h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.85 MMC Receive TCP Error Frame Counter Register (RXTCP_ERR_FRMS)—Offset 23Ch

Number of good IP datagrams whose TCP payload has a checksum error.

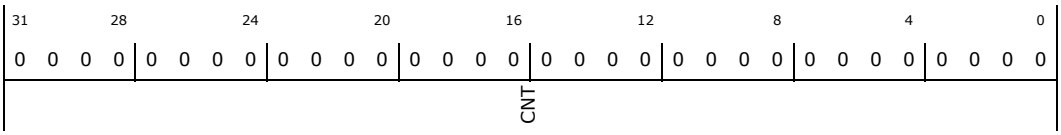
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 23Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.86 MMC Receive ICMP Good Frame Counter Register (RXICMP_GD_FRMS)—Offset 240h

Number of good IP datagrams with a good ICMP payload.

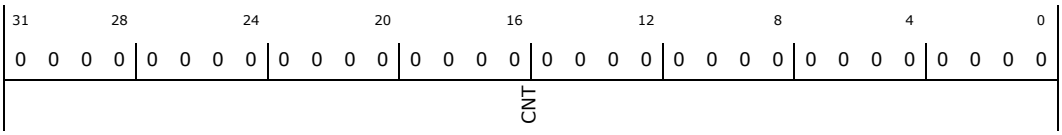
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 240h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.87 MMC Receive ICMP Error Frame Counter Register (RXICMP_ERR_FRMS)—Offset 244h

Number of good IP datagrams whose ICMP payload has a checksum error.

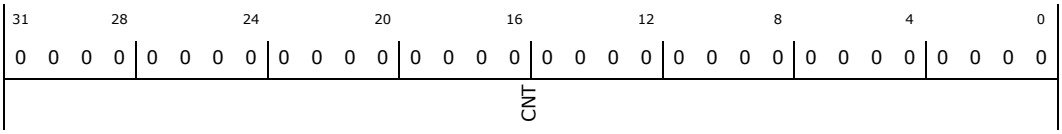
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 244h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.88 MMC Receive IPV4 Good Octet Counter Register (RXIPV4_GD_OCTETS)—Offset 250h

Number of bytes received in good IPv4 datagrams encapsulating TCP, UDP, or ICMP data. (Ethernet header, FCS, pad, or IP pad bytes are not included in this counter or in the octet counters listed below).

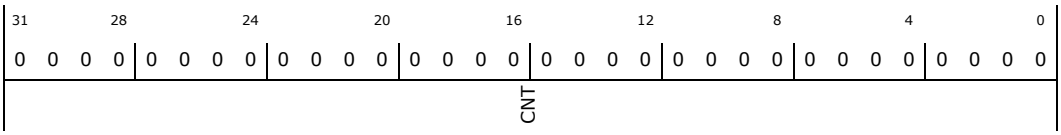
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 250h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.89 MMC Receive IPV4 Header Error Octet Counter Register (RXIPV4_HDRERR_OCTETS)—Offset 254h

Number of bytes received in IPv4 datagrams with header errors (checksum, length, version mismatch). The value in the Length field of IPv4 header is used to update this counter.

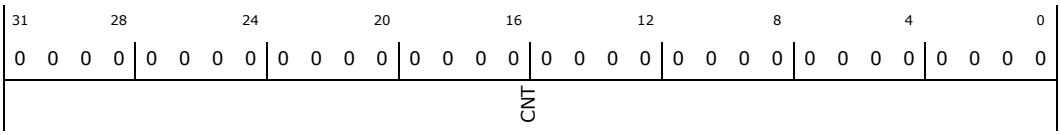
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 254h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.90 MMC Receive IPV4 No Payload Octet Counter Register (RXIPV4_NOPAY_OCTETS)—Offset 258h

Number of bytes received in IPv4 datagrams that did not have a TCP, UDP, or ICMP payload. The value in the IPv4 headers Length field is used to update this counter.

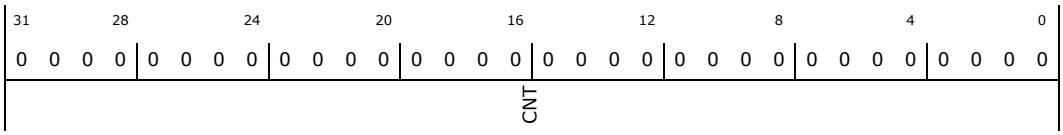
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 258h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.91 MMC Receive IPV4 Fragmented Octet Counter Register (RXIPV4_FRAG_OCTETS)—Offset 25Ch

Number of bytes received in fragmented IPv4 datagrams. The value in the IPv4 headers Length field is used to update this counter.

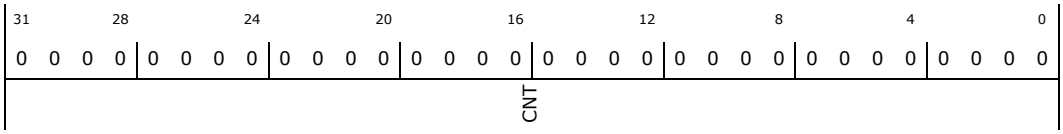
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 25Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.92 MMC Receive IPV4 UDP Checksum Disabled Octet Counter Register (RXIPV4_UDSBL_OCTETS)—Offset 260h

Number of bytes received in a UDP segment that had the UDP checksum disabled. This counter does not count IP Header bytes.

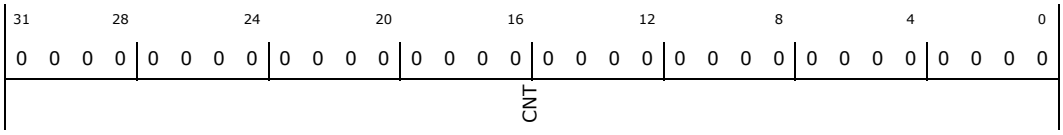
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 260h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.93 MMC Receive IPV6 Good Octet Counter Register (RXIPV6_GD_OCTETS)—Offset 264h

Number of bytes received in good IPv6 datagrams encapsulating TCP, UDP or ICMPv6 data.

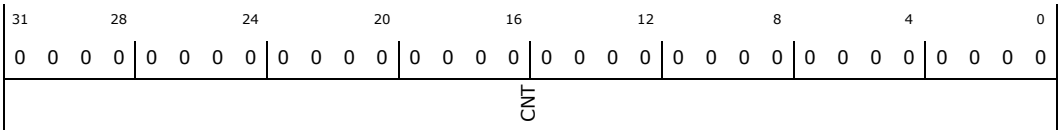
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 264h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.94 MMC Receive IPV6 Good Octet Counter Register (RXIPV6_HDRERR_OCTETS)—Offset 268h

Number of bytes received in IPv6 datagrams with header errors (length, version mismatch). The value in the IPv6 headers Length field is used to update this counter.

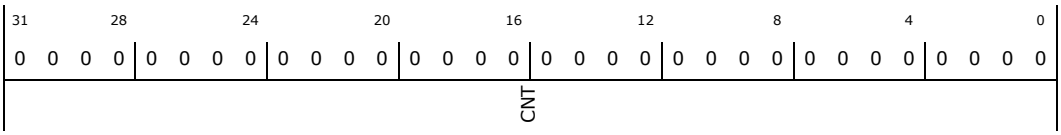
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 268h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.95 MMC Receive IPV6 Header Error Octet Counter Register (RXIPV6_NOPAY_OCTETS)—Offset 26Ch

Number of bytes received in IPv6 datagrams that did not have a TCP, UDP, or ICMP payload. The value in the IPv6 headers Length field is used to update this counter.

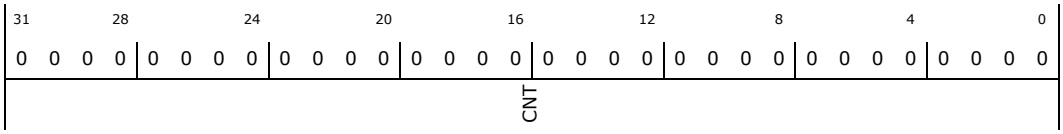
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 26Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.96 MMC Receive IPV6 No Payload Octet Counter Register (RXUDP_GD_OCTETS)—Offset 270h

Number of bytes received in a good UDP segment. This counter (and the counters below) does not count IP header bytes.

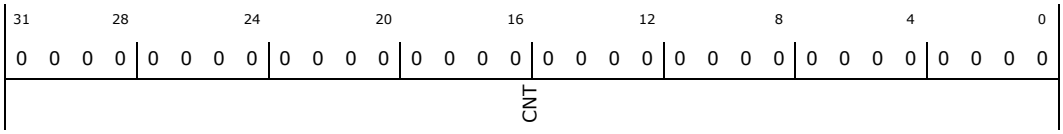
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 270h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.97 MMC Receive UDP Good Octet Counter Register (RXUDP_ERR_OCTETS)—Offset 274h

Number of bytes received in a UDP segment that had checksum errors.

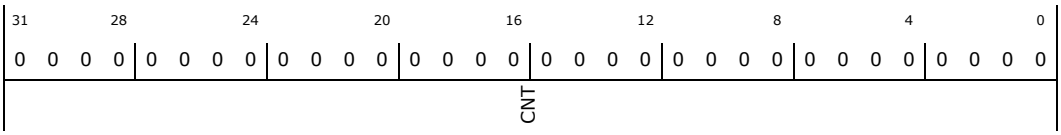
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 274h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.98 MMC Receive TCP Good Octet Counter Register (RXTCP_GD_OCTETS)—Offset 278h

Number of bytes received in a good TCP segment.

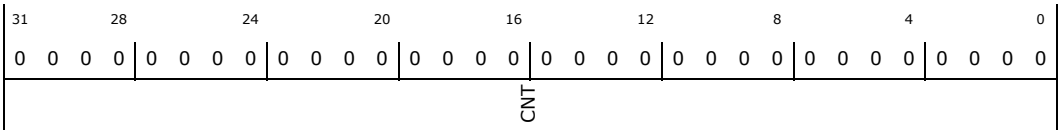
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 278h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.99 MMC Receive TCP Error Octet Counter Register (RXTCP_ERR_OCTETS)—Offset 27Ch

Number of bytes received in a TCP segment with checksum errors.

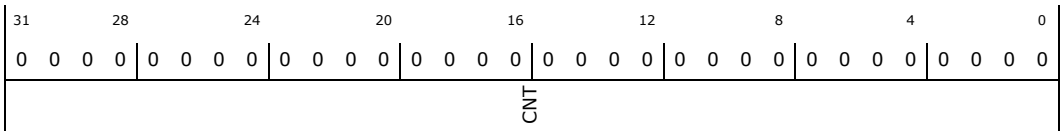
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 27Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.100 MMC Receive ICMP Good Octet Counter Register (RXICMP_GD_OCTETS)—Offset 280h

Number of bytes received in a good ICMP segment.

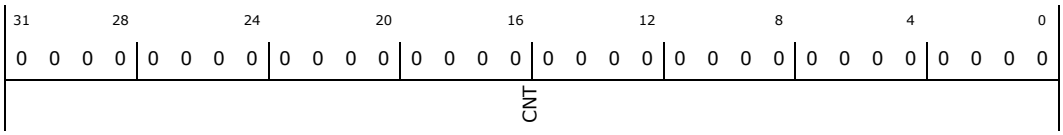
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 280h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.101 MMC Receive ICMP Error Octet Counter Register (RXICMP_ERR_OCTETS)—Offset 284h

Number of bytes received in an ICMP segment with checksum errors.

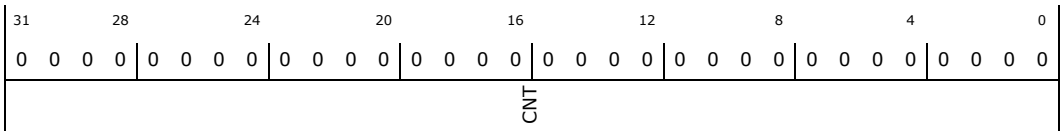
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 284h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Counter value (CNT):** Reserved.

15.6.102 VLAN Tag Inclusion or Replacement Register (Register 353) (GMAC_REG_353)—Offset 584h

The VLAN Tag Inclusion or Replacement register contains the VLAN tag for insertion or replacement in the transmit frames.

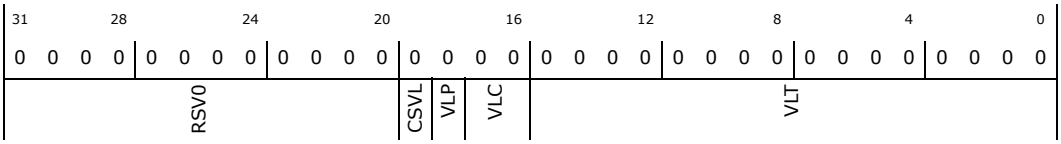
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 584h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	000h RO	**Reserved (RSV0):** Reserved.
19	0b RW	**C-VLAN or S-VLAN (CSVL):** When this bit is set, S-VLAN type (0x88A8) is inserted or replaced in the 13th and 14th bytes of transmitted frames. When this bit is reset, C-VLAN type (0x8100) is inserted or replaced in the transmitted frames.
18	0b RW	**VLAN Priority Control (VLP):** When this bit is set, the control Bits [17:16] are used for VLAN deletion, insertion, or replacement. When this bit is reset, the internal control signal from the MTL layer is used, and Bits [17:16] are ignored.
17:16	00b RW	**VLAN Tag Control in Transmit Frames (VLC):** 2'b00: No VLAN tag deletion, insertion, or replacement 2'b01: VLAN tag deletion. The MAC removes the VLAN type (bytes 13 and 14) and VLAN tag (bytes 15 and 16) of all transmitted frames with VLAN tags. 2'b10: VLAN tag insertion. The MAC inserts VLT in bytes 15 and 16 of the frame after inserting the Type value (0x8100/0x88a8) in bytes 13 and 14. This operation is performed on all transmitted frames, irrespective of whether they already have a VLAN tag. 2'b11: VLAN tag replacement. The MAC replaces VLT in bytes 15 and 16 of all VLAN-type transmitted frames (Bytes 13 and 14 are 0x8100/0x88a8). NOTE: Changes to this field take effect only on the start of a frame. If you write this register field when a frame is being transmitted, only the subsequent frame can use the updated value, that is, the current frame does not use the updated value.
15:0	0000h RW	**VLAN Tag for Transmit Frames (VLT):** This field contains the value of the VLAN tag to be inserted or replaced. The value must only be changed when the transmit lines are inactive or during the initialization phase. Bits[15:13] are the User Priority, Bit 12 is the CFI/DEI, and Bits[11:0] are the VLAN tags VID field.

15.6.103 VLAN Hash Table Register (Register 354) (GMAC_REG_354)—Offset 588h

The 16-bit Hash table is used for group address filtering based on VLAN tag when Bit 18 (VTHM) of Register 7 (VLAN Tag Register) is set. For hash filtering, the content of the 16-bit VLAN tag or 12-bit VLAN ID (based on Bit 16 (ETV) of VLAN Tag Register) in the incoming frame is passed through the CRC logic and the upper four bits of the calculated CRC are used to index the contents of the VLAN Hash table. For example, a hash value of 4b'1000 selects Bit 8 of the VLAN Hash table. The hash value of the destination address is calculated in the following way: 1. Calculate the 32-bit CRC for the VLAN tag or ID (See IEEE 802.3, Section 3.2.8 for the steps to calculate CRC32). 2. Perform bitwise reversal for the value obtained in Step 1. 3. Take the upper four bits from the value obtained in Step 2. If the corresponding bit value of the register is 1'b1, the frame is accepted. Otherwise, it is rejected.

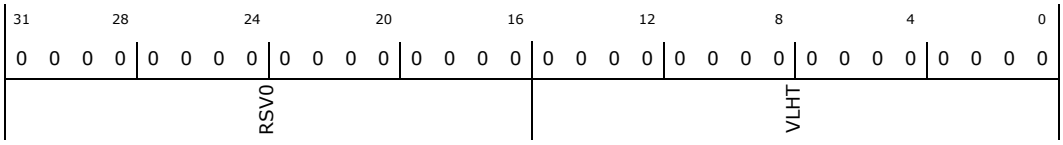
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 588h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSV0):** Reserved.
15:0	0000h RW	**VLAN Hash Table (VLHT):** This field contains the 16-bit VLAN Hash Table.

15.6.104 Timestamp Control Register (Register 448) (GMAC_REG_448)—Offset 700h

This register controls the operation of the System Time generator and the processing of PTP packets for timestamping in the Receiver.

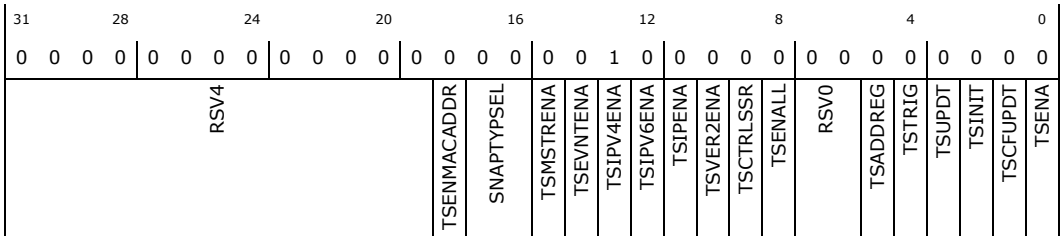
Access Method

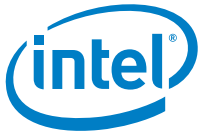
Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 700h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00002000h





Bit Range	Default & Access	Field Name (ID): Description
31:19	0b RO	**Reserved (RSV4):** Reserved.
18	0b RW	**Enable MAC address for PTP Frame Filtering (TSENMACADDR):** When set, the DA MAC address (that matches any MAC Address register) is used to filter the PTP frames when PTP is directly sent over Ethernet.
17:16	00b RW	**Select PTP packets for Taking Snapshots (SNAPTYPSEL):** These bits along with Bits 15 and 14 decide the set of PTP packet types for which snapshot needs to be taken.
15	0b RW	**Enable Snapshot for Messages Relevant to Master (TSMSTRENA):** When set, the snapshot is taken only for the messages relevant to the master node. Otherwise, the snapshot is taken for the messages relevant to the slave node.
14	0b RW	**Enable Timestamp Snapshot for Event Messages (TSEVNTENA):** When set, the timestamp snapshot is taken only for event messages (SYNC, Delay_Req, Pdelay_Req, or Pdelay_Resp). When reset, the snapshot is taken for all messages except Announce, Management, and Signaling.
13	1b RW	**Enable Processing of PTP Frames Sent over IPv4-UDP (TSIPV4ENA):** When set, the MAC receiver processes the PTP packets encapsulated in UDP over IPv4 packets. When this bit is clear, the MAC ignores the PTP transported over UDP-IPv4 packets. This bit is set by default.
12	0b RW	**Enable Processing of PTP Frames Sent Over IPv6-UDP (TSIPV6ENA):** When set, the MAC receiver processes PTP packets encapsulated in UDP over IPv6 packets. When this bit is clear, the MAC ignores the PTP transported over UDP-IPv6 packets.
11	0b RW	**Enable Processing of PTP over Ethernet Frames (TSIPENA):** When set, the MAC receiver processes the PTP packets encapsulated directly in the Ethernet frames. When this bit is clear, the MAC ignores the PTP over Ethernet packets.
10	0b RW	**Enable PTP packet Processing for Version 2 Format (TSVER2ENA):** When set, the PTP packets are processed using the 1588 version 2 format. Otherwise, the PTP packets are processed using the version 1 format.
9	0b RW	**Timestamp Digital or Binary Rollover Control (TSCTRLSSR):** When set, the Timestamp Low register rolls over after 0x3B9A_C9FF value (that is, 1 nanosecond accuracy) and increments the timestamp (High) seconds. When reset, the rollover value of sub-second register is 0x7FFF_FFFF. The sub-second increment has to be programmed correctly depending on the PTP reference clock frequency and the value of this bit.
8	0b RW	**Enable Timestamp for All Frames (TSENALL):** When set, the timestamp snapshot is enabled for all frames received by the MAC.
7:6	00b RO	**Reserved (RSV0):** Reserved.
5	0b RW	**Addend Reg Update (TSADDREG):** When set, the content of the Timestamp Addend register is updated in the PTP block for fine correction. This is cleared when the update is completed. This register bit should be zero before setting it.
4	0b RW	**Timestamp Interrupt Trigger Enable (TSTRIG):** When set, the timestamp interrupt is generated when the System Time becomes greater than the value written in the Target Time register. This bit is reset after the generation of the Timestamp Trigger Interrupt.
3	0b RW	**Timestamp Update (TSUPDT):** When set, the system time is updated (added or subtracted) with the value specified in Register 452 (System Time - Seconds Update Register) and Register 453 (System Time - Nanoseconds Update Register). This bit should be read zero before updating it. This bit is reset when the update is completed in hardware. The Timestamp Higher Word register (if enabled during core configuration) is not updated.
2	0b RW	**Timestamp Initialize (TSINIT):** When set, the system time is initialized (overwritten) with the value specified in the Register 452 (System Time - Seconds Update Register) and Register 453 (System Time - Nanoseconds Update Register). This bit should be read zero before updating it. This bit is reset when the initialization is complete. The Timestamp Higher Word register (if enabled during core configuration) can only be initialized.

Bit Range	Default & Access	Field Name (ID): Description
1	0b RW	**Timestamp Fine or Coarse Update (TSCFUPDT):** When set, this bit indicates that the system times update should be done using the fine update method. When reset, it indicates the system timestamp update should be done using the Coarse method.
0	0b RW	**Timestamp Enable (TSENA):** When set, the timestamp is added for the transmit and receive frames. When disabled, timestamp is not added for the transmit and receive frames and the Timestamp Generator is also suspended. You need to initialize the Timestamp (system time) after enabling this mode. On the receive side, the MAC processes the 1588 frames only if this bit is set.

15.6.105 Sub-Second Increment Register (Register 449) (GMAC_REG_449)—Offset 704h

In the Coarse Update mode (TSCFUPDT bit in Register 448), the value in this register is added to the system time every clock cycle of the internal 50MHz PTP reference clock. In the Fine Update mode, the value in this register is added to the system time whenever the Accumulator gets an overflow.

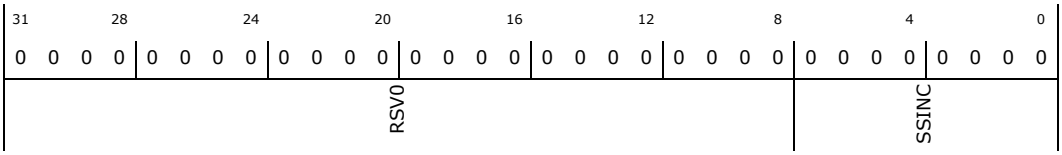
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 704h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	000000h RO	**Reserved (RSV0):** Reserved.
7:0	00h RW	**Sub-second Increment Value (SSINC):** The value programmed in this field is accumulated every clock cycle (of clk_ptp_i) with the contents of the sub-second register. For example, when PTP clock is 50 MHz (period is 20 ns), you should program 20 (0x14) when the System Time-Nanoseconds register has an accuracy of 1 ns (TSCTRLSSR bit is set). When TSCTRLSSR is clear, the Nanoseconds register has a resolution of ~0.465ns. In this case, you should program a value of 43 (0x2B) that is derived by 20ns/0.465.

15.6.106 System Time - Seconds Register (Register 450) (GMAC_REG_450)—Offset 708h

The System Time -Seconds register, along with System-TimeNanoseconds register, indicates the current value of the system time maintained by the MAC. Though it is updated on a continuous basis, there is some delay from the actual time because of clock domain transfer latencies.

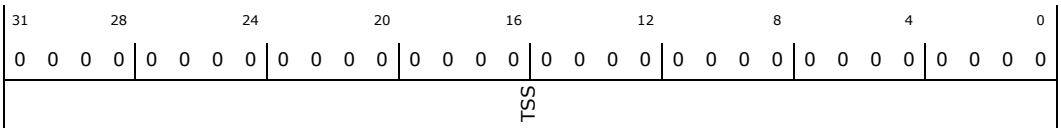
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 708h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Timestamp Second (TSS):** The value in this field indicates the current value in seconds of the System Time maintained by the MAC.

15.6.107 System Time - Nanoseconds Register (Register 451) (GMAC_REG_451)—Offset 70Ch

The value in this field has the sub second representation of time, with an accuracy of 0.46 ns. When TSCTRLSSR is set, each bit represents 1 ns and the maximum value is 0x3B9A_C9FF, after which it rolls-over to zero.

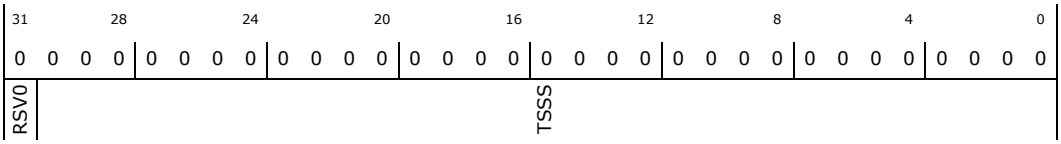
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 70Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Reserved (RSV0):** Reserved.
30:0	00000000h RO	**Timestamp Sub Seconds (TSSS):** The value in this field has the sub second representation of time, with an accuracy of 0.46 ns. When bit 9 (TSCTRLSSR) is set in Register 448 (Timestamp Control Register), each bit represents 1 ns and the maximum value is 0x3B9A_C9FF, after which it rolls-over to zero.

15.6.108 System Time - Seconds Update Register (Register 452) (GMAC_REG_452)—Offset 710h

The System Time - Seconds Update register, along with the System Time - Nanoseconds Update register, initializes or updates the system time maintained by the MAC. You must write both of these registers before setting the TSINIT or TSUPDT bits in the Timestamp Control register.

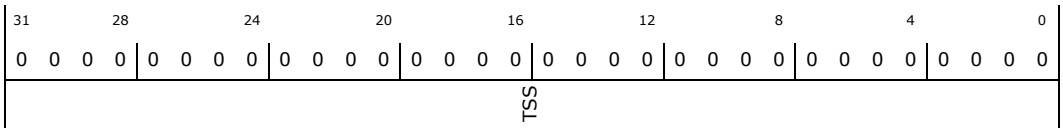
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 710h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Timestamp Second (TSS):** The value in this field indicates the time in seconds to be initialized or added to the system time.

15.6.109 System Time - Nanoseconds Update Register (Register 453) (GMAC_REG_453)—Offset 714h

The System Time - Nanoseconds Update register, along with the System Time - Seconds Update register, initializes or updates the system time maintained by the MAC. You must write both of these registers before setting the TSINIT or TSUPDT bits in the Timestamp Control register.

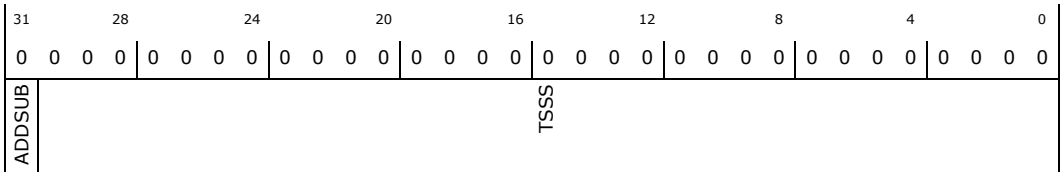
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 714h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Add or subtract time (ADDSUB):** When this bit is set, the time value is subtracted with the contents of the update register. When this bit is reset, the time value is added with the contents of the update register.
30:0	00000000h RW	**Timestamp Sub Second (TSSS):** The value in this field has the sub second representation of time, with an accuracy of 0.46 ns. When bit 9 (TSCTRLSSR) is set in Register 448 (Timestamp Control Register), each bit represents 1 ns and the programmed value should not exceed 0x3B9A_C9FF.

15.6.110 Timestamp Addend Register (Register 454) (GMAC_REG_454)—Offset 718h

This register value is used only when the system time is configured for Fine Update mode (TSCFUPDT bit in Register 448). This register content is added to a 32-bit accumulator in every clock cycle (of the internal 50MHz PTP reference clock) and the system time is updated whenever the accumulator overflows.

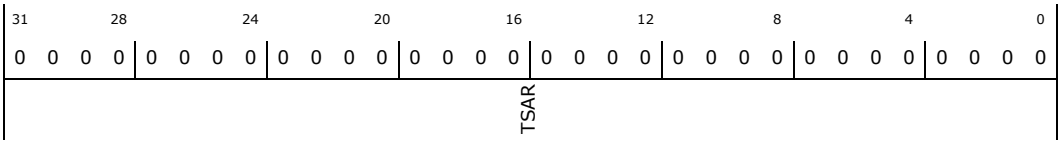
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 718h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Timestamp Addend (TSAR):** This field indicates the 32-bit time value to be added to the Accumulator register to achieve time synchronization.

15.6.111 Target Time Seconds Register (Register 455) (GMAC_REG_455)—Offset 71Ch

The Target Time Seconds register, along with Target Time Nanoseconds register, is used to schedule an interrupt event (Register 458[1] when Advanced Timestamping is enabled; otherwise, TS interrupt bit in Register14[9]) when the system time exceeds the value programmed in these registers.

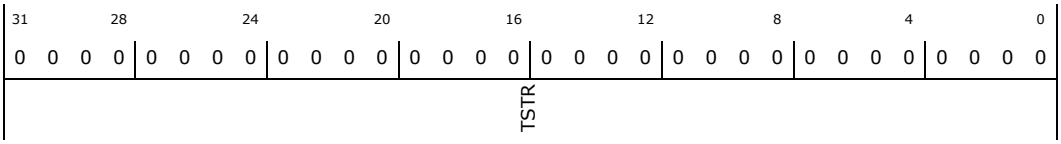
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 71Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RW	**Target Time Seconds (TSTR):** This field stores the time in seconds. When the timestamp value matches or exceeds both Target Timestamp registers, then based on Bits [6:5] of Register 459 (PPS Control Register), the MAC starts or stops the PPS signal output and generates an interrupt (if enabled).

15.6.112 Target Time Nanoseconds Register (Register 456) (GMAC_REG_456)—Offset 720h

This register is present only when the IEEE 1588 Timestamp feature is selected without external timestamp input.

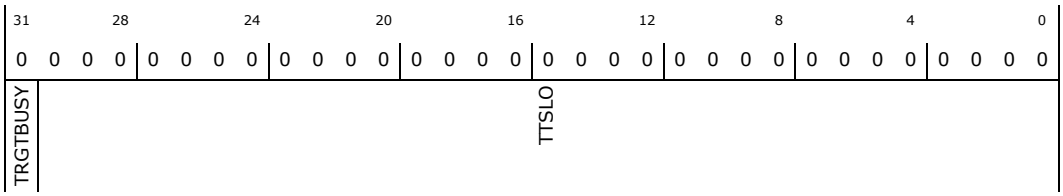
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 720h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Target Time Register Busy (TRGTBUSY):** The MAC sets this bit when the PPSCMD field (Bits[3:0]) in Register 459 (PPS Control Register) is programmed to 010 or 011. Programming the PPSCMD field to 010 or 011, instructs the MAC to synchronize the Target Time Registers to the PTP clock domain. The MAC clears this bit after synchronizing the Target Time Registers to the PTP clock domain The application must not update the Target Time Registers when this bit is read as 1. Otherwise, the synchronization of the previous programmed time gets corrupted. This bit is reserved when the Enable Flexible Pulse-Per-Second Output feature is not selected.
30:0	00000000h RW	**Target Timestamp Low (TTSLO):** The Target Time Nanoseconds register, along with Target Time Seconds register, is used to schedule an interrupt event (Register 458[1] when Advanced Timestamping is enabled; otherwise, TS interrupt bit in Register14[9]) when the system time exceeds the value programmed in these registers.

15.6.113 System Time - Higher Word Seconds Register (Register 457) (GMAC_REG_457)—Offset 724h

Contains the most significant 16-bits of the timestamp seconds value.

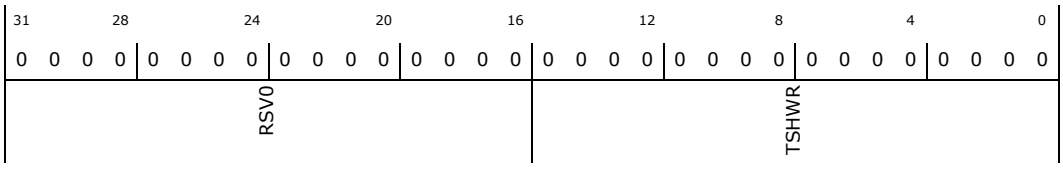
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 724h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0000h RO	**Reserved (RSV0):** Reserved.
15:0	0000h RW	**Timestamp Higher Word (TSHWR):** This field contains the most significant 16-bits of the timestamp seconds value. The register is directly written to initialize the value. This register is incremented when there is an overflow from the 32-bits of the System Time - Seconds register.

15.6.114 Timestamp Status Register (Register 458) (GMAC_REG_458)—Offset 728h

All non reserved bits are cleared when the host reads this register.

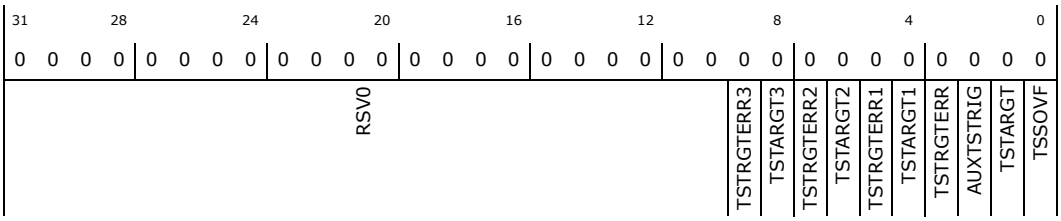
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 728h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV0):** Reserved.
9	0b RO/CR	**Timestamp Target Time Error (TSTRGTERR3):** This bit is set when the target time, being programmed in Register 496 and Register 497, is already elapsed. This bit is cleared when read by the application.
8	0b RO/CR	**Timestamp Target Time Reached for Target Time PPS3 (TSTARGT3):** When set, this bit indicates that the value of system time is greater than or equal to the value specified in Register 496 (PPS3 Target Time High Register) and Register 497 (PPS3 Target Time Low Register).
7	0b RO/CR	**Timestamp Target Time Error (TSTRGTERR2):** This bit is set when the target time, being programmed in Register 488 and Register 489, is already elapsed. This bit is cleared when read by the application.
6	0b RO/CR	**Timestamp Target Time Reached for Target Time PPS2 (TSTARGT2):** When set, this bit indicates that the value of system time is greater than or equal to the value specified in Register 488 (PPS2 Target Time High Register) and Register 489 (PPS2 Target Time Low Register).
5	0b RO/CR	**Timestamp Target Time Error (TSTRGTERR1):** This bit is set when the target time, being programmed in Register 480 and Register 481, is already elapsed. This bit is cleared when read by the application.
4	0b RO/CR	**Timestamp Target Time Reached for Target Time PPS1 (TSTARGT1):** When set, this bit indicates that the value of system time is greater than or equal to the value specified in Register 480 (PPS1 Target Time High Register) and Register 481 (PPS1 Target Time Low Register).
3	0b RO/CR	**Timestamp Target Time Error (TSTRGTERR):** This bit is set when the target time, being programmed in Target Time Registers, is already elapsed. This bit is cleared when read by the application.
2	0b RO	**Reserved (AUXTSTRIG):** Reserved.
1	0b RO/CR	**Timestamp Target Time Reached (TSTARGT):** When set, this bit indicates that the value of system time is greater or equal to the value specified in the Register 455 (Target Time Seconds Register) and Register 456 (Target Time Nanoseconds Register).
0	0b RO/CR	**Timestamp Seconds Overflow (TSSOVF):** When set, this bit indicates that the seconds value of the timestamp (when supporting version 2 format) has overflowed beyond 32'hFFFF_FFFF.

15.6.115 Bus Mode Register (Register 0) (DMA_REG_0)—Offset 1000h

The Bus Mode register establishes the bus operating modes for the DMA.

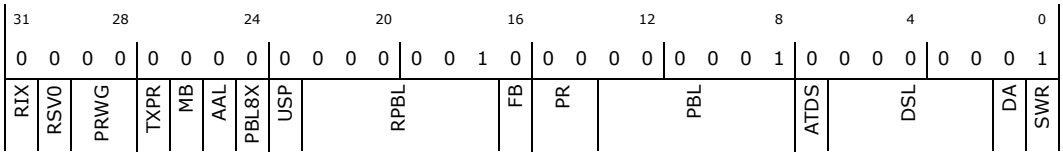
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

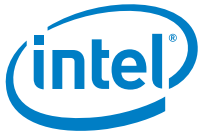
Offset: [BAR0] + 1000h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00020101h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**RIX:** Rebuild INCRx Burst When this bit is set high and the AHB master gets an EBT (Retry, Split, or Losing bus grant), the AHB master interface rebuilds the pending beats of any burst transfer initiated with INCRx. The AHB master interface rebuilds the beats with a combination of specified bursts with INCRx and SINGLE. By default, the AHB master interface rebuilds pending beats of an EBT with an unspecified (INCR) burst. This bit is valid only in the GMAC-AHB configuration. It is reserved in all other configuration.
30	0b RO	**Reserved (RSV0):** Reserved.
29:28	00b RO	**Channel Priority Weights (PRWG):** This field sets the priority weights for Channel 0 during the round-robin arbitration between the DMA channels for the system bus. 00: The priority weight is 1. 01: The priority weight is 2. 10: The priority weight is 3. 11: The priority weight is 4.
27	0b RW	**Transmit Priority (TXPR):** When set, this bit indicates that the transmit DMA has higher priority than the receive DMA during arbitration for the system-side bus.
26	0b RW	**Mixed Burst (MB):** When this bit is set high and the FB bit is low, the AHB Master interface starts all bursts of length more than 16 with INCR (undefined burst) whereas it reverts to fixed burst transfers (INCRx and SINGLE) for burst length of 16 and less. NOTE: for bandwidth reason, it is recommended to avoid using mixed bursts. Recommended setting is MB=0, FB=1.
25	0b RW	**Address Aligned Beats (AAL):** When this bit is set high and the FB bit is equal to 1, the AHB interface generates all bursts aligned to the start address LS bits. If the FB bit is equal to 0, the first burst (accessing the data buffer's start address) is not aligned, but subsequent bursts are aligned to the address.
24	0b RW	**8xPBL Mode (PBL8X):** When set high, this bit multiplies the programmed PBL value (Bits[22:17] and Bits[13:8]) eight times. Therefore, the DMA transfers the data in 8, 16, 32, 64, 128, and 256 beats depending on the PBL value. NOTE: This bit function is not backward compatible. Before release 3.50a, this bit was 4xPBL.
23	0b RW	**Use Separate PBL (USP):** When set high, this bit configures the Rx DMA to use the value configured in Bits[22:17] as PBL. The PBL value in Bits[13:8] is applicable only to the Tx DMA operations. When reset to low, the PBL value in Bits[13:8] is applicable for both DMA engines.



Bit Range	Default & Access	Field Name (ID): Description
22:17	000001b RW	**Rx DMA PBL (RPBL):** This field indicates the maximum number of beats to be transferred in one Rx DMA transaction. This is the maximum value that is used in a single block Read or Write. The Rx DMA always attempts to burst as specified in the RPBL bit each time it starts a Burst transfer on the host bus. You can program RPBL with values of 1, 2, 4, 8, 16, and 32. Any other value results in undefined behavior. This field is valid and applicable only when USP is set high.
16	0b RW	**Fixed Burst (FB):** This bit controls whether the AHB or AXI Master interface performs fixed burst transfers or not. When set, the AHB interface uses only SINGLE, INCR4, INCR8, or INCR16 during start of the normal burst transfers. When reset, the AHB or AXI interface uses SINGLE and INCR burst transfer operations. NOTE: for bandwidth reason, it is recommended to avoid using mixed bursts. Recommended setting is MB=0, FB=1.
15:14	00b RW	**Priority Ratio (PR):** These bits control the priority ratio in the weighted round-robin arbitration between the Rx DMA and Tx DMA. These bits are valid only when Bit 1 (DA) is reset. The priority ratio is Rx:Tx or Tx:Rx depending on whether Bit 27 (TXPR) is reset or set. 00: The Priority Ratio is 1:1. 01: The Priority Ratio is 2:1. 10: The Priority Ratio is 3:1. 11: The Priority Ratio is 4:1.
13:8	000001b RW	**Programmable Burst Length (PBL):** These bits indicate the maximum number of beats to be transferred in one DMA transaction. This is the maximum value that is used in a single block Read or Write. The DMA always attempts to burst as specified in PBL each time it starts a Burst transfer on the host bus. PBL can be programmed with permissible values of 1, 2, 4, 8, 16, and 32. Any other value results in undefined behavior. When USP is set high, this PBL value is applicable only for Tx DMA transactions. If the number of beats to be transferred is more than 32, then perform the following steps: 1. Set the 8xPBL mode. 2. Set the PBL. For example, if the maximum number of beats to be transferred is 64, then first set 8xPBL to 1 and then set PBL to 8. All values up to 256 are allowed using a combination of PBL and 8xPBL. All PBL values are supported in the full-duplex mode and half-duplex modes.
7	0b RW	**Alternate (Enhanced) Descriptor Size (ATDS):** When set, the size of the alternate descriptor increases to 32 bytes (8 DWORDS). This is required when the Advanced Timestamp feature or the IPC Full Offload Engine (Type 2) is enabled in the receiver. The enhanced descriptor is not required if the Advanced Timestamp and IPC Full Checksum Offload (Type 2) features are not enabled. In such cases, you can use the 16 bytes descriptor to save 4 bytes of memory. When reset, the descriptor size reverts back to 4 DWORDs (16 bytes). This bit preserves the backward compatibility for the descriptor size.
6:2	00000b RW	**Descriptor Skip Length (DSL):** This bit specifies the number of Word, Dword, or Lword (depending on the 32-bit, 64-bit, or 128-bit bus) to skip between two unchained descriptors. The address skipping starts from the end of current descriptor to the start of next descriptor. When the DSL value is equal to zero, then the descriptor table is taken as contiguous by the DMA in Ring mode.
1	0b RW	**DMA Arbitration Scheme (DA):** This bit specifies the arbitration scheme between the transmit and receive paths of Channel 0. 0: Weighted round-robin with Rx:Tx or Tx:Rx. The priority between the paths is according to the priority specified in bits 15:14 (PR) and priority weights specified in Bit 27 (TXPR). 1: Fixed priority. The transmit path has priority over receive path when Bit 27 (TXPR) is set. Otherwise, receive path has priority over the transmit path.
0	1b RW	**Software Reset (SWR):** When this bit is set, the MAC DMA Controller resets the logic and all internal registers of the MAC. It is cleared automatically after the reset operation has completed in all of the MAC clock domains. Before reprogramming any register of the MAC, you should read a zero (0) value in this bit . NOTE: The reset operation is completed only when all resets in all active clock domains are de-asserted. Therefore, it is essential that all the PHY inputs clocks (applicable for the selected PHY interface) are present for the software reset completion.

15.6.116 Transmit Poll Demand Register (Register 1) (DMA_REG_1)—Offset 1004h

The Transmit Poll Demand register enables the Tx DMA to check whether or not the DMA owns the current descriptor. The Transmit Poll Demand command is given to wake up the Tx DMA if it is in the Suspend mode. The Tx DMA can go into the Suspend mode because of an Underflow error in a transmitted frame or the unavailability of descriptors owned by it. You can give this command anytime and the Tx DMA resets this command when it again starts fetching the current descriptor from host memory.

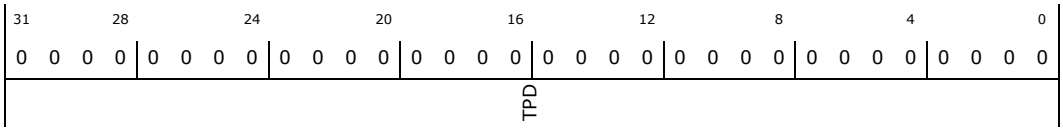
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1004h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000b RW	**Transmit Poll Demand (TPD):** When these bits are written with any value, the DMA reads the current descriptor pointed to by Register 18 (Current Host Transmit Descriptor Register). If that descriptor is not available (owned by the Host), the transmission returns to the Suspend state and the Bit 2 (TU) of Register 5 (Status Register) is asserted. If the descriptor is available, the transmission resumes.

15.6.117 Receive Poll Demand Register (Register 2) (DMA_REG_2)—Offset 1008h

The Receive Poll Demand register enables the receive DMA to check for new descriptors. This command is used to wake up the Rx DMA from the Suspend state. The RxDMA can go into the Suspend state only because of the unavailability of descriptors it owns.

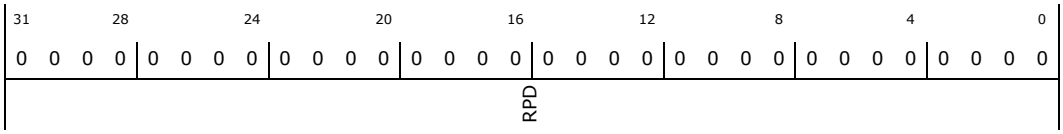
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1008h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000b RW	**Receive Poll Demand (RPD):** When these bits are written with any value, the DMA reads the current descriptor pointed to by Register 19 (Current Host Receive Descriptor Register). If that descriptor is not available (owned by the Host), the reception returns to the Suspended state and the Bit 7 (RU) of Register 5 (Status Register) is not asserted. If the descriptor is available, the Rx DMA returns to the active state.

15.6.118 Receive Descriptor List Address Register (Register 3) (DMA_REG_3)—Offset 100Ch

The Receive Descriptor List Address register points to the start of the Receive Descriptor List. The descriptor lists reside in the host's physical memory space and must be Word, Dword, or Lword-aligned (for 32-bit, 64-bit, or 128-bit data bus). The DMA internally converts it to bus width aligned address by making the corresponding LS bits low. Writing to this register is permitted only when reception is stopped. When stopped, this register must be written to before the receive Start command is given. You can write to this register only when Rx DMA has stopped, that is, Bit 1 (SR) is set to zero in Register 6 (Operation Mode Register). When stopped, this register can be written with a new descriptor list address. When you set the SR bit to 1, the DMA takes the newly programmed descriptor base address. If this register is not changed when the SR bit is set to 0, then the DMA takes the descriptor address where it was stopped earlier.

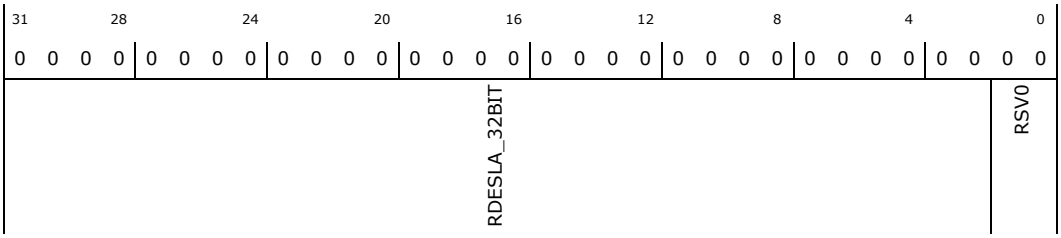
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 100Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	00000000h RW	**Start of Receive List (RDESLA_32BIT):** This field contains the base address of the first descriptor in the Receive Descriptor list. The LSB bits (1:0) for 32-bit bus width are ignored and internally taken as all-zero by the DMA. Therefore, these LSB bits are read-only (RO).
1:0	00b RO	**Reserved (RSV0):** Reserved.

15.6.119 Transmit Descriptor List Address Register (Register 4) (DMA_REG_4)—Offset 1010h

The Transmit Descriptor List Address register points to the start of the Transmit Descriptor List. The descriptor lists reside in the host's physical memory space and must be Word, Dword, or Lword-aligned (for 32-bit, 64-bit, or 128-bit data bus). The DMA internally converts it to bus width aligned address by making the corresponding

LSB to low. You can write to this register only when the Tx DMA has stopped, that is, Bit 13 (ST) is set to zero in Register 6 (Operation Mode Register). When stopped, this register can be written with a new descriptor list address. When you set the ST bit to 1, the DMA takes the newly programmed descriptor base address. If this register is not changed when the ST bit is set to 0, then the DMA takes the descriptor address where it was stopped earlier.

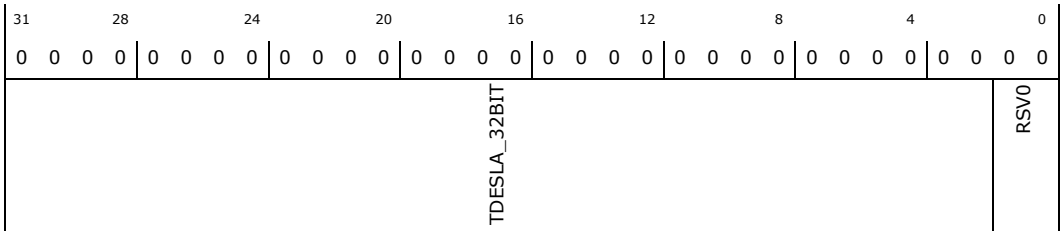
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1010h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	00000000h RW	**Start of Transmit List (TDESLA_32BIT):** This field contains the base address of the first descriptor in the Transmit Descriptor list. The LSB bits (1:0) for 32-bit bus width are ignored and are internally taken as all-zero by the DMA. Therefore, these LSB bits are read-only (RO).
1:0	00b RO	**Reserved (RSV0):** Reserved.

15.6.120 Status Register (Register 5) (DMA_REG_5)—Offset 1014h

The Status register contains all status bits that the DMA reports to the host. The Software driver reads this register during an interrupt service routine or polling. Most of the fields in this register cause the host to be interrupted. The bits of this register are not cleared when read. Writing 1'b1 to (unreserved) Bits[16:0] of this register clears these bits and writing 1'b0 has no effect. Each field (Bits[16:0]) can be masked by masking the appropriate bit in Register 7 (Interrupt Enable Register).

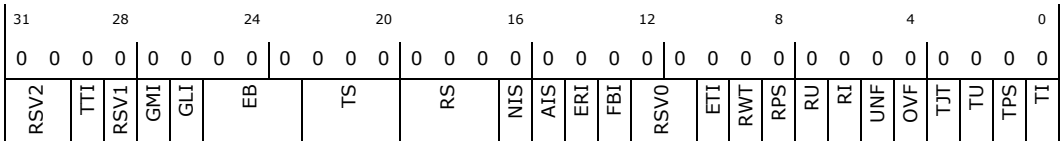
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1014h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0b RO	**Reserved (RSV2):** Reserved.
29	0b RO	**Timestamp Trigger Interrupt (TTI):** This bit indicates an interrupt event in the Timestamp Generator block of MAC. The software must read the corresponding registers in the MAC to get the exact cause of interrupt and clear its source to reset this bit to 1'b0. When this bit is high, the interrupt signal from the MAC subsystem is high.
28	0b RO	**Reserved (RSV1):** Reserved.
27	0b RO	**MAC MMC Interrupt (GMI):** This bit reflects an interrupt event in the MAC Management Counters (MMC) module. The software must read the corresponding registers in the MAC to get the exact cause of interrupt and clear the source of interrupt to make this bit as 1'b0. The interrupt signal from the MAC subsystem is high when this bit is high.
26	0b RO	**Reserved (GLI):** Reserved.
25:23	000b RO	**Error Bits (EB):** This field indicates the type of error that caused a Bus Error, for example, error response on the AHB or AXI interface. This field is valid only when Bit 13 (FBI) is set. This field does not generate an interrupt. * Bit 23 1'b1: Error during data transfer by the Tx DMA 1'b0: Error during data transfer by the Rx DMA * Bit 24 1'b1: Error during read transfer 1'b0: Error during write transfer * Bit 25 1'b1: Error during descriptor access 1'b0: Error during data buffer access
22:20	000b RO	**Transmit Process State (TS):** This field indicates the Transmit DMA FSM state. This field does not generate an interrupt. 3'b000: Stopped; Reset or Stop Transmit Command issued 3'b001: Running; Fetching Transmit Transfer Descriptor 3'b010: Running; Waiting for status 3'b011: Running; Reading Data from host memory buffer and queuing it to transmit buffer (Tx FIFO) 3'b100: TIME_STAMP write state 3'b101: Reserved for future use 3'b110: Suspended; Transmit Descriptor Unavailable or Transmit Buffer Underflow 3'b111: Running; Closing Transmit Descriptor
19:17	000b RO	**Received Process State (RS):** This field indicates the Receive DMA FSM state. This field does not generate an interrupt. 3'b000: Stopped: Reset or Stop Receive Command issued 3'b001: Running: Fetching Receive Transfer Descriptor 3'b010: Reserved for future use 3'b011: Running: Waiting for receive packet 3'b100: Suspended: Receive Descriptor Unavailable 3'b101: Running: Closing Receive Descriptor 3'b110: TIME_STAMP write state 3'b111: Running: Transferring the receive packet data from receive buffer to host memory
16	0b RW	**Normal Interrupt Summary (NIS):** Normal Interrupt Summary bit value is the logical OR of the following when the corresponding interrupt bits are enabled in Register 7 (Interrupt Enable Register): Register 5[0]: Transmit Interrupt Register 5[2]: Transmit Buffer Unavailable Register 5[6]: Receive Interrupt Register 5[14]: Early Receive Interrupt Only unmasked bits (interrupts for which interrupt enable is set in Register 7) affect the Normal Interrupt Summary bit. This is a sticky bit and must be cleared (by writing 1 to this bit) each time a corresponding bit, which causes NIS to be set, is cleared.

Bit Range	Default & Access	Field Name (ID): Description
15	0b RW	**Abnormal Interrupt Summary (AIS):** Abnormal Interrupt Summary bit value is the logical OR of the following when the corresponding interrupt bits are enabled in Register 7 (Interrupt Enable Register): Register 5[1]: Transmit Process Stopped Register 5[3]: Transmit Jabber Timeout Register 5[4]: Receive FIFO Overflow Register 5[5]: Transmit Underflow Register 5[7]: Receive Buffer Unavailable Register 5[8]: Receive Process Stopped Register 5[9]: Receive Watchdog Timeout Register 5[10]: Early Transmit Interrupt Register 5[13]: Fatal Bus Error Only unmasked bits affect the Abnormal Interrupt Summary bit. This is a sticky bit and must be cleared each time a corresponding bit, which causes AIS to be set, is cleared.
14	0b RW	**Early Receive Interrupt (ERI):** This bit indicates that the DMA had filled the first data buffer of the packet. Bit 6 (RI) of this register automatically clears this bit.
13	0b RW	**Fatal Bus Error Interrupt (FBI):** This bit indicates that a bus error occurred, as described in Bits[25:23]. When this bit is set, the corresponding DMA engine disables all of its bus accesses.
12:11	00b RO	**Reserved (RSV0):** Reserved.
10	0b RW	**Early Transmit Interrupt (ETI):** This bit indicates that the frame to be transmitted is fully transferred to the MTL Transmit FIFO.
9	0b RW	**Receive Watchdog Timeout (RWT):** This bit is asserted when a frame with length greater than 2,048 bytes is received (10, 240 when Jumbo Frame mode is enabled).
8	0b RW	**Receive Process Stopped (RPS):** This bit is asserted when the Receive Process enters the Stopped state.
7	0b RW	**Receive Buffer Unavailable (RU):** This bit indicates that the host owns the Next Descriptor in the Receive List and the DMA cannot acquire it. The Receive Process is suspended. To resume processing Receive descriptors, the host should change the ownership of the descriptor and issue a Receive Poll Demand command. If no Receive Poll Demand is issued, the Receive Process resumes when the next recognized incoming frame is received. This bit is set only when the previous Receive Descriptor is owned by the DMA.
6	0b RW	**Receive Interrupt (RI):** This bit indicates that the frame reception is complete. When reception is complete, the Bit 31 of RDES1 (Disable Interrupt on Completion) is reset in the last Descriptor, and the specific frame status information is updated in the descriptor. The reception remains in the Running state.
5	0b RW	**Transmit Underflow (UNF):** This bit indicates that the Transmit Buffer had an Underflow during frame transmission. Transmission is suspended and an Underflow Error TDES0[1] is set.
4	0b RW	**Receive Overflow (OVF):** This bit indicates that the Receive Buffer had an Overflow during frame reception. If the partial frame is transferred to the application, the overflow status is set in RDES0[11].
3	0b RW	**Transmit Jabber Timeout (TJT):** This bit indicates that the Transmit Jabber Timer expired, which happens when the frame size exceeds 2,048 (10,240 bytes when the Jumbo frame is enabled). When the Jabber Timeout occurs, the transmission process is aborted and placed in the Stopped state. This causes the Transmit Jabber Timeout TDES0[14] flag to assert.
2	0b RW	**Transmit Buffer Unavailable (TU):** This bit indicates that the host owns the Next Descriptor in the Transmit List and the DMA cannot acquire it. Transmission is suspended. Bits[22:20] explain the Transmit Process state transitions. To resume processing Transmit descriptors, the host should change the ownership of the descriptor by setting TDES0[31] and then issue a Transmit Poll Demand command.
1	0b RW	**Transmit Process Stopped (TPS):** This bit is set when the transmission is stopped.

Bit Range	Default & Access	Field Name (ID): Description
0	0b RW	**Transmit Interrupt (TI):** This bit indicates that the frame transmission is complete. When transmission is complete, the Bit 31 (Interrupt on Completion) of TDES1 is reset in the first descriptor, and the specific frame status information is updated in the descriptor.

15.6.121 Operation Mode Register (Register 6) (DMA_REG_6)—Offset 1018h

The Operation Mode register establishes the Transmit and Receive operating modes and commands. This register should be the last CSR to be written as part of the DMA initialization.

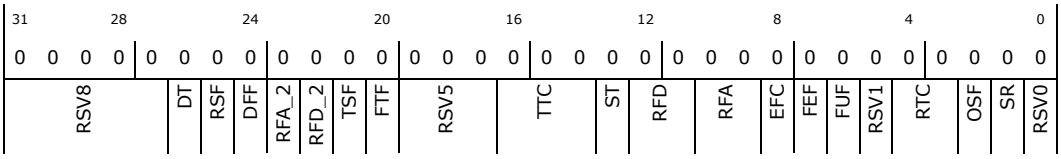
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1018h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:27	0h RO	**Reserved (RSV8):** Reserved.
26	0b RW	**Disable Dropping of TCP/IP Checksum Error Frames (DT):** When this bit is set, the MAC does not drop the frames which only have errors detected by the Receive Checksum Offload engine. Such frames do not have any errors (including FCS error) in the Ethernet frame received by the MAC but have errors only in the encapsulated payload. When this bit is reset, all error frames are dropped if the FEF bit is reset.
25	0b RW	**Receive Store and Forward (RSF):** When this bit is set, the MTL reads a frame from the Rx FIFO only after the complete frame has been written to it, ignoring the RTC bits. When this bit is reset, the Rx FIFO operates in the cut-through mode, subject to the threshold specified by the RTC bits.
24	0b RW	**Disable Flushing of Received Frames (DFF):** When this bit is set, the Rx DMA does not flush any frames because of the unavailability of receive descriptors or buffers as it does normally when this bit is reset.
23	0b RW	**MSB of Threshold for Activating Flow Control (RFA_2):** If the Rx FIFO depth is 8 KB or more, this bit (when set) provides additional threshold levels for activating the flow control in both half-duplex and full-duplex modes. This bit (as Most Significant Bit) along with the RFA (Bits[10:9]) gives the following thresholds for activating flow control: 100: Full minus 5 KB, that is, FULL - 5KB 101: Full minus 6 KB, that is, FULL - 6KB 110: Full minus 7 KB, that is, FULL - 7KB 111: Reserved

Bit Range	Default & Access	Field Name (ID): Description
22	0b RW	**MSB of Threshold for Deactivating Flow Control (RFD_2):** If the Rx FIFO size is 8 KB or more, this bit (when set) provides additional threshold levels for deactivating the flow control in both half-duplex and full-duplex modes. This bit (as Most Significant Bit) along with the RFD (Bits[12:11]) gives the following thresholds for deactivating flow control: 100: Full minus 5 KB, that is, FULL - 5KB 101: Full minus 6 KB, that is, FULL - 6KB 110: Full minus 7 KB, that is, FULL - 7KB 111: Reserved
21	0b RW	**Transmit Store and Forward (TSF):** When this bit is set, transmission starts when a full frame resides in the MTL Transmit FIFO. When this bit is set, the TTC values specified in Bits[16:14] are ignored. This bit should be changed only when the transmission is stopped.
20	0b RW	**Flush Transmit FIFO (FTF):** When this bit is set, the transmit FIFO controller logic is reset to its default values and thus all data in the Tx FIFO is lost or flushed. This bit is cleared internally when the flushing operation is completed. The Operation Mode register should not be written to until this bit is cleared. The data which is already accepted by the MAC transmitter is not flushed. It is scheduled for transmission and results in underflow and runt frame transmission. NOTE: The flush operation is complete only when the Tx FIFO is emptied of its contents and all the pending Transmit Status of the transmitted frames are accepted by the host. To complete this flush operation, the PHY transmit clock is required to be active.
19:17	000b RO	**Reserved (RSV5):** Reserved.
16:14	000b RW	**Transmit Threshold Control (TTC):** These bits control the threshold level of the MTL Transmit FIFO. Transmission starts when the frame size within the MTL Transmit FIFO is larger than the threshold. In addition, full frames with a length less than the threshold are also transmitted. These bits are used only when Bit 21 (TSF) is reset. 000: 64 001: 128 010: 192 011: 256 100: 40 101: 32 110: 24 111: 16
13	0b RW	**Start or Stop Transmission Command (ST):** When this bit is set, transmission is placed in the Running state, and the DMA checks the Transmit List at the current position for a frame to be transmitted. Descriptor acquisition is attempted either from the current position in the list, which is the Transmit List Base Address set by Register 4 (Transmit Descriptor List Address Register), or from the position retained when transmission was stopped previously. If the DMA does not own the current descriptor, transmission enters the Suspended state and Bit 2 (Transmit Buffer Unavailable) of Register 5 (Status Register) is set. The Start Transmission command is effective only when transmission is stopped. If the command is issued before setting Register 4 (Transmit Descriptor List Address Register), then the DMA behavior is unpredictable. When this bit is reset, the transmission process is placed in the Stopped state after completing the transmission of the current frame. The Next Descriptor position in the Transmit List is saved, and it becomes the current position when transmission is restarted. To change the list address, you need to program Register 4 (Transmit Descriptor List Address Register) with a new value when this bit is reset. The new value is considered when this bit is set again. The stop transmission command is effective only when the transmission of the current frame is complete or the transmission is in the Suspended state.
12:11	00b RW	**Threshold for Deactivating Flow Control (RFD):** These bits control the threshold (Fill-level of Rx FIFO) at which the flow control is de-asserted after activation (in half-duplex and full-duplex). 00: Full minus 1 KB, that is, FULL - 1KB 01: Full minus 2 KB, that is, FULL - 2KB 10: Full minus 3 KB, that is, FULL - 3KB 11: Full minus 4 KB, that is, FULL - 4KB The de-assertion is effective only after flow control is asserted. If the Rx FIFO is 8 KB or more, an additional bit (RFD[2]) is used for more threshold levels as described in Bit 22.

Bit Range	Default & Access	Field Name (ID): Description
10:9	00b RW	**Threshold for Activating Flow Control (RFA):** These bits control the threshold (Fill level of Rx FIFO) at which the flow control is activated (in half-duplex and full-duplex). 00: Full minus 1 KB, that is, FULL - 1KB 01: Full minus 2 KB, that is, FULL - 2KB 10: Full minus 3 KB, that is, FULL - 3KB 11: Full minus 4 KB, that is, FULL - 4KB These values only apply to Rx FIFOs of 4 KB or more when the EFC bit is set high. If the Rx FIFO is 8 KB or more, an additional bit (RFA[2]) is used for more threshold levels as described in Bit 23.
8	0b RW	**Enable HW Flow Control (EFC):** When this bit is set, the flow control signal operation based on the fill-level of Rx FIFO is enabled. When reset, the flow control operation is disabled. This bit is not used (reserved and always reset) when the Rx FIFO is less than 4 KB.
7	0b RW	**Forward Error Frames (FEF):** When this bit is reset, the Rx FIFO drops frames with error status (CRC error, collision error, GMII_ER, giant frame, watchdog timeout, or overflow). However, if the start byte (write) pointer of a frame is already transferred to the read controller side (in Threshold mode), then the frame is not dropped. When the FEF bit is set, all frames except runt error frames are forwarded to the DMA. If the Bit 25 (RSF) is set and the Rx FIFO overflows when a partial frame is written, then the frame is dropped irrespective of the FEF bit setting. However, if the Bit 25 (RSF) is reset and the Rx FIFO overflows when a partial frame is written, then a partial frame may be forwarded to the DMA.
6	0b RW	**Forward Undersized Good Frames (FUF):** When set, the Rx FIFO forwards Undersized frames (frames with no Error and length less than 64 bytes) including pad-bytes and CRC. When reset, the Rx FIFO drops all frames of less than 64 bytes, unless a frame is already transferred because of the lower value of Receive Threshold, for example, RTC = 01.
5	0b RO	**Reserved (RSV1):** Reserved.
4:3	00b RW	**Receive Threshold Control (RTC):** These two bits control the threshold level of the MTL Receive FIFO. Transfer (request) to DMA starts when the frame size within the MTL Receive FIFO is larger than the threshold. In addition, full frames with length less than the threshold are transferred automatically. These bits are valid only when the RSF bit is zero, and are ignored when the RSF bit is set to 1. 00: 64 01: 32 10: 96 11: 128
2	0b RW	**Operate on Second Frame (OSF):** When this bit is set, it instructs the DMA to process the second frame of the Transmit data even before the status for the first frame is obtained.
1	0b RW	**Start or Stop Receive (SR):** When this bit is set, the Receive process is placed in the Running state. The DMA attempts to acquire the descriptor from the Receive list and processes the incoming frames. The descriptor acquisition is attempted from the current position in the list, which is the address set by Register 3 (Receive Descriptor List Address Register) or the position retained when the Receive process was previously stopped. If the DMA does not own the descriptor, reception is suspended and Bit 7 (Receive Buffer Unavailable) of Register 5 (Status Register) is set. The Start Receive command is effective only when the reception has stopped. If the command is issued before setting Register 3 (Receive Descriptor List Address Register), the DMA behavior is unpredictable. When this bit is cleared, the Rx DMA operation is stopped after the transfer of the current frame. The next descriptor position in the Receive list is saved and becomes the current position after the Receive process is restarted. The Stop Receive command is effective only when the Receive process is in either the Running (waiting for receive packet) or in the Suspended state.
0	0b RO	**Reserved (RSV0):** Reserved.

15.6.122 Interrupt Enable Register (Register 7) (DMA_REG_7)—Offset 101Ch

The Interrupt Enable register enables the interrupts reported by Register 5 (Status Register). Setting a bit to 1'b1 enables a corresponding interrupt. After a hardware or software reset, all interrupts are disabled.

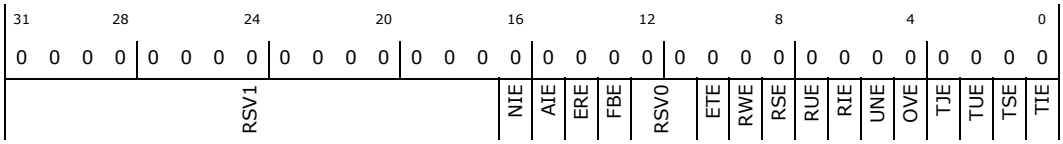
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 101Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:17	0000h RO	**Reserved (RSV1):** Reserved.
16	0b RW	**Normal Interrupt Summary Enable (NIE):** When this bit is set, normal interrupt summary is enabled. When this bit is reset, normal interrupt summary is disabled. This bit enables the following interrupts in Register 5 (Status Register): Register 5[0]: Transmit Interrupt Register 5[2]: Transmit Buffer Unavailable Register 5[6]: Receive Interrupt Register 5[14]: Early Receive Interrupt
15	0b RW	**Abnormal Interrupt Summary Enable (AIE):** When this bit is set, abnormal interrupt summary is enabled. When this bit is reset, the abnormal interrupt summary is disabled. This bit enables the following interrupts in Register 5 (Status Register): Register 5[1]: Transmit Process Stopped Register 5[3]: Transmit Jabber Timeout Register 5[4]: Receive Overflow Register 5[5]: Transmit Underflow Register 5[7]: Receive Buffer Unavailable Register 5[8]: Receive Process Stopped Register 5[9]: Receive Watchdog Timeout Register 5[10]: Early Transmit Interrupt Register 5[13]: Fatal Bus Error
14	0b RW	**Early Receive Interrupt Enable (ERE):** When this bit is set with Normal Interrupt Summary Enable (Bit 16), the Early Receive Interrupt is enabled. When this bit is reset, the Early Receive Interrupt is disabled.
13	0b RW	**Fatal Bus Error Enable (FBE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Fatal Bus Error Interrupt is enabled. When this bit is reset, the Fatal Bus Error Enable Interrupt is disabled.
12:11	00b RO	**Reserved (RSV0):** Reserved.
10	0b RW	**Early Transmit Interrupt Enable (ETE):** When this bit is set with an Abnormal Interrupt Summary Enable (Bit 15), the Early Transmit Interrupt is enabled. When this bit is reset, the Early Transmit Interrupt is disabled.
9	0b RW	**Receive Watchdog Timeout Enable (RWE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Receive Watchdog Timeout Interrupt is enabled. When this bit is reset, the Receive Watchdog Timeout Interrupt is disabled.
8	0b RW	**Receive Stopped Enable (RSE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Receive Stopped Interrupt is enabled. When this bit is reset, the Receive Stopped Interrupt is disabled.

Bit Range	Default & Access	Field Name (ID): Description
7	0b RW	**Receive Buffer Unavailable Enable (RUE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Receive Buffer Unavailable Interrupt is enabled. When this bit is reset, the Receive Buffer Unavailable Interrupt is disabled.
6	0b RW	**Receive Interrupt Enable (RIE):** When this bit is set with Normal Interrupt Summary Enable (Bit 16), the Receive Interrupt is enabled. When this bit is reset, the Receive Interrupt is disabled.
5	0b RW	**Underflow Interrupt Enable (UNE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Transmit Underflow Interrupt is enabled. When this bit is reset, the Underflow Interrupt is disabled.
4	0b RW	**Overflow Interrupt Enable (OVE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Receive Overflow Interrupt is enabled. When this bit is reset, the Overflow Interrupt is disabled.
3	0b RW	**Transmit Jabber Timeout Enable (TJE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Transmit Jabber Timeout Interrupt is enabled. When this bit is reset, the Transmit Jabber Timeout Interrupt is disabled.
2	0b RW	**Transmit Buffer Unavailable Enable (TUE):** When this bit is set with Normal Interrupt Summary Enable (Bit 16), the Transmit Buffer Unavailable Interrupt is enabled. When this bit is reset, the Transmit Buffer Unavailable Interrupt is disabled.
1	0b RW	**Transmit Stopped Enable (TSE):** When this bit is set with Abnormal Interrupt Summary Enable (Bit 15), the Transmission Stopped Interrupt is enabled. When this bit is reset, the Transmission Stopped Interrupt is disabled.
0	0b RW	**Transmit Interrupt Enable (TIE):** When this bit is set with Normal Interrupt Summary Enable (Bit 16), the Transmit Interrupt is enabled. When this bit is reset, the Transmit Interrupt is disabled.

15.6.123 Missed Frame and Buffer Overflow Counter Register (Register 8) (DMA_REG_8)—Offset 1020h

The DMA maintains two counters to track the number of frames missed during reception.

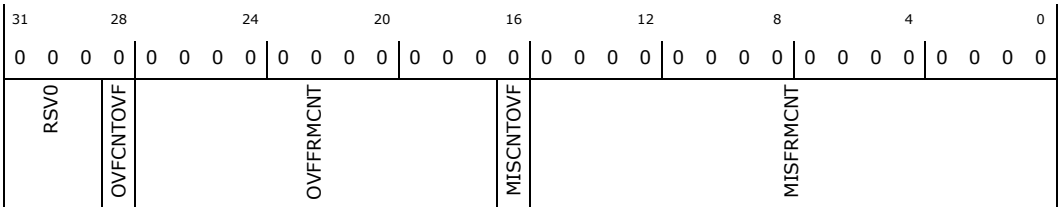
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1020h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	000b RO	**Reserved (RSV0):** Reserved.
28	0b RO	**FIFO Overflow Counter Overflow (OVFCNTOVF):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
27:17	000h RO	**FIFO Overflow Counter (OVFFRMCNT):** This field indicates the number of frames missed by the application due to buffer overflow conditions and runt frames (good frames of less than 64 bytes) dropped by the MTL. The counter is cleared when this register is read with the LS Byte enabled.
16	0b RO	**Missed Frame Counter Overflow (MISCNTOVF):** Reserved.
15:0	0000h RO	**Missed Frame Counter (MISFRMCNT):** This field indicates the number of frames missed by the controller because of the Host Receive Buffer being unavailable. This counter is incremented each time the DMA discards an incoming frame. The counter is cleared when this register is read with the LS Byte enabled.

15.6.124 Receive Interrupt Watchdog Timer Register (Register 9) (DMA_REG_9)—Offset 1024h

This register, when written with non-zero value, enables the watchdog timer for the Receive Interrupt (Bit 6) of Register 5 (Status Register).

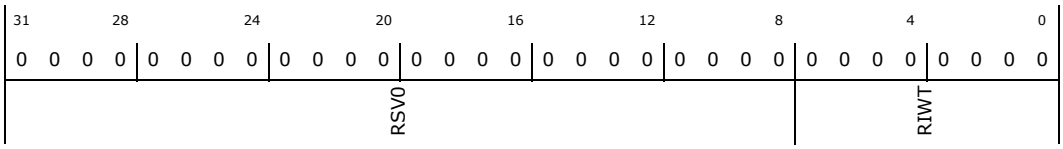
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1024h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	000000h RO	**Reserved (RSV0):** Reserved.
7:0	00h RW	**RI Watchdog Timer Count (RIWT):** This bit indicates the number of system clock cycles multiplied by 256 for which the watchdog timer is set. The watchdog timer gets triggered with the programmed value after the Rx DMA completes the transfer of a frame for which the RI status bit is not set because of the setting in the corresponding descriptor RDES1[31]. When the watchdog timer runs out, the RI bit is set and the timer is stopped. The watchdog timer is reset when the RI bit is set high because of automatic setting of RI as per RDES1[31] of any received frame.

15.6.125 AHB Status Register (Register 11) (DMA_REG_11)—Offset 102Ch

This register provides the active status of the AHB master interface interface's read and write channels. This register is useful for debugging purposes.

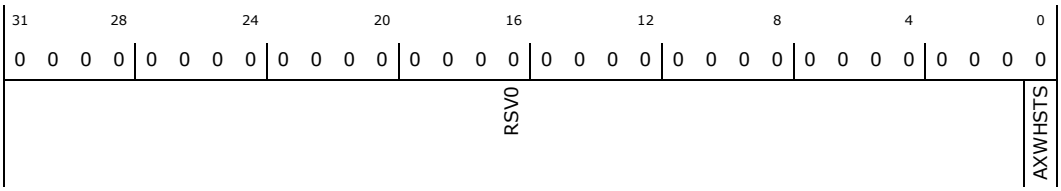
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 102Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	00000000h RO	**Reserved (RSV0):** Reserved.
0	0b RO	**AHB Master Status (AXWHSTS):** This bit indicates that the AHB master interface FSMs are in the non-idle state.

15.6.126 Current Host Transmit Descriptor Register (Register 18) (DMA_REG_18)—Offset 1048h

The Current Host Transmit Descriptor register points to the start address of the current Transmit Descriptor read by the DMA.

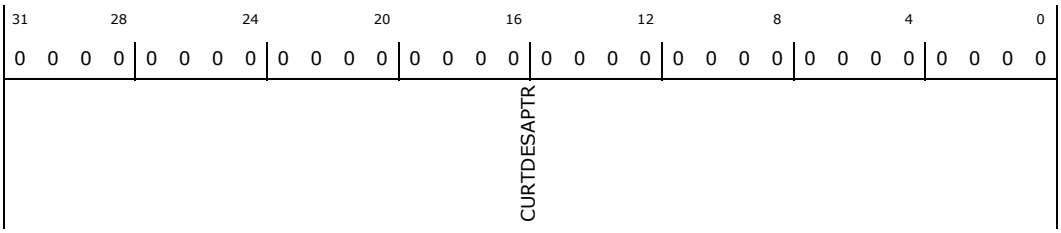
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1048h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Host Transmit Descriptor Address Pointer (CURTDESAPTR):** Cleared on Reset. Pointer updated by the DMA during operation.

15.6.127 Current Host Receive Descriptor Register (Register 19) (DMA_REG_19)—Offset 104Ch

The Current Host Receive Descriptor register points to the start address of the current Receive Descriptor read by the DMA.

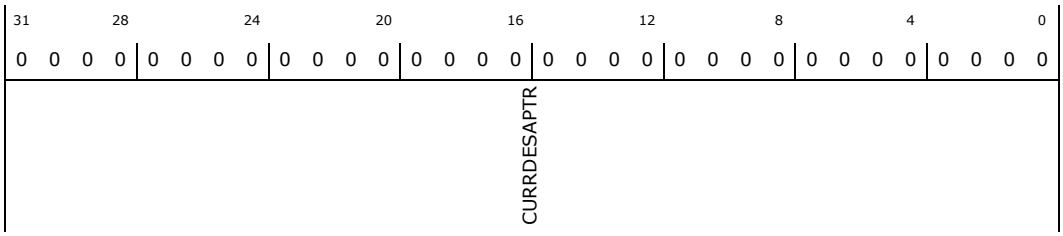
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 104Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Host Receive Descriptor Address Pointer (CURRDESAPTR):** Cleared on Reset. Pointer updated by the DMA during operation.

15.6.128 Current Host Transmit Buffer Address Register (Register 20) (DMA_REG_20)—Offset 1050h

The Current Host Transmit Buffer Address register points to the current Transmit Buffer Address being read by the DMA.

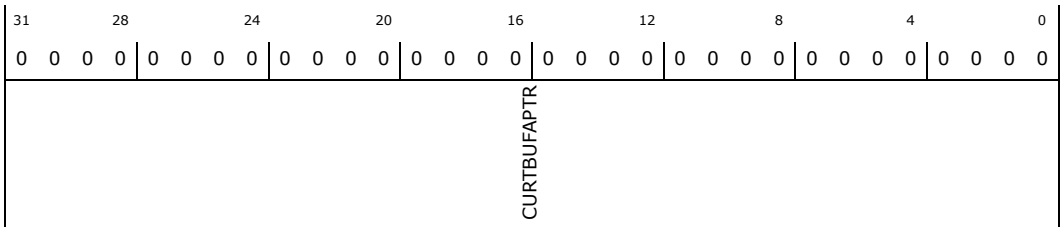
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1050h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Host Transmit Buffer Address Pointer (CURTBUFAPTR):** Cleared on Reset. Pointer updated by the DMA during operation.

15.6.129 Current Host Receive Buffer Address Register (Register 21) (DMA_REG_21)—Offset 1054h

The Current Host Receive Buffer Address register points to the current Receive Buffer address being read by the DMA.

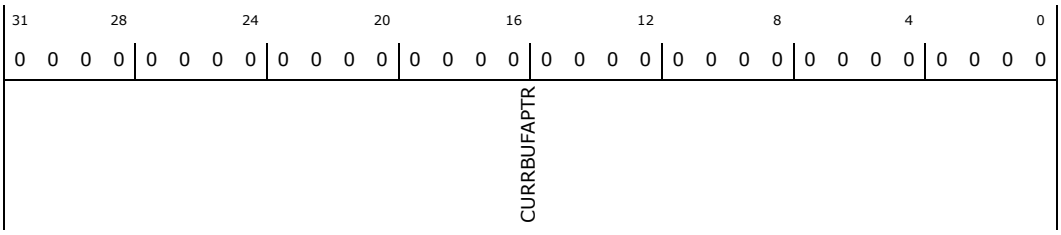
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1054h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	00000000h RO	**Host Receive Buffer Address Pointer (CURRBUFAPTR):** Cleared on Reset. Pointer updated by the DMA during operation.

15.6.130 HW Feature Register (Register 22) (DMA_REG_22)—Offset 1058h

This register indicates the presence of the optional features or functions of the MAC. Set field indicates the feature is supported. The software driver can use this register to dynamically enable or disable the programs related to the optional blocks.

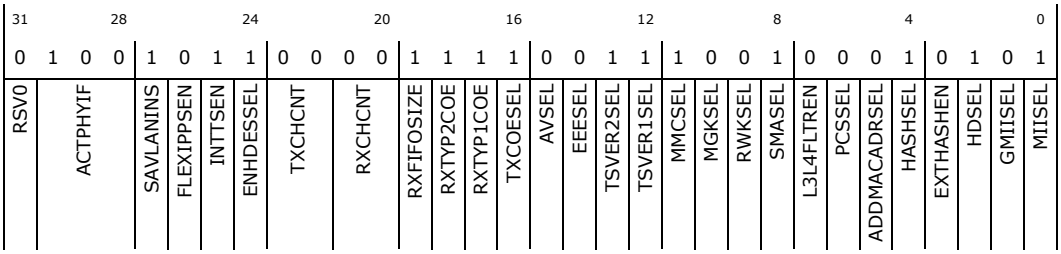
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1058h

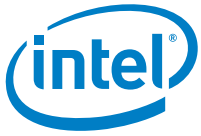
BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:6] + 10h

Default: 4B0F3915h



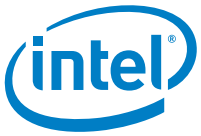
Bit Range	Default & Access	Field Name (ID): Description
31	0b RO	**Reserved (RSV0):** Reserved.
30:28	100b RO	**Active or Selected PHY interface (ACTPHYIF):** This field indicates the supported PHY interface: 0000: GMII or MII 0001: RGMII 0010: SGMII 0011: TBI 0100: RMII 0101: RTBI 0110: SMII 0111: RevMII All Others: Reserved
27	1b RO	**Source Address or VLAN Insertion (SAVLANINS):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
26	0b RO	**Flexible Pulse-Per-Second Output (FLEXIPPSEN):** Reserved.
25	1b RO	**Timestamping with Internal System Time (INTTSEN):** Reserved.
24	1b RO	**Alternate (Enhanced Descriptor) (ENHDESSEL):** Reserved.
23:22	00b RO	**Number of additional Tx channels (TXCHCNT):** Reserved.
21:20	00b RO	**Number of additional Rx channels (RXCHCNT):** Reserved.
19	1b RO	**Rx FIFO > 2,048 Bytes (RXFIFOSIZE):** Reserved.
18	1b RO	**IP Checksum Offload (Type 2) in Rx (RXTYP2COE):** Reserved.
17	1b RO	**IP Checksum Offload (Type 1) in Rx (RXTYP1COE):** Reserved.
16	1b RO	**Checksum Offload in Tx (TXCOESEL):** Reserved.
15	0b RO	**AV Feature (AVSEL):** Reserved.
14	0b RO	**Energy Efficient Ethernet (EEESEL):** Reserved.
13	1b RO	**IEEE 1588-2008 Advanced Timestamp (TSVER2SEL):** Reserved.
12	1b RO	**Only IEEE 1588-2002 Timestamp (TSVER1SEL):** Reserved.
11	1b RO	**RMON Module (MMCSEL):** Reserved.
10	0b RO	**PMT Magic Packet (MGKSEL):** Reserved.
9	0b RO	**PMT Remote Wakeup (RWKSEL):** Reserved.
8	1b RO	**SMA (MDIO) Interface (SMASEL):** Reserved.
7	0b RO	**L3L4FLTREN:** Reserved.
6	0b RO	**PCS registers (PCSSEL):** Reserved.
5	0b RO	**Multiple MAC Address Registers (ADDMACADRSEL):** Reserved.
4	1b RO	**HASH Filter (HASHSEL):** Reserved.
3	0b RO	**Expanded DA Hash Filter (EXTHASHEN):** Reserved.
2	1b RO	**Half-Duplex support (HDSEL):** Reserved.
1	0b RO	**1000 Mbps Support (GMIISEL):** Reserved.



Bit Range	Default & Access	Field Name (ID): Description
0	1b RO	**10 and 100 Mbps Support (MIISEL):** Reserved.

§ §



October 2013
Document Number: 329676-001US

16.0 USB 2.0

The Intel® Quark SoC X1000 USB subsystem provides a two-port USB 2.0 Host Controller and one USB 2.0 Device port.

16.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 85. Signals

Signal Name	Direction/ Type	Description
USBH[0/1]_DP USBH[0/1]_DN	I/O	Universal Serial Bus Host Port 0 and Port 1. Differentials: Bus Data/ Address/Command Bus
USBH0_OC_B USBH1_OC_B	I	Over current Indicators: These signals set corresponding bits in the USB controller to indicate that an over current condition has occurred. Overcurrent indicators are provided for both Host ports.
USBH0_PWR_EN USBH1_PWR_EN	O	Power Enable signal to the USB host port
USBD_DP USBD_DN	I/O	Universal Serial Bus Device Port. Differentials: Bus Data/ Address/ Command Bus
OUSBCOMP	0	RCOMP OUT. Note: Please check the Platform Design Guide for connection details for this COMP pin.
IUSBCOMP	I	RCOMP IN. Note: Please check the Platform Design Guide for connection details for this COMP pin.

16.2 Features

16.2.1 USB2.0 Host Controller Features

- 2-Port USB 2.0 Host Controller compatible with the following standards:
  - — Universal Serial Bus Specification (Revision 2.0, April 27, 2000)
  - — Enhanced Host Controller Interface Specification for Universal Serial Bus (Revision1.0, March 12, 2002)
  - — EHCI 1.1 Addendum (Revision v0.6, October 2007)
  - — OpenHCI: Open Host Controller Interface Specification for USB (Release 1.0a, September 14, 1999)

- EHCI features
  Supported**:**
  — 512-byte Packet Buffer depth for in/out data buffering
  — Programmable Packet Buffer depth
  — Extended capability pointer (EECP = 8'hC0)
  — Programmable frame list flag
  — 32-bit only addressing capability
  — Per port power control
  — PCI Power Management
  Not supported:
  — Descriptor/data prefetching
  — Asynchronous schedule park capability
  — HSIC functionality
  — Link Power Management (LPM) ECN
- OHCI features
  Supported:
  — One OHCI companion controller
  — Per port power control
  Not supported:
  — Keyboard/Mouse legacy interface

16.2.2 USB2.0 Device Features

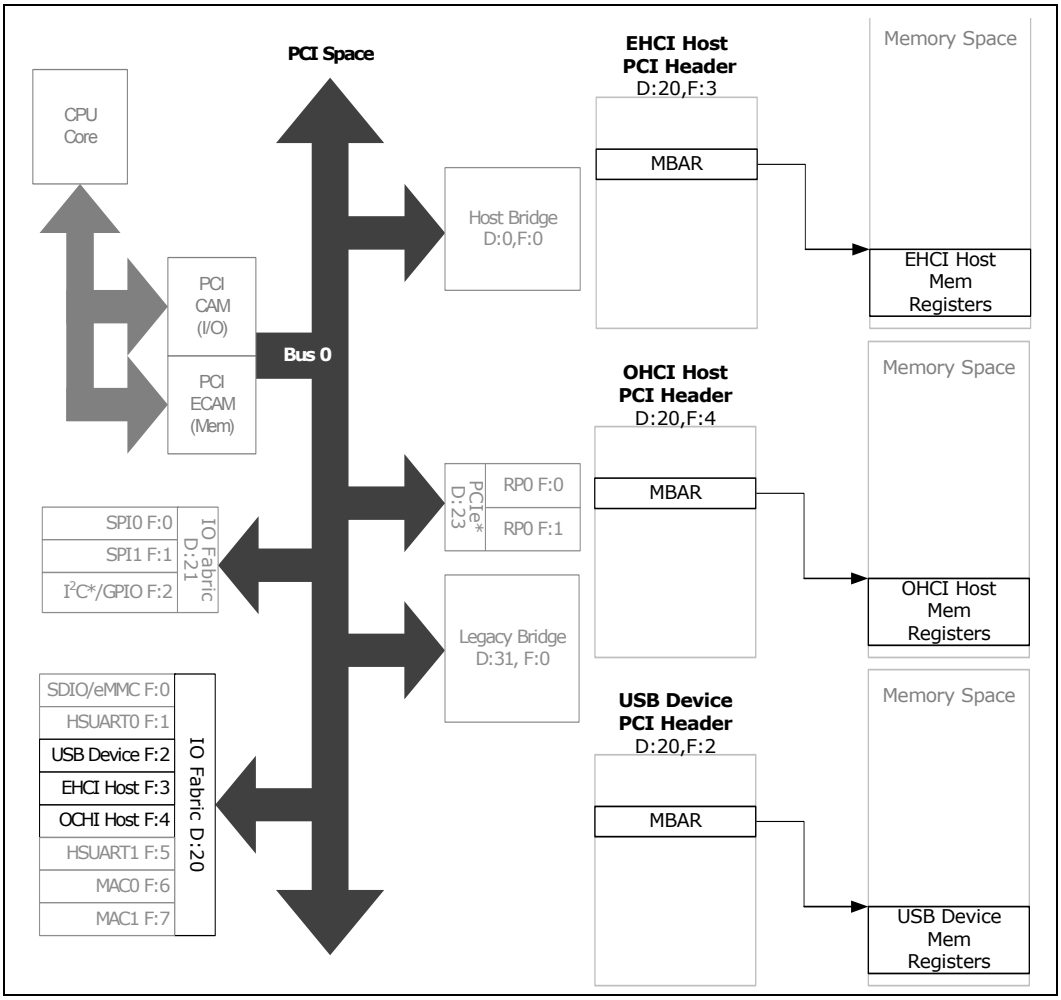
- High-speed (480 MHz)
- 3 logical endpoints in addition to logical endpoint 0
- 1 configuration in addition to configuration 0
- Enables user-configurable endpoint information
- Multiple data packets for each OUT endpoint (Multiple Receive FIFO).
- Both DMA option and Slave-Only modes
- True scatter-gather DMA implementation
- Descriptor-based memory structures in application memory when in DMA mode

16.3 References

- USB 2.0 specification at http://www.usb.org/developers/docs

16.4 Register Map

Figure 28. USB Register Map



See Chapter 5.0, "Register Access Methods" for additional information.

16.5 PCI Configuration Registers

16.5.1 USB Device

Table 86. Summary of PCI Configuration Registers—0/20/2

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 424	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 425	0939h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 425	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 426	0010h

Table 86. Summary of PCI Configuration Registers—0/20/2 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 426	0C03FE10h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 427	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 427	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 428	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 428	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 429	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 429	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 430	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 430	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 430	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 431	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 431	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 432	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 432	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 432	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 433	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 433	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 433	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 434	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 435	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 435	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 436	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 436	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 436	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 437	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 437	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 438	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 438	00000000h

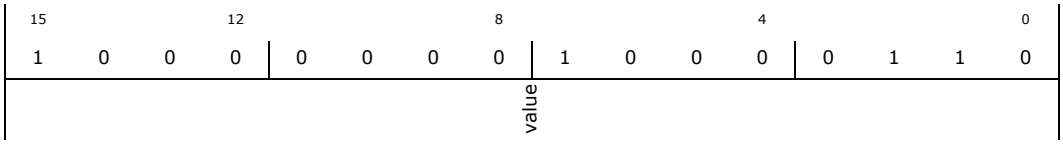
16.5.1.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:2] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

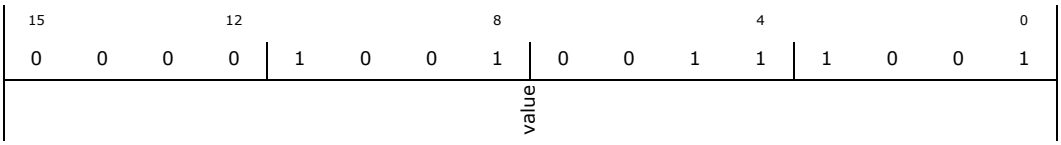
16.5.1.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:2] + 2h

Default: 0939h



Bit Range	Default & Access	Description
15: 0	0939h RO	**Device ID (value):** PCI Device ID

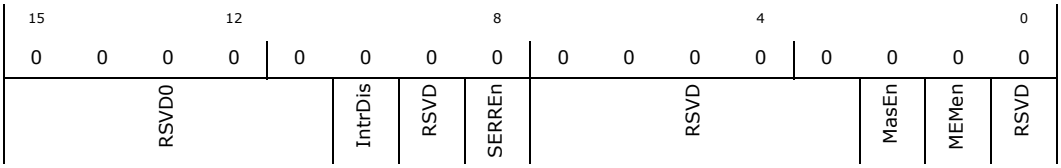
16.5.1.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:2] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.

Bit Range	Default & Access	Description
0	0h RO	**Reserved (RSVD):** Reserved.

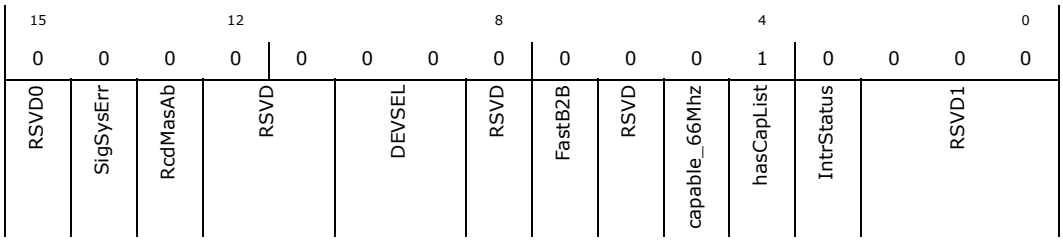
16.5.1.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:2] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

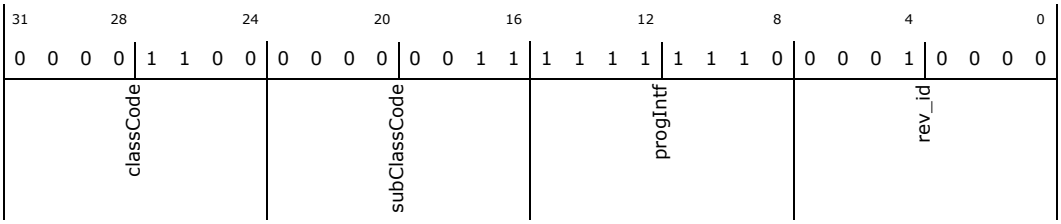
16.5.1.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:2] + 8h

Default: 0C03FE10h



Bit Range	Default & Access	Description
31: 24	0Ch RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	03h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	FEh RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

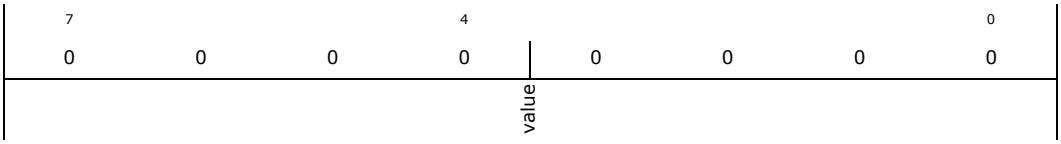
16.5.1.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register
(Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:2] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

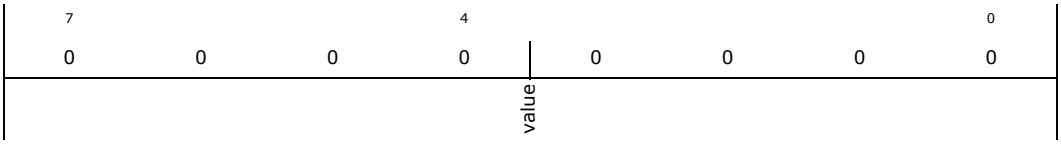
16.5.1.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register
(Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:2] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

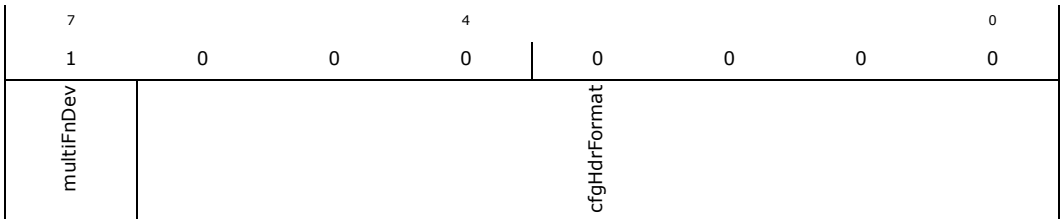
16.5.1.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:2] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

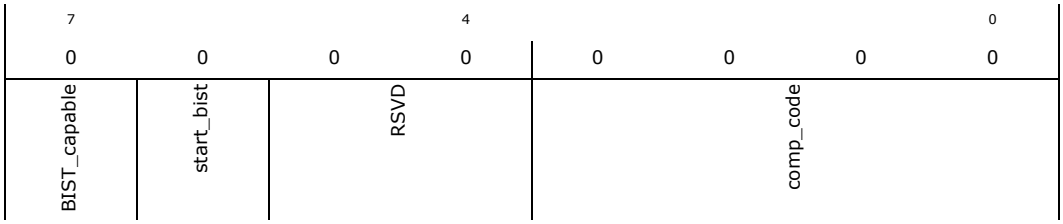
16.5.1.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:2] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

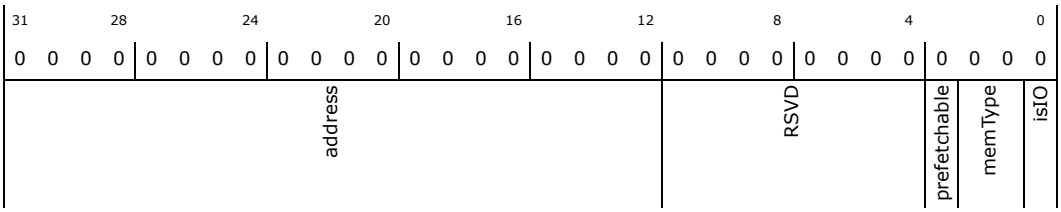
16.5.1.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

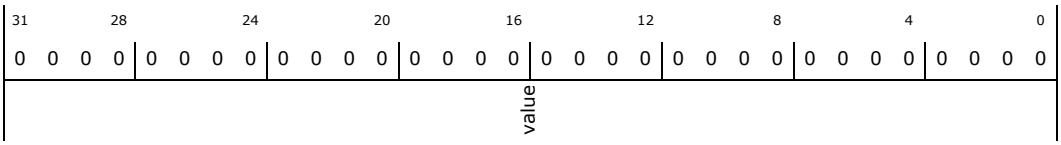
16.5.1.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:2] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

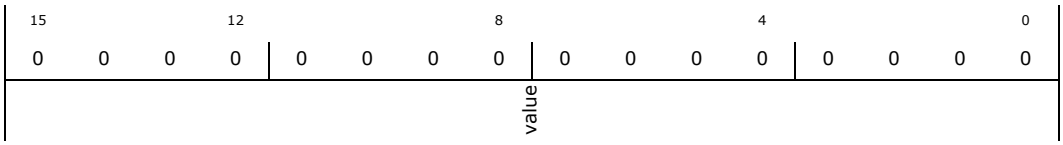
16.5.1.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:2] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

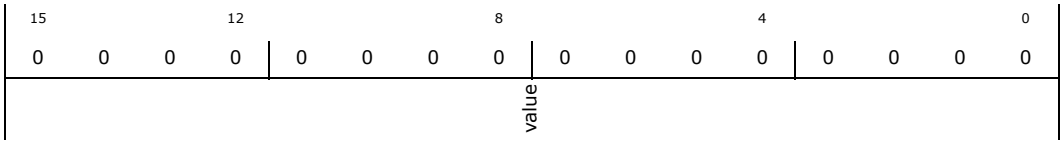
16.5.1.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:2] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

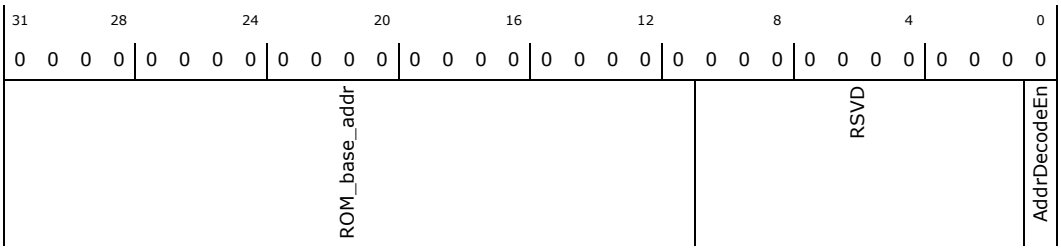
16.5.1.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:2] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

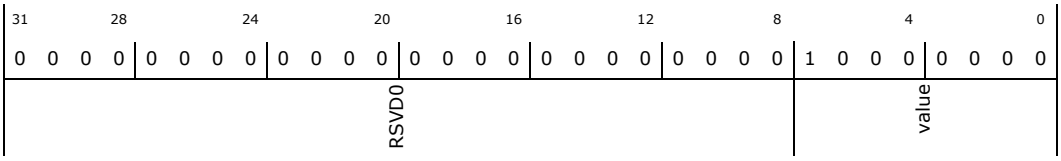
16.5.1.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:2] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

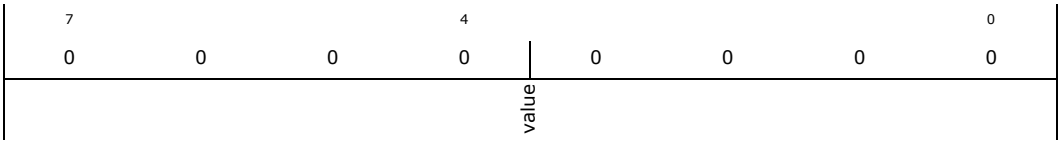
16.5.1.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:20, F:2] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

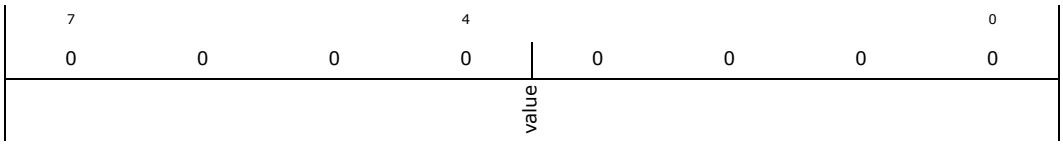
16.5.1.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:20, F:2] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

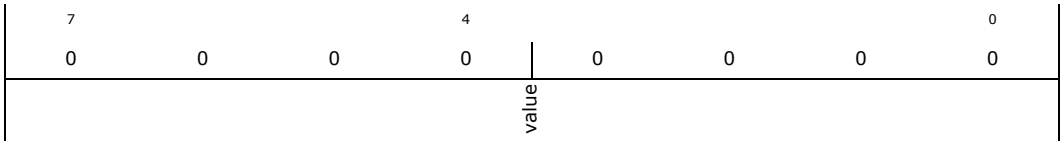
16.5.1.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:2] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

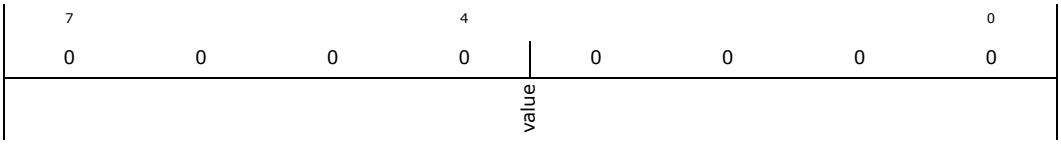
16.5.1.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:2] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

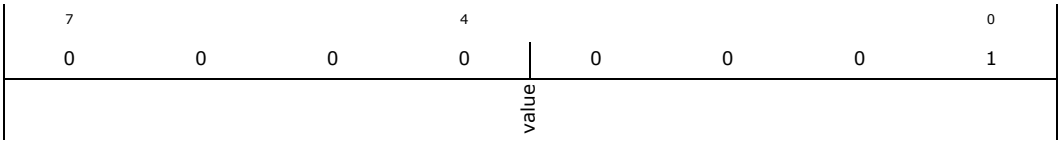
16.5.1.20 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:2] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

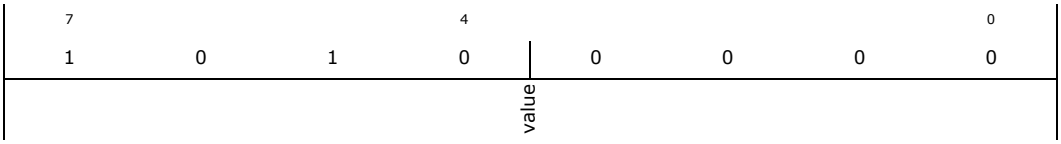
16.5.1.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:2] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

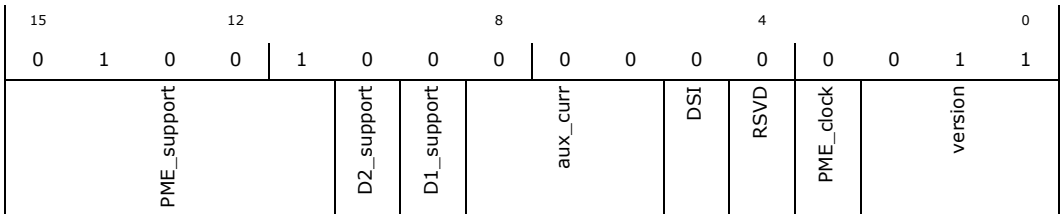
16.5.1.22 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:2] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

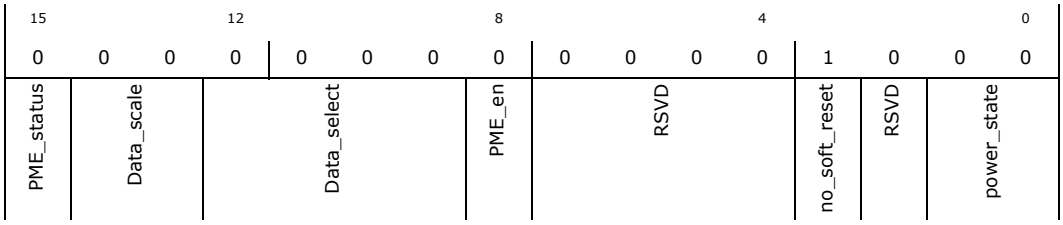
16.5.1.23 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:2] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported

Bit Range	Default & Access	Description
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

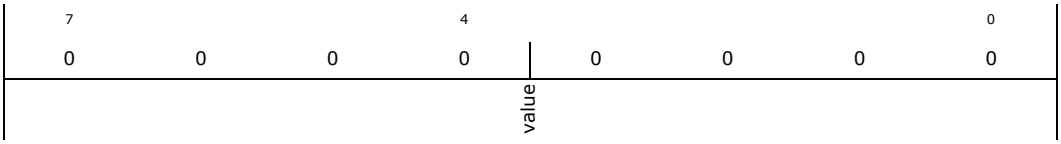
16.5.1.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:2] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

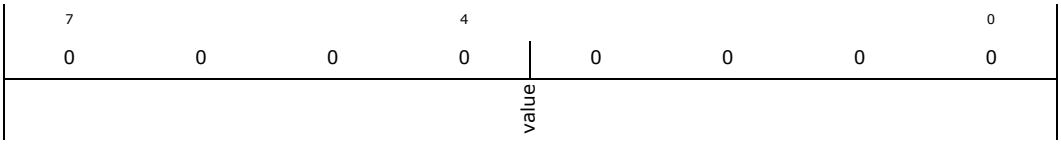
16.5.1.25 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:2] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

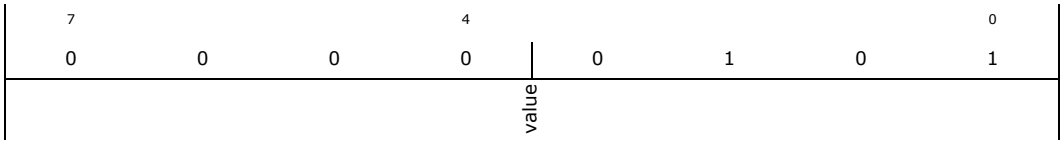
16.5.1.26 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:2] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

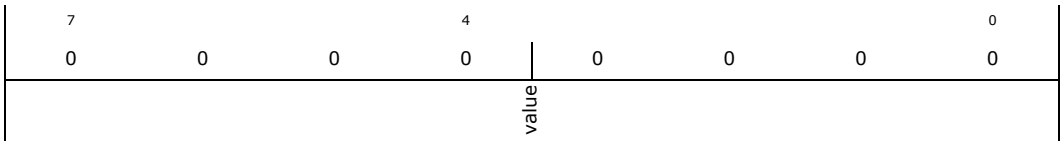
16.5.1.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:2] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

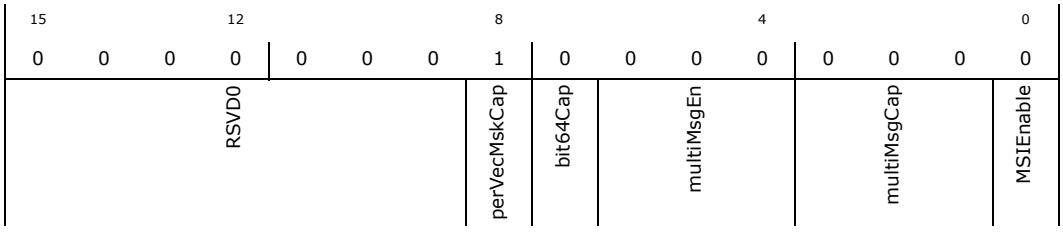
16.5.1.28 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:2] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

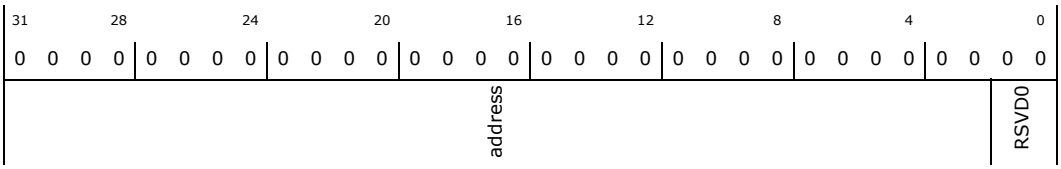
16.5.1.29 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:2] + A4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

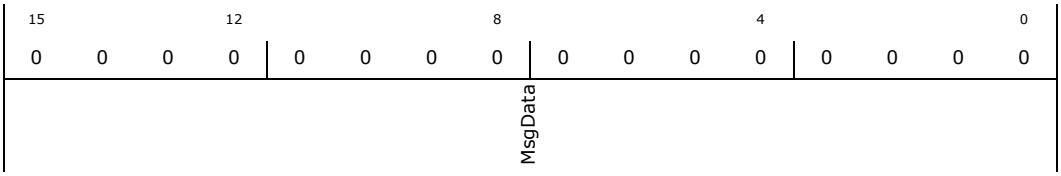
16.5.1.30 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:2] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

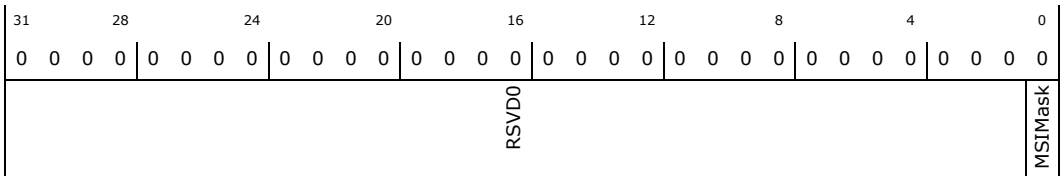
16.5.1.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:2] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

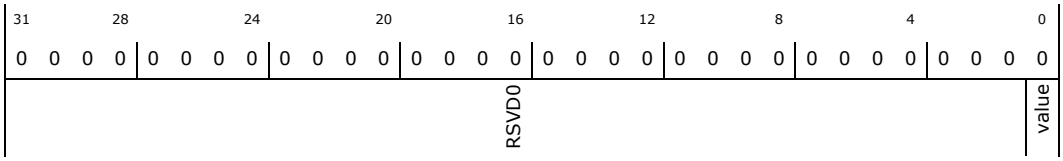
16.5.1.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

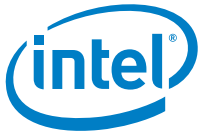
Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:2] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.



16.5.2 USB EHCI

Table 87. Summary of PCI Configuration Registers—0/20/3

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 424	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 425	0939h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 425	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 426	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 426	0C032010h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 427	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 427	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 428	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 443	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 444	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 445	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 445	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 445	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 446	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 446	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 446	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 447	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 447	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 448	00h
60h	60h	"Serial Bus Release Number Register (SBRN)—Offset 60h" on page 448	20h
61h	61h	"Frame Length Adjustment Register (FLADJ)—Offset 61h" on page 448	20h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 449	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 449	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 449	F803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 450	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 451	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 451	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 451	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 452	C0h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 452	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 453	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 453	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 454	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 454	00000000h
C0h	C3h	"USB Legacy Support Extended Capability (USBLEGSUP)—Offset C0h" on page 454	00000001h
C4h	C7h	"USB Legacy Support Control/Status (USBLEGCTLSTS)—Offset C4h" on page 455	00000000h

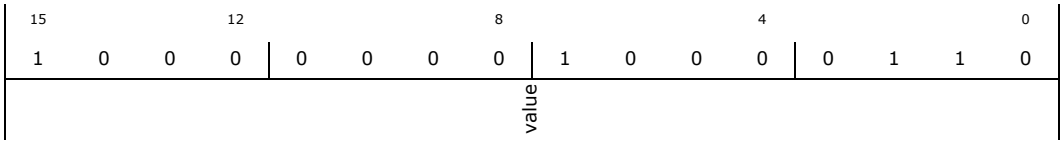
16.5.2.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:3] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

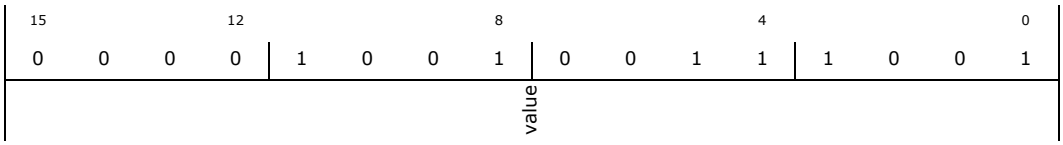
16.5.2.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:3] + 2h

Default: 0939h



Bit Range	Default & Access	Description
15: 0	0939h RO	**Device ID (value):** PCI Device ID

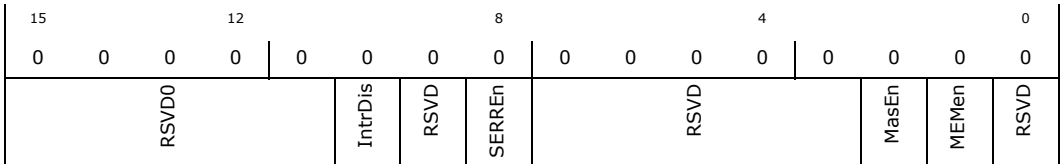
16.5.2.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:3] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved

Bit Range	Default & Access	Description
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

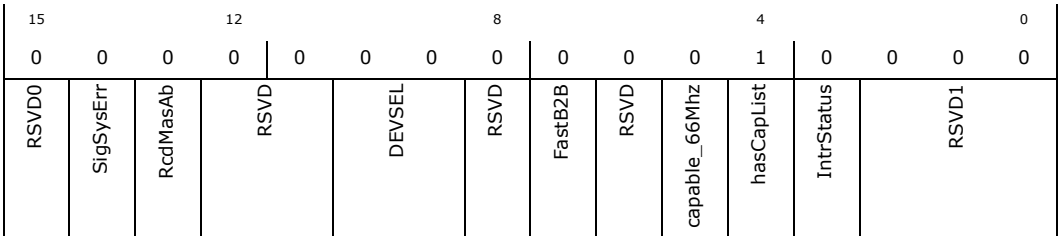
16.5.2.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:3] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

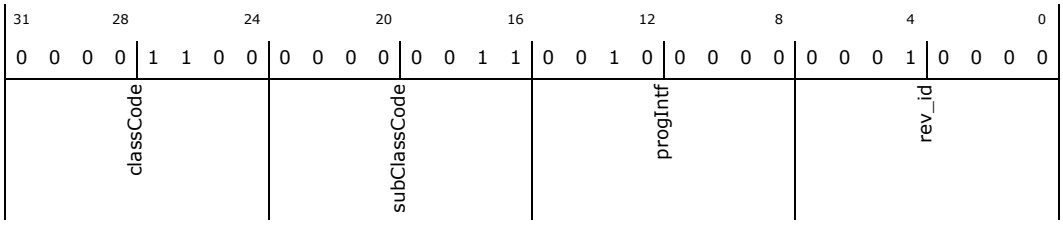
16.5.2.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:3] + 8h

Default: 0C032010h



Bit Range	Default & Access	Description
31: 24	0Ch RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	03h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	20h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

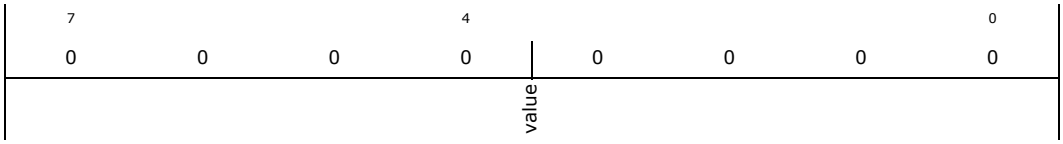
16.5.2.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:3] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

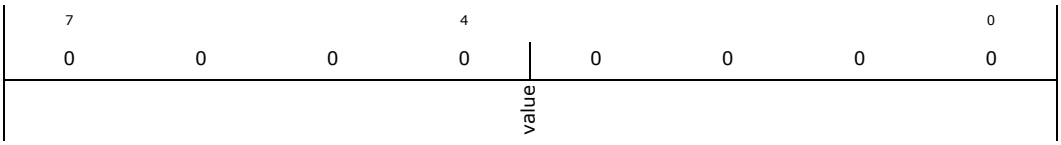
16.5.2.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:3] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

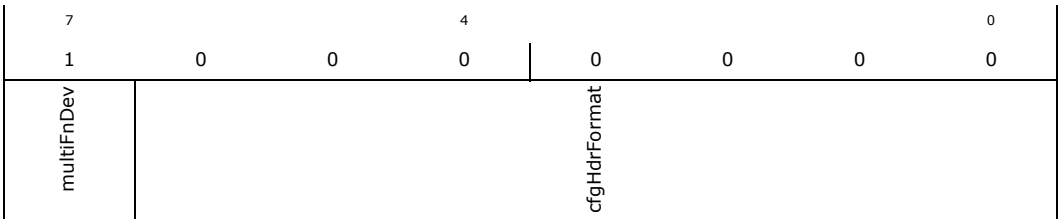
16.5.2.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:3] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

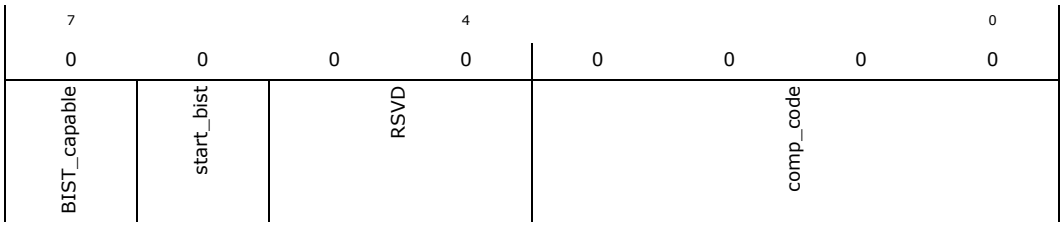
16.5.2.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:3] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

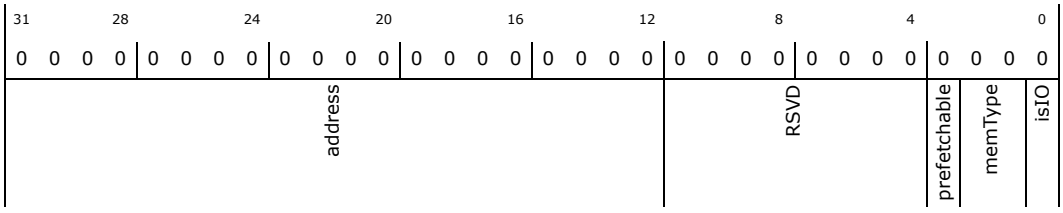
16.5.2.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

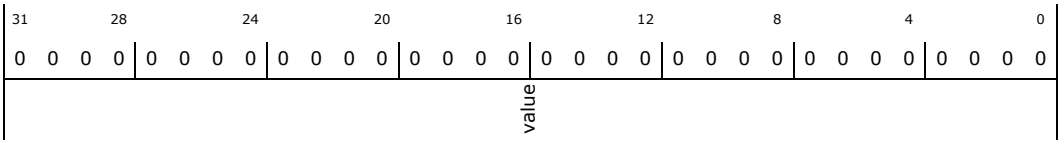
16.5.2.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:3] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

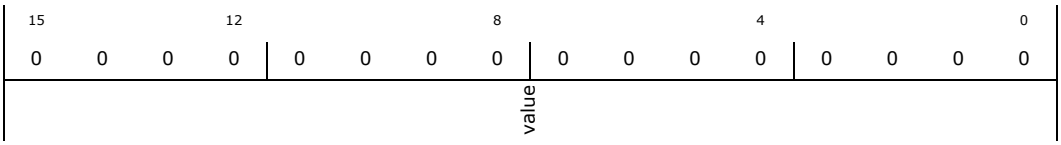
16.5.2.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:3] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

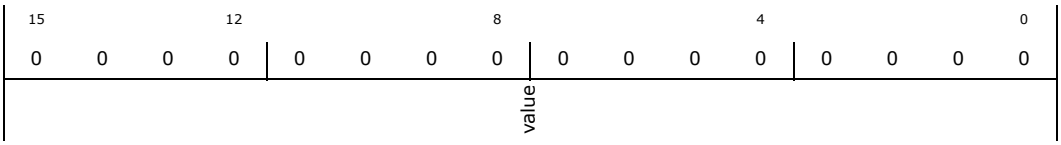
16.5.2.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:3] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

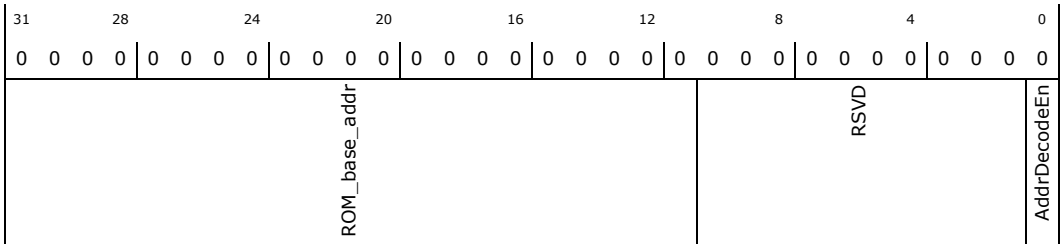
16.5.2.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:3] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

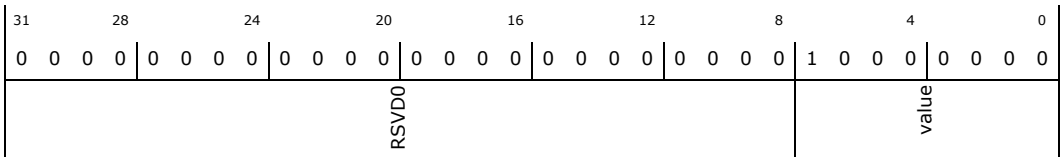
16.5.2.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:3] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

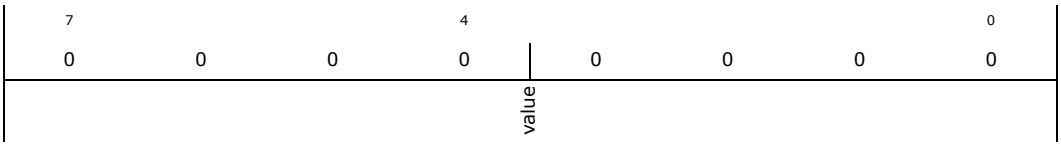
16.5.2.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:20, F:3] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

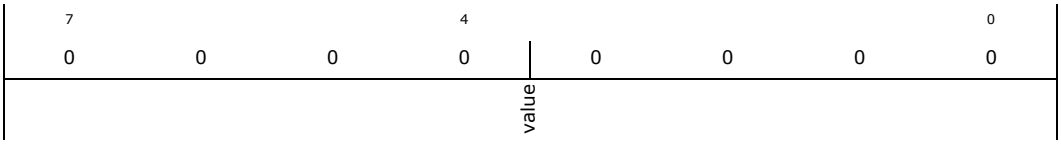
16.5.2.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:20, F:3] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

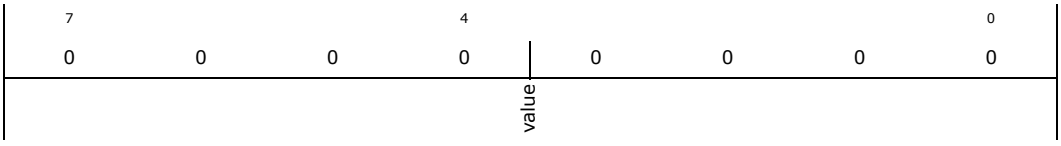
16.5.2.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:3] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

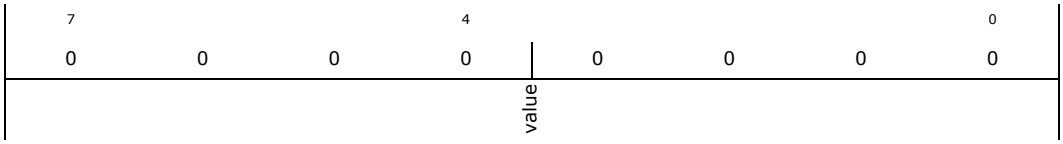
16.5.2.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:3] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

16.5.2.20 Serial Bus Release Number Register (SBRN)—Offset 60h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

SBRN: [B:0, D:20, F:3] + 60h

Default: 20h

Bit Range	Default & Access	Description
7: 0	20h RO	**Serial Bus Specification Release Number (SBRN):** Serial Bus Specification Release Number.

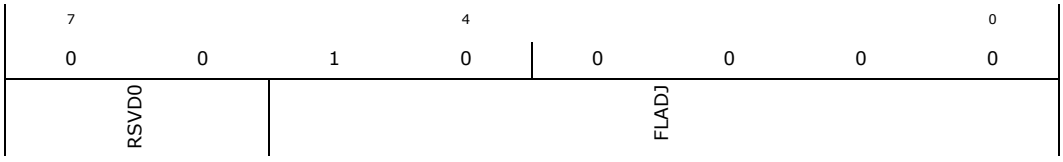
16.5.2.21 Frame Length Adjustment Register (FLADJ)—Offset 61h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

FLADJ: [B:0, D:20, F:3] + 61h

Default: 20h



Bit Range	Default & Access	Description
7: 6	0h RO	**RSVD0 (RSVD0):** Reserved

Bit Range	Default & Access	Description
5: 0	20h RW	**Frame Length Timing Value (FLADJ):** Each decimal value change to this register corresponds to 16 highspeed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives a SOF cycle time of 60000.

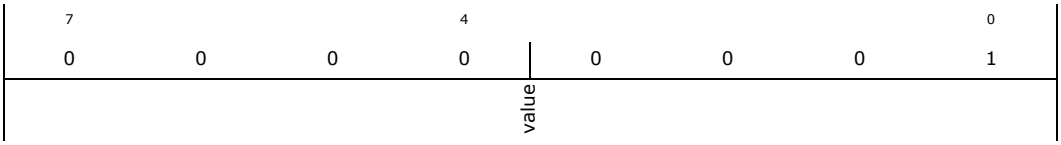
16.5.2.22 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:3] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

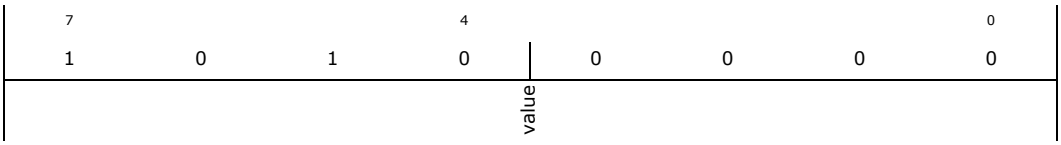
16.5.2.23 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:3] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

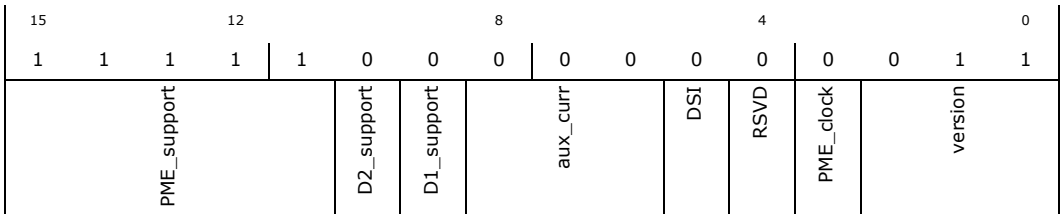
16.5.2.24 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:3] + 82h

Default: F803h



Bit Range	Default & Access	Description
15: 11	1Fh RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

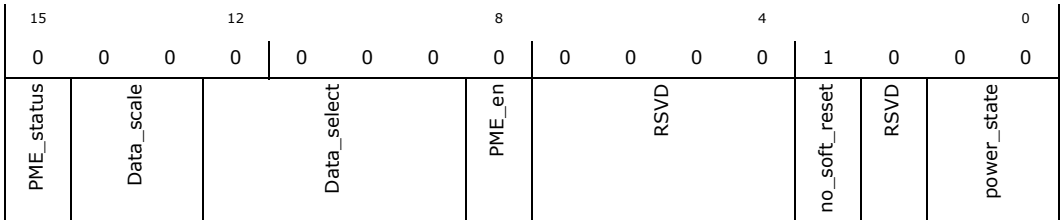
16.5.2.25 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:3] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported

Bit Range	Default & Access	Description
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

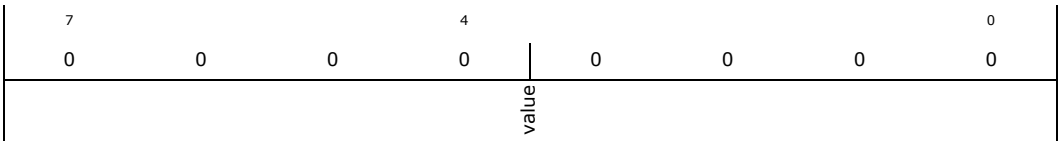
16.5.2.26 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:3] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

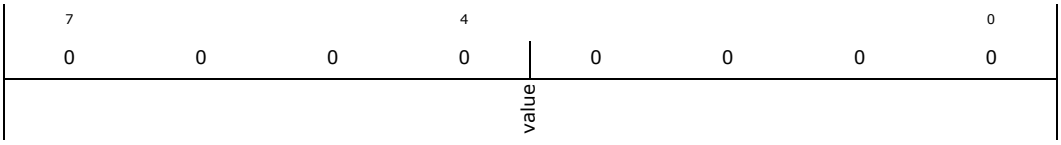
16.5.2.27 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:3] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

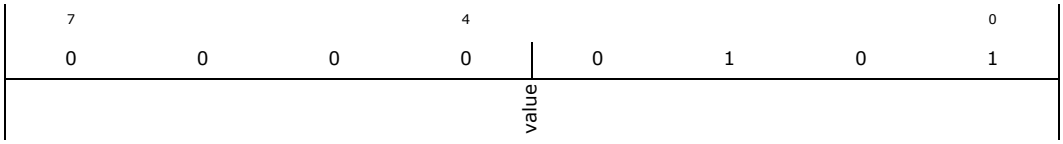
16.5.2.28 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:3] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

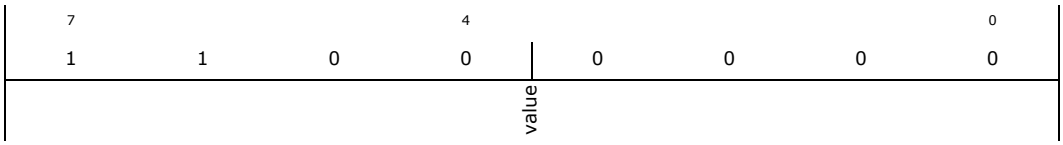
16.5.2.29 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:3] + A1h

Default: C0h



Bit Range	Default & Access	Description
7: 0	C0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xC0 to point to the USB Legacy Support Extended Capability Structure

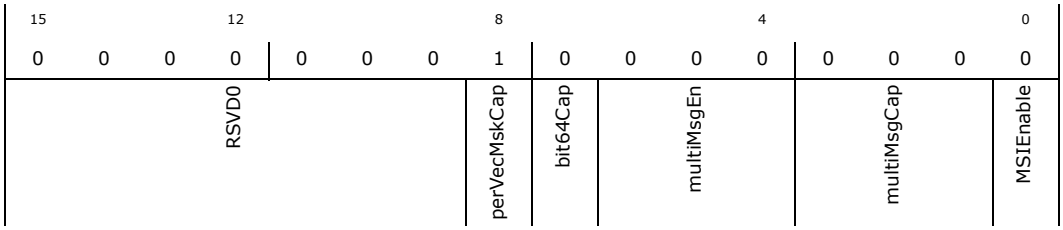
16.5.2.30 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:3] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved

Bit Range	Default & Access	Description
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

16.5.2.31 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:3] + A4h

Default: 00000000h

Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

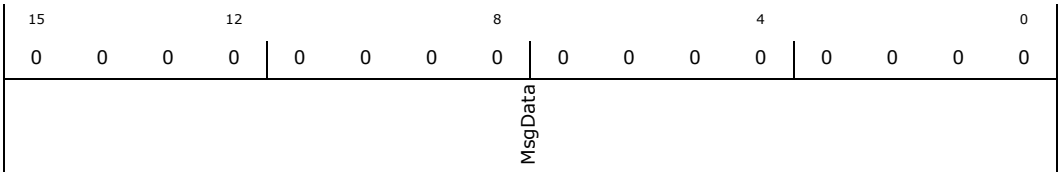
16.5.2.32 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:3] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

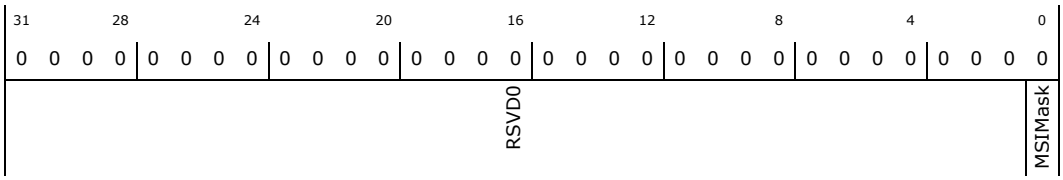
16.5.2.33 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:3] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

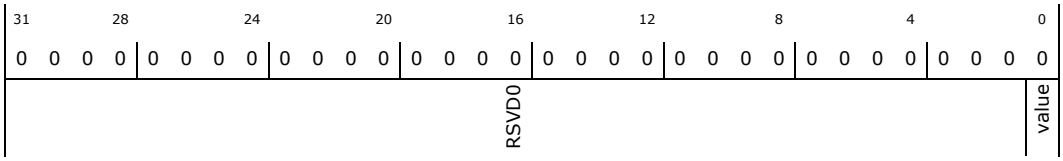
16.5.2.34 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:3] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

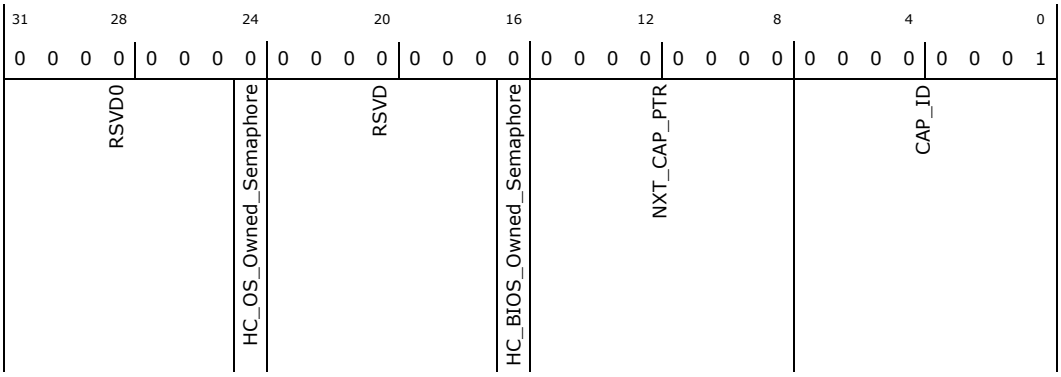
16.5.2.35 USB Legacy Support Extended Capability (USBLEGSUP)—Offset C0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

USBLEGSUP: [B:0, D:20, F:3] + C0h

Default: 00000001h



Bit Range	Default & Access	Description
31: 25	0h RO	**RSVD0 (RSVD0):** Reserved
24	0b RW	**HC OS Owned Semaphore (HC_OS_Owned_Semaphore):** System software sets this bit to request ownership of the EHCI controller. Ownership is obtained when this bit reads as one and the HC BIOS Owned Semaphore bit reads as zero.
23: 17	00h RO	**Reserved (RSVD):** Reserved.
16	0b RW	**HC BIOS Owned Semaphore (HC_BIOS_Owned_Semaphore):** The BIOS sets this bit to establish ownership of the EHCI controller. System BIOS will set this bit to a zero in response to a request for ownership of the EHCI controller by system software.
15: 8	0h RO	**Next EHCI Extended Capability Pointer (NXT_CAP_PTR):** This field points to the PCI configuration space offset of the next extended capability pointer. A value of 00h indicates the end of the extended capability list.
7: 0	01h RO	**Capability ID (CAP_ID):** This field identifies the extended capability. A value of 01h identifies the capability as Legacy Support. This extended capability requires one additional 32-bit register for control/status information, and this register is located at offset EECP+04h.

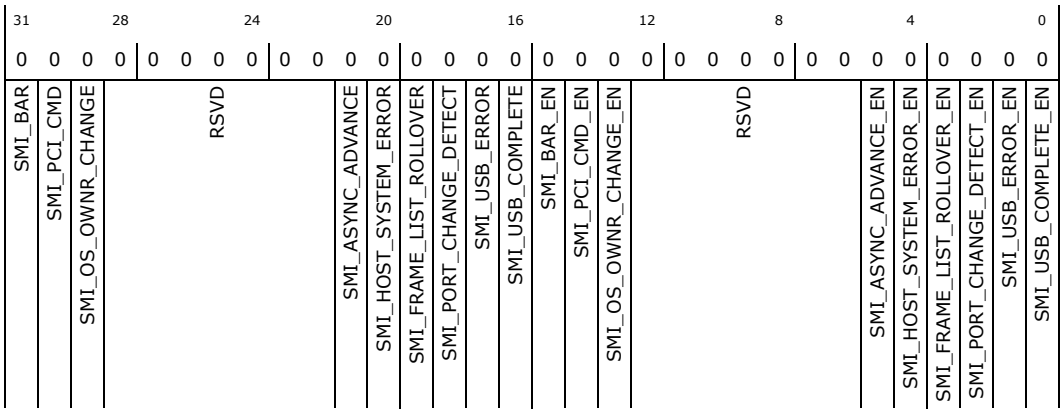
16.5.2.36 USB Legacy Support Control/Status (USBLEGCTLSTS)—Offset C4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

USBLEGCTLSTS: [B:0, D:20, F:3] + C4h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**SMI on BAR (SMI_BAR):** This bit is set to one whenever the Base Address Register (BAR) is written.
30	0b RW	**SMI on PCI Command (SMI_PCI_CMD):** This bit is set to one whenever the PCI Command Register is written.
29	0b RW	**SMI on OS Ownership Change (SMI_OS_OWNR_CHANGE):** This bit is set to one whenever the HC OS Owned Semaphore bit in the USBLEGSUP register transitions from 1 to a 0 or 0 to a 1
28: 22	00h RO	**Reserved (RSVD):** Reserved.
21	0b RO	**SMI on Async Advance (SMI_ASYNC_ADVANCE):** Shadow bit of the Interrupt on Async Advance bit in the USBSTS register see Section 2.3.2 for definition. To set this bit to a zero, system software must write a one to the Interrupt on Async Advance bit in the USBSTS register.
20	0b RO	**SMI on Host System Error (SMI_HOST_SYSTEM_ERROR):** Shadow bit of Host System Error bit in the USBSTS register, see Section 2.3.2 for definition and effects of the events associated with this bit being set to a one. To set this bit to a zero, system software must write a one to the Host System Error bit in the USBSTS register.
19	0b RO	**SMI on Frame List Rollover (SMI_FRAME_LIST_ROLLOVER):** Shadow bit of Frame List Rollover bit in the USBSTS register see Section 2.3.2 for definition. To set this bit to a zero, system software must write a one to the Frame List Rollover bit in the USBSTS register.
18	0b RO	**SMI on Port Change Detect (SMI_PORT_CHANGE_DETECT):** Shadow bit of Port Change Detect bit in the USBSTS register see Section 2.3.2 for definition. To set this bit to a zero, system software must write a one to the Port Change Detect bit in the USBSTS register.
17	0b RO	**SMI on USB Error (SMI_USB_ERROR):** Shadow bit of USB Error Interrupt (USBERRINT) bit in the USBSTS register see Section 2.3.2 for definition. To set this bit to a zero, system software must write a one to the USB Error Interrupt bit in the USBSTS register.
16	0b RO	**SMI on USB Complete (SMI_USB_COMPLETE):** Shadow bit of USB Interrupt (USBINT) bit in the USBSTS register see Section 2.3.2 for definition. To set this bit to a zero, system software must write a one to the USB Interrupt bit in the USBSTS register.
15	0b RW	**SMI on BAR Enable (SMI_BAR_EN):** When this bit is one and SMI on BAR is one, then the host controller will issue an SMI.
14	0b RW	**SMI on PCI Command Enable (SMI_PCI_CMD_EN):** When this bit is one and SMI on PCI Command is one, then the host controller will issue an SMI.
13	0b RW	**SMI on OS Ownership Enable (SMI_OS_OWNR_CHANGE_EN):** When this bit is a one AND the OS Ownership Change bit is one, the host controller will issue an SMI.

Bit Range	Default & Access	Description
12: 6	00h RO	**Reserved (RSVD):** Reserved.
5	0b RW	**SMI on Async Advance Enable (SMI_ASYNC_ADVANCE_EN):** When this bit is a one, and the SMI on Async Advance bit (above) in this register is a one, the host controller will issue an SMI immediately
4	0b RW	**SMI on Host System Error Enable (SMI_HOST_SYSTEM_ERROR_EN):** When this bit is a one, and the SMI on Host System Error bit (above) in this register is a one, the host controller will issue an SMI immediately
3	0b RW	**SMI on Frame List Rollover Enable (SMI_FRAME_LIST_ROLLOVER_EN):** When this bit is a one, and the SMI on Frame List Rollover bit (above) in this register is a one, the host controller will issue an SMI immediately.
2	0b RW	**SMI on Port Change Enable (SMI_PORT_CHANGE_DETECT_EN):** When this bit is a one, and the SMI on Port Change Detect bit (above) in this register is a one, the host controller will issue an SMI immediately.
1	0b RW	**SMI on USB Error Enable (SMI_USB_ERROR_EN):** When this bit is a one, and the SMI on USB Error bit (above) in this register is a one, the host controller will issue an SMI immediately.
0	0b RW	**USB SMI Enable (SMI_USB_COMPLETE_EN):** When this bit is a one, and the SMI on USB Complete bit (above) in this register is a one, the host controller will issue an SMI immediately.

16.5.3 USB OHCI

Table 88. Summary of PCI Configuration Registers—0/20/4

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 424	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 425	093Ah
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 425	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 426	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 426	0C031010h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 427	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 427	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 428	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 428	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 429	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 429	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 430	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 430	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 430	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 446	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 446	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 447	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 447	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 448	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 449	01h

Table 88. Summary of PCI Configuration Registers—0/20/4 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 449	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 449	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 450	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 451	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 451	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 451	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 452	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 452	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 453	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 453	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 471	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 454	00000000h

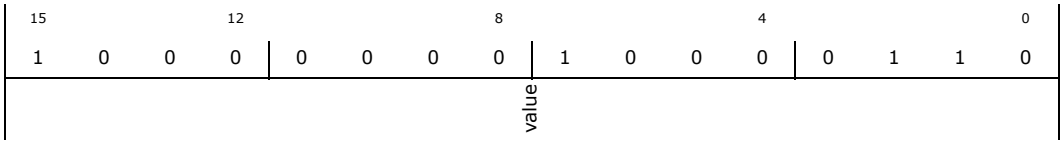
16.5.3.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:4] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

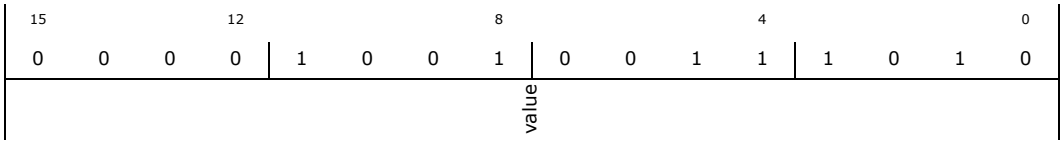
16.5.3.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:4] + 2h

Default: 093Ah



Bit Range	Default & Access	Description
15: 0	093Ah RO	**Device ID (value):** PCI Device ID

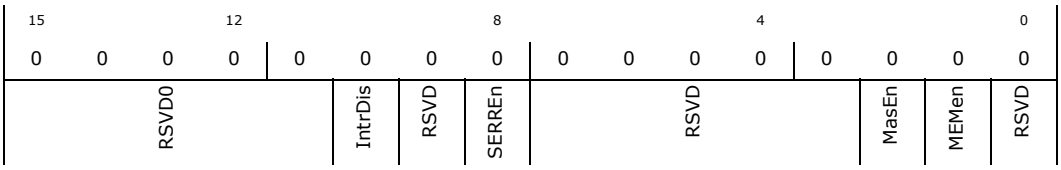
16.5.3.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:4] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

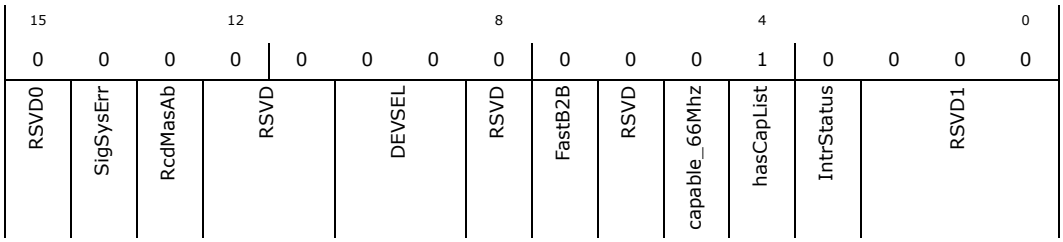
16.5.3.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:4] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

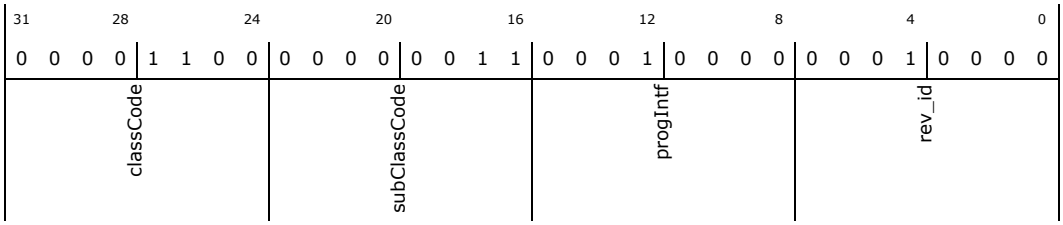
16.5.3.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:4] + 8h

Default: 0C031010h



Bit Range	Default & Access	Description
31: 24	0Ch RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	03h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	10h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

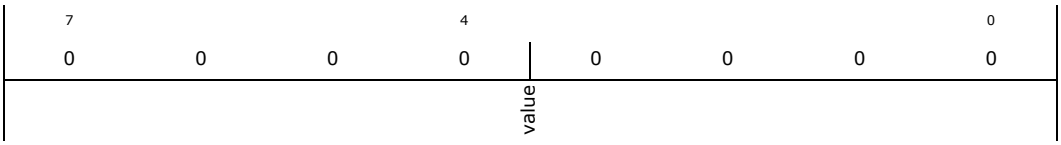
16.5.3.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:4] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

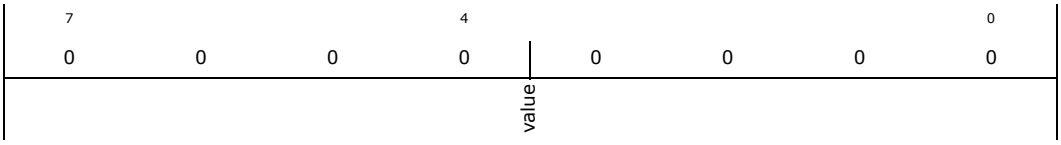
16.5.3.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:4] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

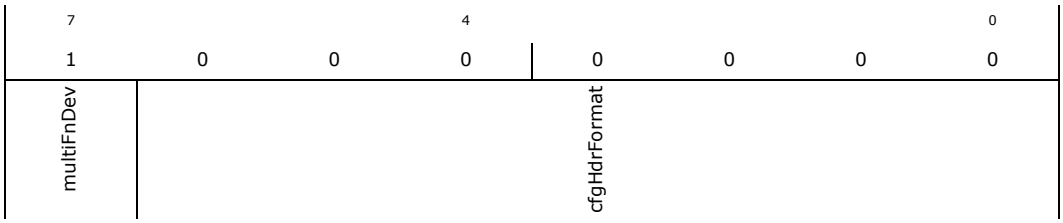
16.5.3.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:4] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

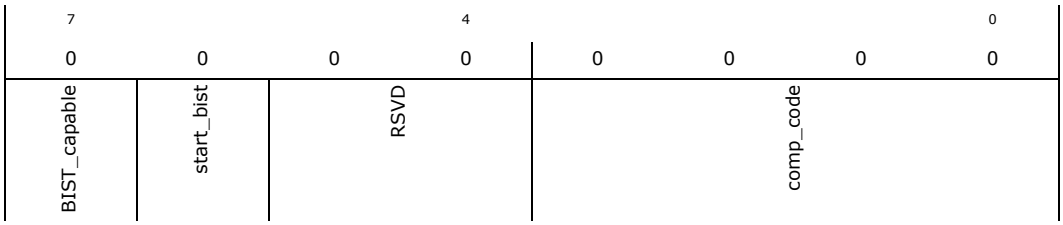
16.5.3.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:4] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

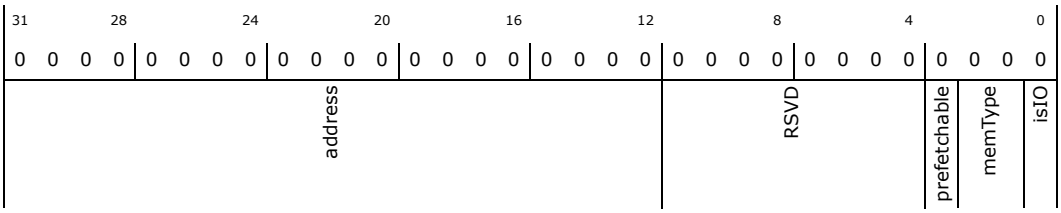
16.5.3.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

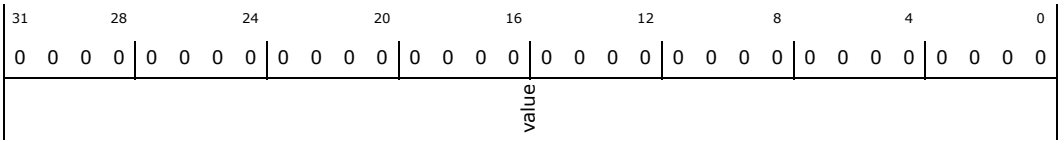
16.5.3.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:4] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

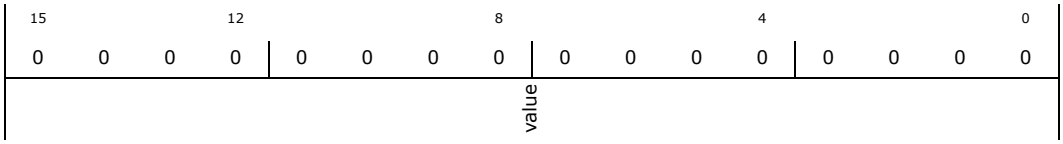
16.5.3.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:4] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

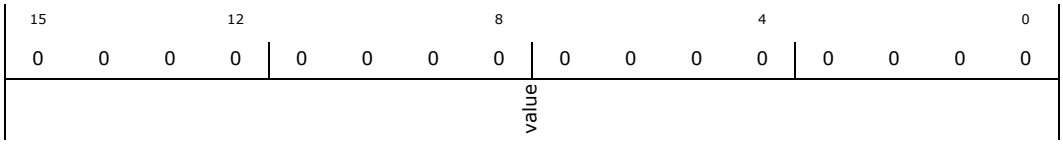
16.5.3.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:4] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

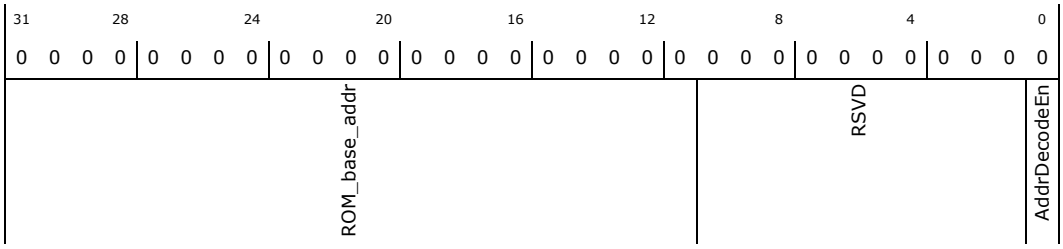
16.5.3.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:4] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

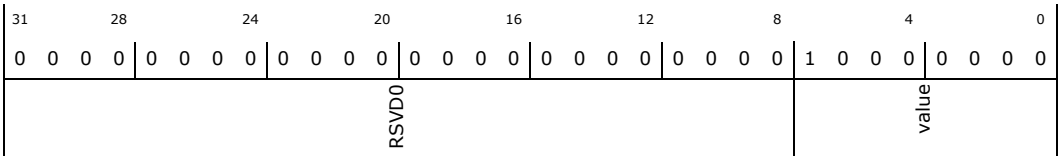
16.5.3.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:4] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

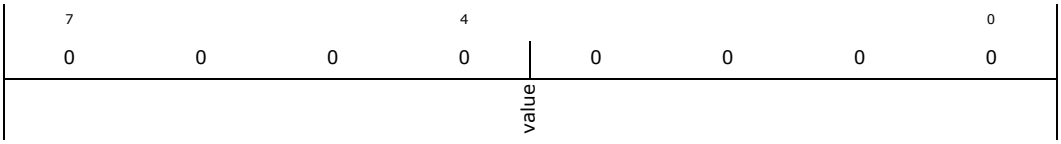
16.5.3.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:20, F:4] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

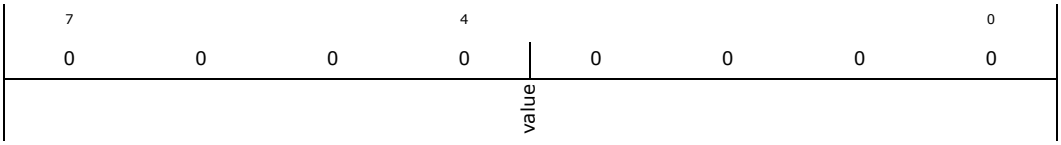
16.5.3.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:20, F:4] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

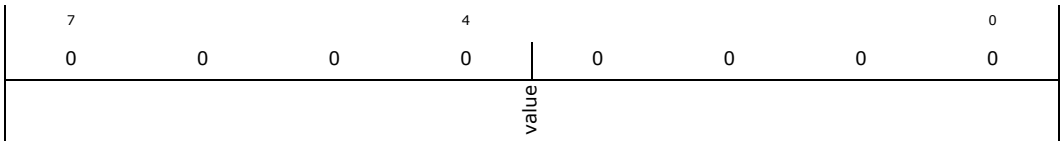
16.5.3.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:4] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

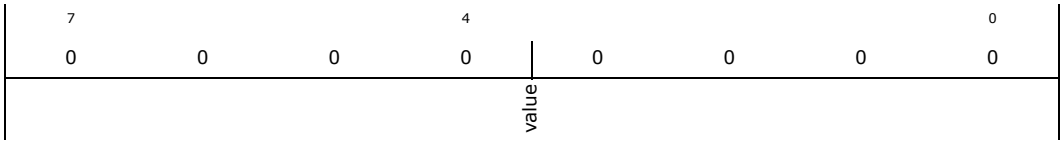
16.5.3.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:4] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

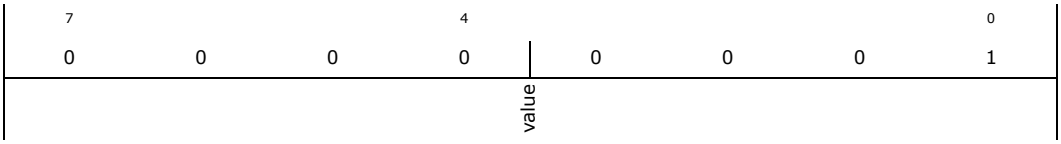
16.5.3.20 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:4] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

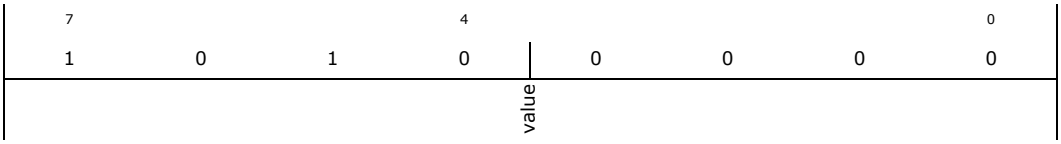
16.5.3.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:4] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

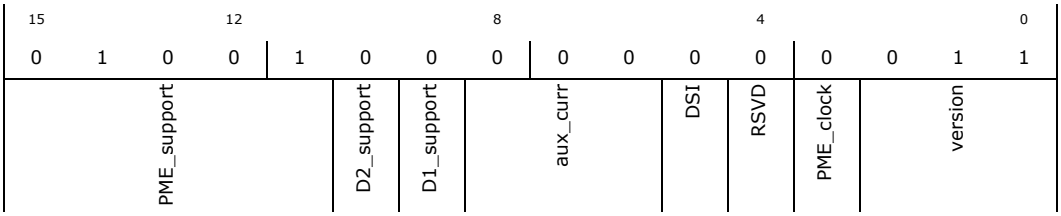
16.5.3.22 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:4] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported

Bit Range	Default & Access	Description
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

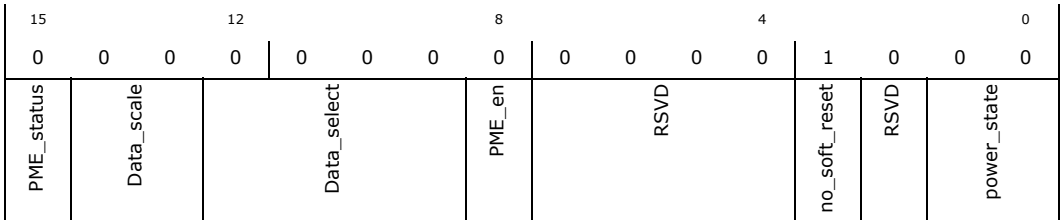
16.5.3.23 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:4] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

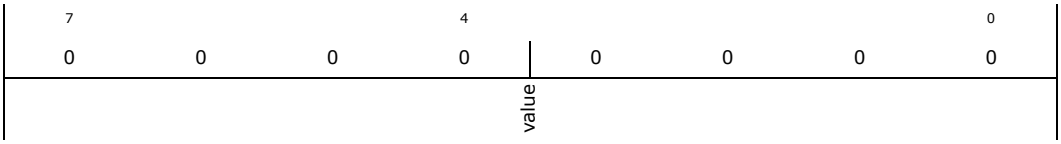
16.5.3.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:4] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

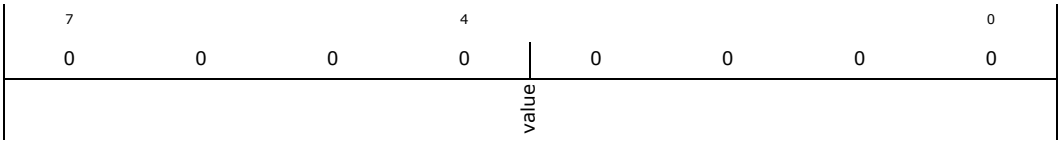
16.5.3.25 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:4] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

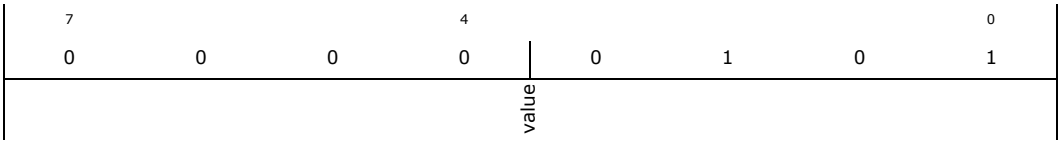
16.5.3.26 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:4] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

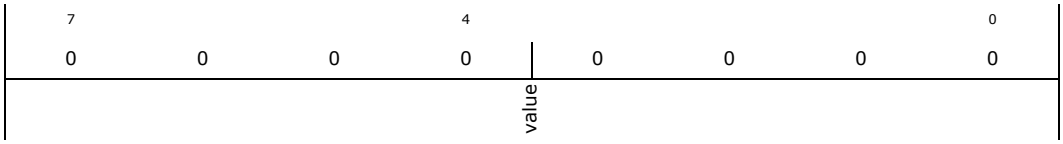
16.5.3.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:4] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

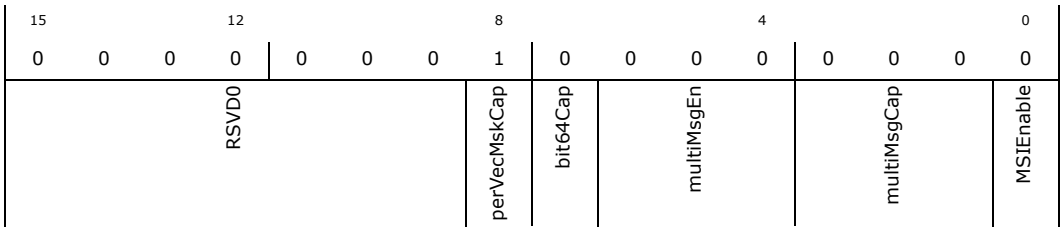
16.5.3.28 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:4] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

16.5.3.29 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:4] + A4h

Default: 00000000h

Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

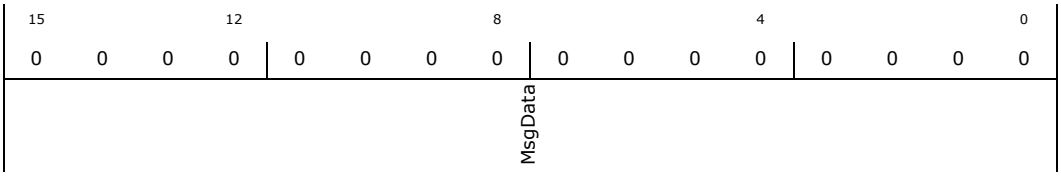
16.5.3.30 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register
(Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:4] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

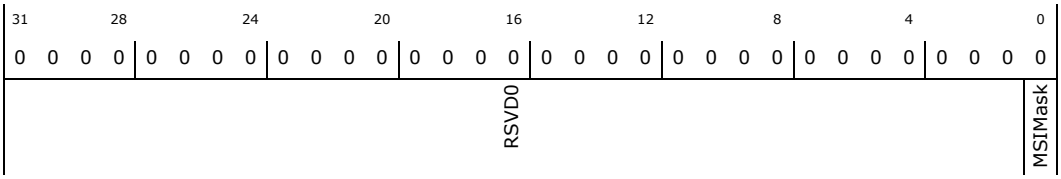
16.5.3.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:4] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

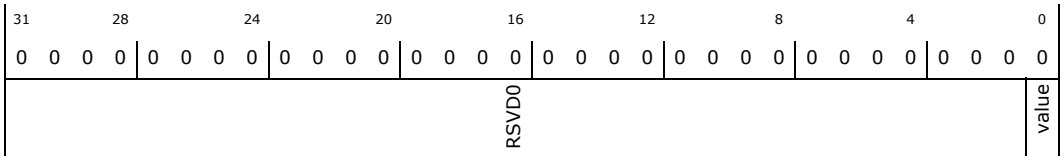
16.5.3.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:4] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

16.6 Memory Mapped Registers

16.6.1 USB Device

Table 89. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"IN Endpoint 0 Control Register (ep0_in_ctrl_udc_reg)—Offset 0h" on page 475	00000000h
4h	7h	"IN Endpoint 0 Status Register (ep0_in_sts_udc_reg)—Offset 4h" on page 476	00000000h
8h	Bh	"IN Endpoint 0 Buffer Size Register (ep0_in_bufsize_udc_reg)—Offset 8h" on page 478	00000000h
Ch	Fh	"IN Endpoint 0 Maximum Packet Size Register (ep0_in_mpkt_sz_reg)—Offset Ch" on page 479	00000000h
14h	17h	"IN Endpoint 0 Data Descriptor Pointer Register (ep0_in_desptr_udc_reg)—Offset 14h" on page 479	00000000h
1Ch	1Fh	"IN Endpoint 0 Write Confirmation register (for Slave-Only mode) (ep0_wr_cfrm_udc_reg)—Offset 1Ch" on page 480	00000000h
20h	23h	"IN Endpoint 1 Control Register (ep1_in_ctrl_udc_reg)—Offset 20h" on page 480	00000000h
24h	27h	"IN Endpoint 1 Status Register (ep1_in_sts_udc_reg)—Offset 24h" on page 481	00000000h
28h	2Bh	"IN Endpoint 1 Buffer Size Register (ep1_in_bufsize_udc_reg)—Offset 28h" on page 483	00000000h

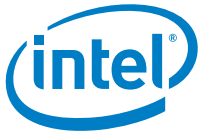


Table 89. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
2Ch	2Fh	"IN Endpoint 1 Maximum Packet Size Register (ep1_in_mpkt_sz_reg)—Offset 2Ch" on page 484	00000000h
34h	37h	"IN Endpoint 1 Data Descriptor Pointer Register (ep1_in_desptr_udc_reg)—Offset 34h" on page 484	00000000h
3Ch	3Fh	"IN Endpoint 1 Write Confirmation register (for Slave-Only mode) (ep1_wr_cfrm_udc_reg)—Offset 3Ch" on page 485	00000000h
40h	43h	"IN Endpoint 2 Control Register (ep2_in_ctrl_udc_reg)—Offset 40h" on page 485	00000000h
44h	47h	"IN Endpoint 2 Status Register (ep2_in_sts_udc_reg)—Offset 44h" on page 486	00000000h
48h	4Bh	"IN Endpoint 2 Buffer Size Register (ep2_in_bufsize_udc_reg)—Offset 48h" on page 488	00000000h
4Ch	4Fh	"IN Endpoint 2 Maximum Packet Size Register (ep2_in_mpkt_sz_reg)—Offset 4Ch" on page 489	00000000h
54h	57h	"IN Endpoint 2 Data Descriptor Pointer Register (ep2_in_desptr_udc_reg)—Offset 54h" on page 489	00000000h
5Ch	5Fh	"IN Endpoint 2 Write Confirmation register (for Slave-Only mode) (ep2_wr_cfrm_udc_reg)—Offset 5Ch" on page 490	00000000h
60h	63h	"IN Endpoint 3 Control Register (ep3_in_ctrl_udc_reg)—Offset 60h" on page 490	00000000h
64h	67h	"IN Endpoint 3 Status Register (ep3_in_sts_udc_reg)—Offset 64h" on page 491	00000000h
68h	6Bh	"IN Endpoint 3 Buffer Size Register (ep3_in_bufsize_udc_reg)—Offset 68h" on page 493	00000000h
6Ch	6Fh	"IN Endpoint 3 Maximum Packet Size Register (ep3_in_mpkt_sz_reg)—Offset 6Ch" on page 494	00000000h
74h	77h	"IN Endpoint 3 Data Descriptor Pointer Register (ep3_in_desptr_udc_reg)—Offset 74h" on page 494	00000000h
7Ch	7Fh	"IN Endpoint 3 Write Confirmation register (for Slave-Only mode) (ep3_wr_cfrm_udc_reg)—Offset 7Ch" on page 495	00000000h
200h	203h	"OUT Endpoint 0 Control Register (ep0_out_ctrl_udc_reg)—Offset 200h" on page 495	00000000h
204h	207h	"OUT Endpoint 0 Status Register (ep0_out_sts_udc_reg)—Offset 204h" on page 496	00000100h
208h	20Bh	"OUT Endpoint 0 Receive Packet Frame Number Register (ep0_out_rpf_udc_reg)—Offset 208h" on page 498	00000000h
20Ch	20Fh	"OUT Endpoint 0 Buffer Size Register (ep0_out_bufsize_udc_reg)—Offset 20Ch" on page 499	00000000h
210h	213h	"OUT Endpoint 0 SETUP Buffer Pointer Register (ep0_subptr_udc_reg)—Offset 210h" on page 499	00000000h
214h	217h	"OUT Endpoint 0 Data Descriptor Pointer Register (ep0_out_desptr_udc_reg)—Offset 214h" on page 500	00000000h
21Ch	21Fh	"OUT Endpoint 0 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep0_rd_cfrm_udc_reg)—Offset 21Ch" on page 500	00000000h
220h	223h	"OUT Endpoint 1 Control Register (ep1_out_ctrl_udc_reg)—Offset 220h" on page 501	00000000h
224h	227h	"OUT Endpoint 1 Status Register (ep1_out_sts_udc_reg)—Offset 224h" on page 502	00000100h
228h	22Bh	"OUT Endpoint 1 Receive Packet Frame Number Register (ep1_out_rpf_udc_reg)—Offset 228h" on page 504	00000000h
22Ch	22Fh	"OUT Endpoint 1 Buffer Size Register (ep1_out_bufsize_udc_reg)—Offset 22Ch" on page 505	00000000h
230h	233h	"OUT Endpoint 1 SETUP Buffer Pointer Register (ep1_subptr_udc_reg)—Offset 230h" on page 505	00000000h
234h	237h	"OUT Endpoint 1 Data Descriptor Pointer Register (ep1_out_desptr_udc_reg)—Offset 234h" on page 506	00000000h

Table 89. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
23Ch	23Fh	"OUT Endpoint 1 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep1_rd_cfrm_udc_reg)—Offset 23Ch" on page 506	00000000h
240h	243h	"OUT Endpoint 2 Control Register (ep2_out_ctrl_udc_reg)—Offset 240h" on page 507	00000000h
244h	247h	"OUT Endpoint 2 Status Register (ep2_out_sts_udc_reg)—Offset 244h" on page 508	00000100h
248h	24Bh	"OUT Endpoint 2 Receive Packet Frame Number Register (ep2_out_rpf_udc_reg)—Offset 248h" on page 510	00000000h
24Ch	24Fh	"OUT Endpoint 2 Buffer Size Register (ep2_out_bufsize_udc_reg)—Offset 24Ch" on page 511	00000000h
250h	253h	"OUT Endpoint 2 SETUP Buffer Pointer Register (ep2_subptr_udc_reg)—Offset 250h" on page 511	00000000h
254h	257h	"OUT Endpoint 2 Data Descriptor Pointer Register (ep2_out_desptr_udc_reg)—Offset 254h" on page 512	00000000h
25Ch	25Fh	"OUT Endpoint 2 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep2_rd_cfrm_udc_reg)—Offset 25Ch" on page 512	00000000h
260h	263h	"OUT Endpoint 3 Control Register (ep3_out_ctrl_udc_reg)—Offset 260h" on page 513	00000000h
264h	267h	"OUT Endpoint 3 Status Register (ep3_out_sts_udc_reg)—Offset 264h" on page 514	00000100h
268h	26Bh	"OUT Endpoint 3 Receive Packet Frame Number Register (ep3_out_rpf_udc_reg)—Offset 268h" on page 516	00000000h
26Ch	26Fh	"OUT Endpoint 3 Buffer Size Register (ep3_out_bufsize_udc_reg)—Offset 26Ch" on page 517	00000000h
270h	273h	"OUT Endpoint 3 SETUP Buffer Pointer Register (ep3_subptr_udc_reg)—Offset 270h" on page 517	00000000h
274h	277h	"OUT Endpoint 3 Data Descriptor Pointer Register (ep3_out_desptr_udc_reg)—Offset 274h" on page 518	00000000h
27Ch	27Fh	"OUT Endpoint 3 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep3_rd_cfrm_udc_reg)—Offset 27Ch" on page 518	00000000h
400h	403h	"Device Configuration Register (d_cfg_udc_reg)—Offset 400h" on page 519	00000020h
404h	407h	"Device Control Register (d_ctrl_udc_reg)—Offset 404h" on page 520	00000400h
408h	40Bh	"Device Status Register (d_sts_udc_reg)—Offset 408h" on page 522	00000000h
40Ch	40Fh	"Device Interrupt Register (d_intr_udc_reg)—Offset 40Ch" on page 523	00000000h
410h	413h	"Device Interrupt Mask Register (d_intr_msk_udc_reg)—Offset 410h" on page 524	00000000h
414h	417h	"Endpoints Interrupt Register (ep_intr_udc_reg)—Offset 414h" on page 525	00000000h
418h	41Bh	"Endpoints Interrupt Mask Register (ep_intr_msk_udc_reg)—Offset 418h" on page 525	00000000h
41Ch	41Fh	"Test Mode Register (test_mode_udc_reg)—Offset 41Ch" on page 526	00000000h
420h	423h	"Product Release Number Register (revision_udc_reg)—Offset 420h" on page 527	3234352Ah
500h	503h	"SETUP command address pointer register (udc_desc_addr_udc_reg)—Offset 500h" on page 527	00000000h
504h	507h	"Physical Endpoint 0 Register (udc_ep_ne_udc_reg_0)—Offset 504h" on page 528	00000000h
508h	50Bh	"Physical Endpoint 1 Register (udc_ep_ne_udc_reg_1)—Offset 508h" on page 528	00000000h
50Ch	50Fh	"Physical Endpoint 2 Register (udc_ep_ne_udc_reg_2)—Offset 50Ch" on page 529	00000000h
510h	513h	"Physical Endpoint 3 Register (udc_ep_ne_udc_reg_3)—Offset 510h" on page 530	00000000h
514h	517h	"Physical Endpoint 4 Register (udc_ep_ne_udc_reg_4)—Offset 514h" on page 531	00000000h
518h	51Bh	"Physical Endpoint 5 Register (udc_ep_ne_udc_reg_5)—Offset 518h" on page 532	00000000h
51Ch	51Fh	"Physical Endpoint 6 Register (udc_ep_ne_udc_reg_6)—Offset 51Ch" on page 532	00000000h

Table 89. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
800h + [0-511]*4h	803h	"RxFIFO Array[0-511] (udc_rx_fifo_reg_array[0-511])—Offset 800h, Count 512, Stride 4h" on page 533	00000000h
1000h + [0-255]*4h	1003h	"TxFIFO 0 Array[0-255] (udc_tx_fifo_reg_0_array[0-255])—Offset 1000h, Count 256, Stride 4h" on page 534	00000000h
1400h + [0-255]*4h	1403h	"TxFIFO 1 Array[0-255] (udc_tx_fifo_reg_1_array[0-255])—Offset 1400h, Count 256, Stride 4h" on page 534	00000000h
1800h + [0-255]*4h	1803h	"TxFIFO 2 Array[0-255] (udc_tx_fifo_reg_2_array[0-255])—Offset 1800h, Count 256, Stride 4h" on page 534	00000000h
1C00h + [0-255]*4h	1C03h	"TxFIFO 3 Array[0-255] (udc_tx_fifo_reg_3_array[0-255])—Offset 1C00h, Count 256, Stride 4h" on page 535	00000000h

16.6.1.1 IN Endpoint 0 Control Register (ep0_in_ctrl_udc_reg)—Offset 0h

This register is used to program the endpoint as required by the application.

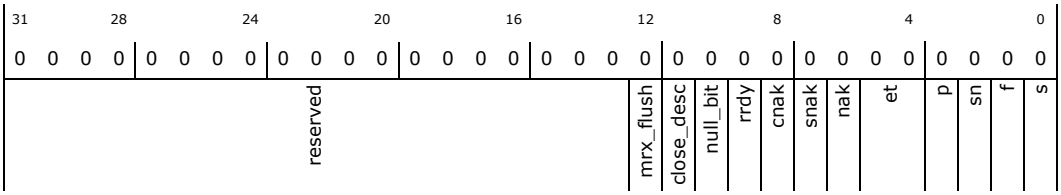
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 RO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RW	**Send Null Packet (null_bit):** This bit provides the application with a mechanism to instruct the USB Device Controller to send a NULL (zero length) packet when no data is available in the particular endpoint TxFIFO. If this bit is set, when no data is available in the endpoint TxFIFO, the USB Device Controller sends a NULL packet.
9	0x0 RO	**Receive Ready (rrdy):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Bit (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: A NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RW	**Poll Demand (p):** Poll demand from the application. The application can set this bit after an IN token is received from the endpoint. The application can also set this bit before an IN token is received for the endpoint, if it has the IN transfer data in advance. Note: After sending a zero-length or short packet, the application must wait for the next XFERDONE_TXEMPTY interrupt, before setting up the P bit for the next transfer.
2	0x0 RO	**Snoop Mode (sn):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
1	0x0 RW	**Flush TxFIFO (f):** The application firmware sets this bit to 1 after it has detected a disconnect/connect on the USB cable, then waits for the IN token endpoint interrupt before resetting this bit to 0. This flushes the stale data out of the TxFIFO. This bit is cleared by the core when TDC (Transmit DMA Complete) occurs on this endpoint.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.2 IN Endpoint 0 Status Register (ep0_in_sts_udc_reg)—Offset 4h

The Endpoint Status register indicates the endpoint status.

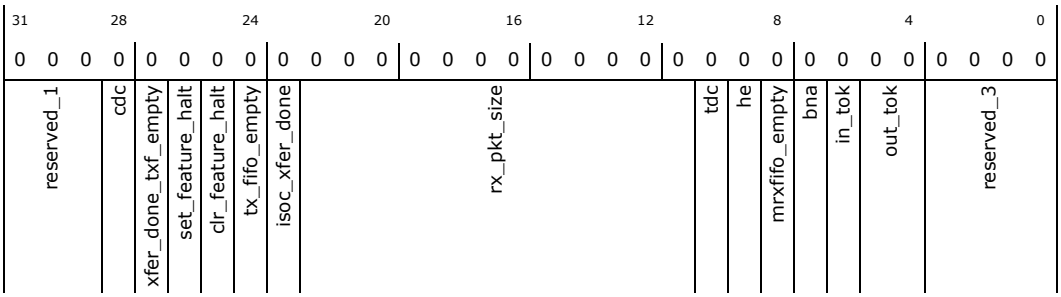
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RW/1C	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This bit indicates that the TXFIFO is empty after the DMA transfer has been completed. The application can use this bit to set the poll bit for the next transfer. The application must first clear this bit after servicing the interrupt.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint. To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature stall command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RW/1C	**Transmit FIFO Empty detected (tx_fifo_empty):** This bit indicates that the Transmit FIFO Empty condition is triggered. Application can use this information to load the subsequent data into the Transmit FIFO. The application must clear this bit after writing the data into the Transmit FIFO. Note: This bit is not used for Isochronous endpoints.
23	0x0 RW/1C	**Isochronous IN transaction complete (isoc_xfer_done):** This bit indicates that the isochronous IN transaction for this endpoint is complete. The application can use this information to program the isochronous IN data for the next microframe. This bit is used only in Slave-Only mode.
22:11	0x0 RO	**Receive Packet Size (rx_pkt_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
10	0x0 RW/1C	**Transmit DMA Completion (tdc):** Indicates the transmit DMA has completed transferring a descriptor chain data to the Tx FIFO. After servicing the interrupt, the application must clear this bit.
9	0x0 RW/1C	**Error response on the host bus (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	0x0 RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RW/1C	**IN token (in_tok):** An IN token has been received by this endpoint. After servicing the interrupt, application must clear this bit.
5:4	0x0 RO	**OUT token (out_tok):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.3 IN Endpoint 0 Buffer Size Register (ep0_in_bufsize_udc_reg)—Offset 8h

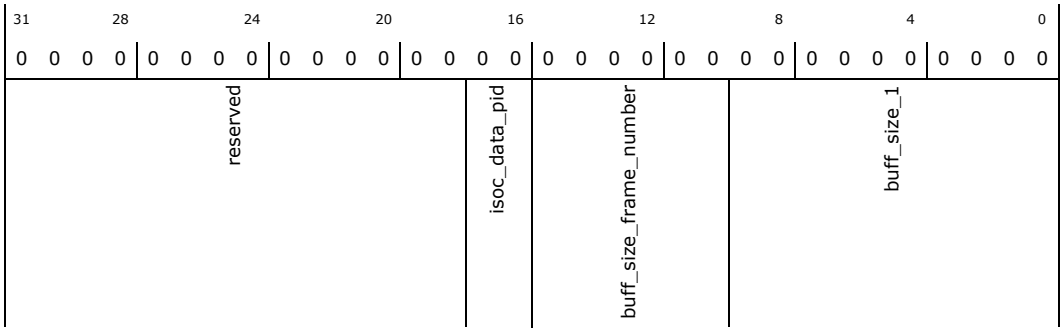
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RW	**Isochronous IN Transaction PID (isoc_data_pid):** Initial data PID to be sent for a high-bandwidth isochronous IN transaction. This field is used only in Slave-Only mode. 00: DATA0 PID is sent 01: DATA0 PID is sent 10: DATA1 PID is sent 11: DATA2 PID is sent
15:10	0x0 RO	**Buffer Size (buff_size_frame_number):** These bits are reserved and should be set to zero.
9:0	0x0 RW	**Buffer Size (buff_size_1):** Buffer size required for this endpoint. The application can program this field to make each endpoints buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Transmit FIFO. NOTE: the maximum size of the TxFIFO is 1024 32-bit word entries.

16.6.1.4 IN Endpoint 0 Maximum Packet Size Register (ep0_in_mpkt_sz_reg)—Offset Ch

This register also specifies the maximum packet size an endpoint should support. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the USB Device Controller register space.

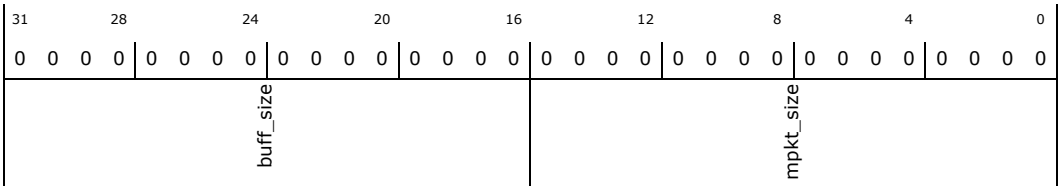
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RO	**Buffer Size (buff_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
15:0	0x0 RO	**Max Packet Size (mpkt_size):** Maximum packet size for the endpoint in bytes. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.5 IN Endpoint 0 Data Descriptor Pointer Register (ep0_in_desptr_udc_reg)—Offset 14h

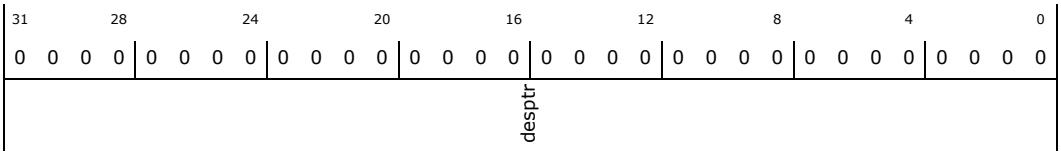
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointers.

16.6.1.6 IN Endpoint 0 Write Confirmation register (for Slave-Only mode) (ep0_wr_cfrm_udc_reg)—Offset 1Ch

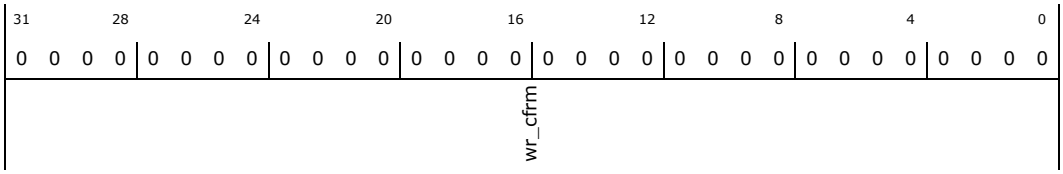
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Write Confirmation (for Slave-Only mode) (wr_cfrm):** Writing to this register confirms the IN data into the TxFIFO.

16.6.1.7 IN Endpoint 1 Control Register (ep1_in_ctrl_udc_reg)—Offset 20h

This register is used to program the endpoint as required by the application.

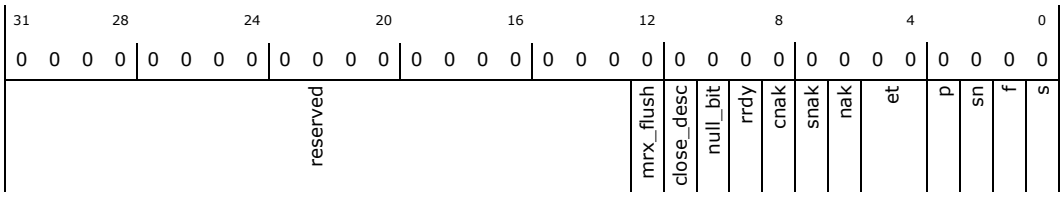
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 RO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RW	**Send Null Packet (null_bit):** This bit provides the application with a mechanism to instruct the USB Device Controller to send a NULL (zero length) packet when no data is available in the particular endpoint TxFIFO. If this bit is set, when no data is available in the endpoint TxFIFO, the USB Device Controller sends a NULL packet.
9	0x0 RO	**Receive Ready (rrdy):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Bit (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: A NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RW	**Poll Demand (p):** Poll demand from the application. The application can set this bit after an IN token is received from the endpoint. The application can also set this bit before an IN token is received for the endpoint, if it has the IN transfer data in advance. Note: After sending a zero-length or short packet, the application must wait for the next XFERDONE_TXEMPTY interrupt, before setting up the P bit for the next transfer.
2	0x0 RO	**Snoop Mode (sn):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
1	0x0 RW	**Flush TxFIFO (f):** The application firmware sets this bit to 1 after it has detected a disconnect/connect on the USB cable, then waits for the IN token endpoint interrupt before resetting this bit to 0. This flushes the stale data out of the TxFIFO. This bit is cleared by the core when TDC (Transmit DMA Complete) occurs on this endpoint.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.8 IN Endpoint 1 Status Register (ep1_in_sts_udc_reg)—Offset 24h

The Endpoint Status register indicates the endpoint status.

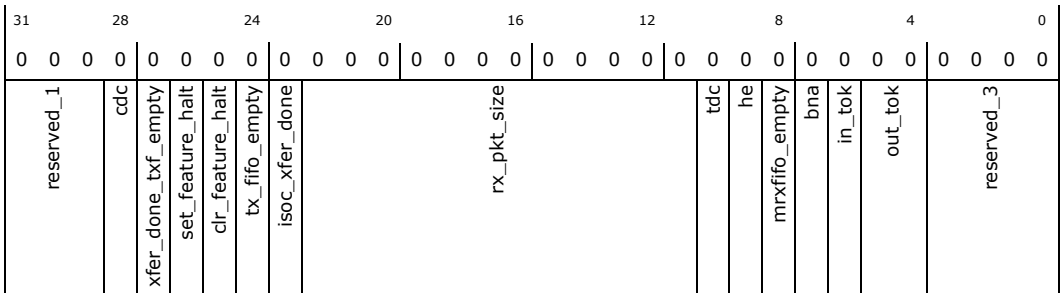
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 24h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RW/1C	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This bit indicates that the TXFIFO is empty after the DMA transfer has been completed. The application can use this bit to set the poll bit for the next transfer. The application must first clear this bit after servicing the interrupt.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint. To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature stall command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RW/1C	**Transmit FIFO Empty detected (tx_fifo_empty):** This bit indicates that the Transmit FIFO Empty condition is triggered. Application can use this information to load the subsequent data into the Transmit FIFO. The application must clear this bit after writing the data into the Transmit FIFO. Note: This bit is not used for Isochronous endpoints.
23	0x0 RW/1C	**Isochronous IN transaction complete (isoc_xfer_done):** This bit indicates that the isochronous IN transaction for this endpoint is complete. The application can use this information to program the isochronous IN data for the next microframe. This bit is used only in Slave-Only mode.
22:11	0x0 RO	**Receive Packet Size (rx_pkt_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
10	0x0 RW/1C	**Transmit DMA Completion (tdc):** Indicates the transmit DMA has completed transferring a descriptor chain data to the Tx FIFO. After servicing the interrupt, the application must clear this bit.
9	0x0 RW/1C	**Error response on the host bus (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	0x0 RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RW/1C	**IN token (in_tok):** An IN token has been received by this endpoint. After servicing the interrupt, application must clear this bit.
5:4	0x0 RO	**OUT token (out_tok):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.9 IN Endpoint 1 Buffer Size Register (ep1_in_bufsize_udc_reg)—Offset 28h

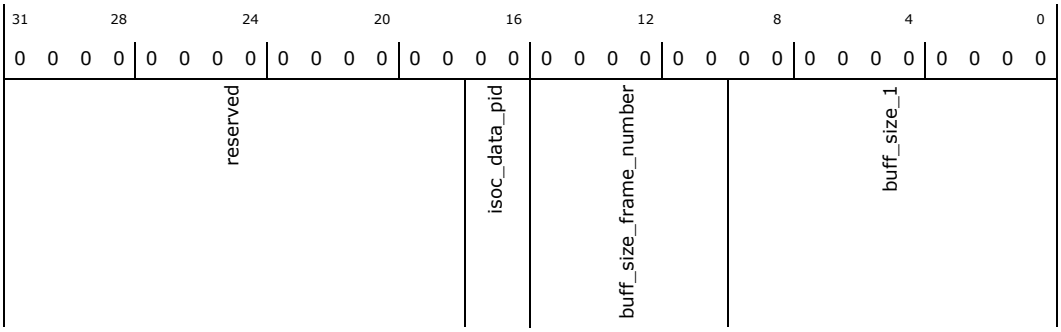
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 28h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RW	**Isochronous IN Transaction PID (isoc_data_pid):** Initial data PID to be sent for a high-bandwidth isochronous IN transaction. This field is used only in Slave-Only mode. 00: DATA0 PID is sent 01: DATA0 PID is sent 10: DATA1 PID is sent 11: DATA2 PID is sent
15:10	0x0 RO	**Buffer Size (buff_size_frame_number):** These bits are reserved and should be set to zero.
9:0	0x0 RW	**Buffer Size (buff_size_1):** Buffer size required for this endpoint. The application can program this field to make each endpoints buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Transmit FIFO. NOTE: the maximum size of the TxFIFO is 1024 32-bit word entries.

16.6.1.10 IN Endpoint 1 Maximum Packet Size Register (ep1_in_mpkt_sz_reg)—Offset 2Ch

This register also specifies the maximum packet size an endpoint should support. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the USB Device Controller register space.

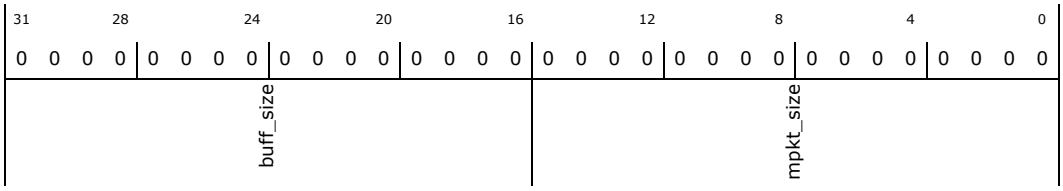
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 2Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RO	**Buffer Size (buff_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
15:0	0x0 RO	**Max Packet Size (mpkt_size):** Maximum packet size for the endpoint in bytes. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.11 IN Endpoint 1 Data Descriptor Pointer Register (ep1_in_desptr_udc_reg)—Offset 34h

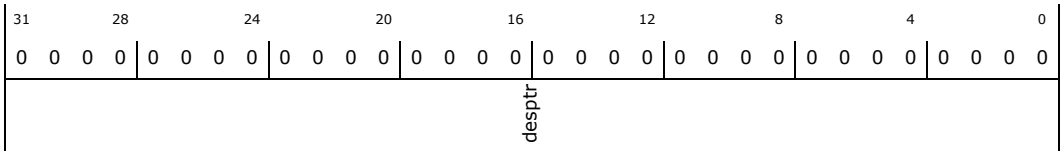
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 34h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointers.

16.6.1.12 IN Endpoint 1 Write Confirmation register (for Slave-Only mode) (ep1_wr_cfrm_udc_reg)—Offset 3Ch

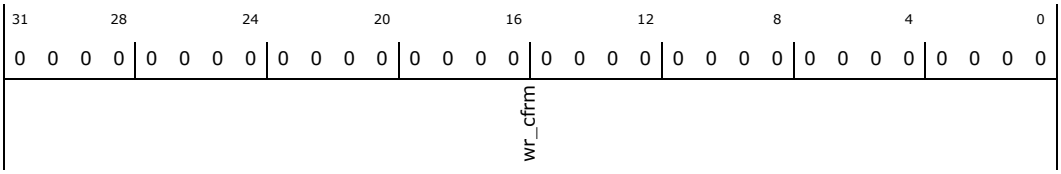
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 3Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Write Confirmation (for Slave-Only mode) (wr_cfrm):** Writing to this register confirms the IN data into the TxFIFO.

16.6.1.13 IN Endpoint 2 Control Register (ep2_in_ctrl_udc_reg)—Offset 40h

This register is used to program the endpoint as required by the application.

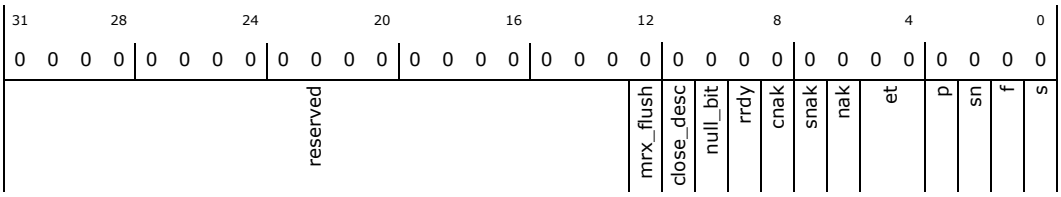
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 40h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 RO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RW	**Send Null Packet (null_bit):** This bit provides the application with a mechanism to instruct the USB Device Controller to send a NULL (zero length) packet when no data is available in the particular endpoint TxFIFO. If this bit is set, when no data is available in the endpoint TxFIFO, the USB Device Controller sends a NULL packet.
9	0x0 RO	**Receive Ready (rrdy):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Bit (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: A NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RW	**Poll Demand (p):** Poll demand from the application. The application can set this bit after an IN token is received from the endpoint. The application can also set this bit before an IN token is received for the endpoint, if it has the IN transfer data in advance. Note: After sending a zero-length or short packet, the application must wait for the next XFERDONE_TXEMPTY interrupt, before setting up the P bit for the next transfer.
2	0x0 RO	**Snoop Mode (sn):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
1	0x0 RW	**Flush TxFIFO (f):** The application firmware sets this bit to 1 after it has detected a disconnect/connect on the USB cable, then waits for the IN token endpoint interrupt before resetting this bit to 0. This flushes the stale data out of the TxFIFO. This bit is cleared by the core when TDC (Transmit DMA Complete) occurs on this endpoint.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.14 IN Endpoint 2 Status Register (ep2_in_sts_udc_reg)—Offset 44h

The Endpoint Status register indicates the endpoint status.

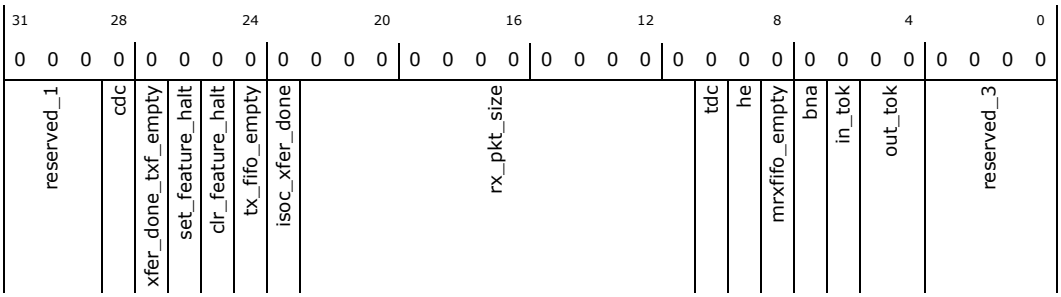
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 44h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RW/1C	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This bit indicates that the TXFIFO is empty after the DMA transfer has been completed. The application can use this bit to set the poll bit for the next transfer. The application must first clear this bit after servicing the interrupt.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint. To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature stall command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RW/1C	**Transmit FIFO Empty detected (tx_fifo_empty):** This bit indicates that the Transmit FIFO Empty condition is triggered. Application can use this information to load the subsequent data into the Transmit FIFO. The application must clear this bit after writing the data into the Transmit FIFO. Note: This bit is not used for Isochronous endpoints.
23	0x0 RW/1C	**Isochronous IN transaction complete (isoc_xfer_done):** This bit indicates that the isochronous IN transaction for this endpoint is complete. The application can use this information to program the isochronous IN data for the next microframe. This bit is used only in Slave-Only mode.
22:11	0x0 RO	**Receive Packet Size (rx_pkt_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
10	0x0 RW/1C	**Transmit DMA Completion (tdc):** Indicates the transmit DMA has completed transferring a descriptor chain data to the Tx FIFO. After servicing the interrupt, the application must clear this bit.
9	0x0 RW/1C	**Error response on the host bus (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	0x0 RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RW/1C	**IN token (in_tok):** An IN token has been received by this endpoint. After servicing the interrupt, application must clear this bit.
5:4	0x0 RO	**OUT token (out_tok):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.15 IN Endpoint 2 Buffer Size Register (ep2_in_bufsize_udc_reg)—Offset 48h

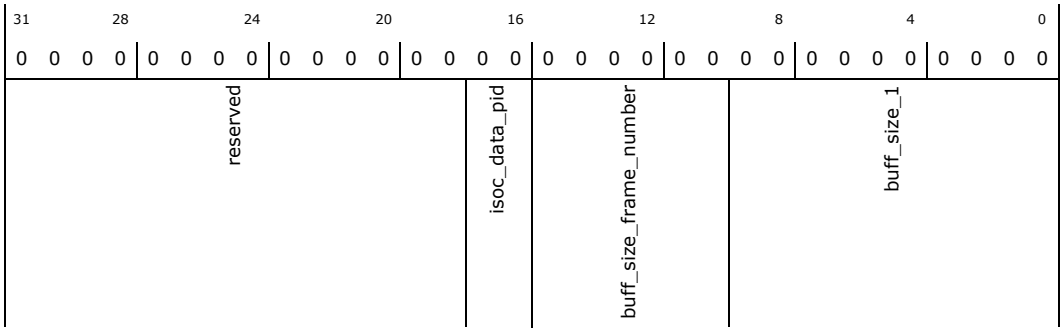
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 48h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RW	**Isochronous IN Transaction PID (isoc_data_pid):** Initial data PID to be sent for a high-bandwidth isochronous IN transaction. This field is used only in Slave-Only mode. 00: DATA0 PID is sent 01: DATA0 PID is sent 10: DATA1 PID is sent 11: DATA2 PID is sent
15:10	0x0 RO	**Buffer Size (buff_size_frame_number):** These bits are reserved and should be set to zero.
9:0	0x0 RW	**Buffer Size (buff_size_1):** Buffer size required for this endpoint. The application can program this field to make each endpoints buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Transmit FIFO. NOTE: the maximum size of the TxFIFO is 1024 32-bit word entries.

16.6.1.16 IN Endpoint 2 Maximum Packet Size Register (ep2_in_mpkt_sz_reg)—Offset 4Ch

This register also specifies the maximum packet size an endpoint should support. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the USB Device Controller register space.

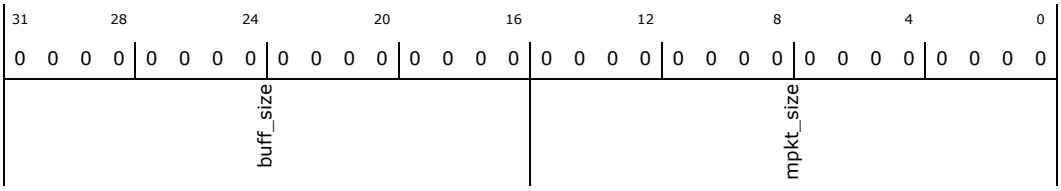
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RO	**Buffer Size (buff_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
15:0	0x0 RO	**Max Packet Size (mpkt_size):** Maximum packet size for the endpoint in bytes. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.17 IN Endpoint 2 Data Descriptor Pointer Register (ep2_in_desptr_udc_reg)—Offset 54h

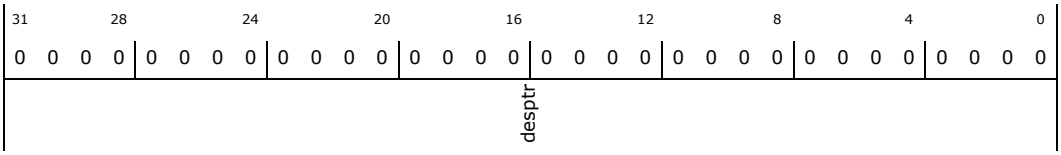
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 54h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointers.

16.6.1.18 IN Endpoint 2 Write Confirmation register (for Slave-Only mode) (ep2_wr_cfrm_udc_reg)—Offset 5Ch

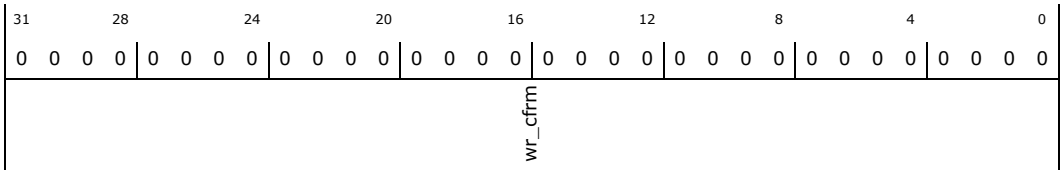
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 5Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Write Confirmation (for Slave-Only mode) (wr_cfrm):** Writing to this register confirms the IN data into the TxFIFO.

16.6.1.19 IN Endpoint 3 Control Register (ep3_in_ctrl_udc_reg)—Offset 60h

This register is used to program the endpoint as required by the application.

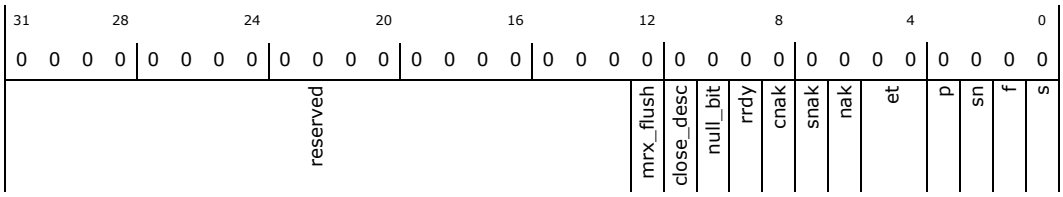
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 60h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 RO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RW	**Send Null Packet (null_bit):** This bit provides the application with a mechanism to instruct the USB Device Controller to send a NULL (zero length) packet when no data is available in the particular endpoint TxFIFO. If this bit is set, when no data is available in the endpoint TxFIFO, the USB Device Controller sends a NULL packet.
9	0x0 RO	**Receive Ready (rrdy):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Bit (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: A NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RW	**Poll Demand (p):** Poll demand from the application. The application can set this bit after an IN token is received from the endpoint. The application can also set this bit before an IN token is received for the endpoint, if it has the IN transfer data in advance. Note: After sending a zero-length or short packet, the application must wait for the next XFERDONE_TXEMPTY interrupt, before setting up the P bit for the next transfer.
2	0x0 RO	**Snoop Mode (sn):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
1	0x0 RW	**Flush TxFIFO (f):** The application firmware sets this bit to 1 after it has detected a disconnect/connect on the USB cable, then waits for the IN token endpoint interrupt before resetting this bit to 0. This flushes the stale data out of the TxFIFO. This bit is cleared by the core when TDC (Transmit DMA Complete) occurs on this endpoint.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.20 IN Endpoint 3 Status Register (ep3_in_sts_udc_reg)—Offset 64h

The Endpoint Status register indicates the endpoint status.

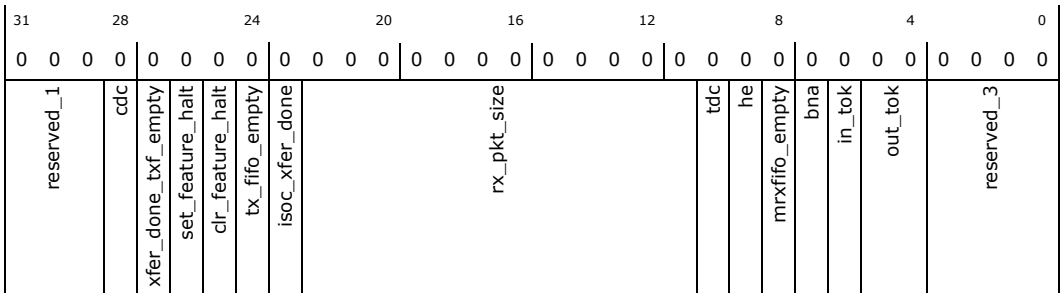
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 64h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RW/1C	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This bit indicates that the TXFIFO is empty after the DMA transfer has been completed. The application can use this bit to set the poll bit for the next transfer. The application must first clear this bit after servicing the interrupt.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint. To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature stall command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RW/1C	**Transmit FIFO Empty detected (tx_fifo_empty):** This bit indicates that the Transmit FIFO Empty condition is triggered. Application can use this information to load the subsequent data into the Transmit FIFO. The application must clear this bit after writing the data into the Transmit FIFO. Note: This bit is not used for Isochronous endpoints.
23	0x0 RW/1C	**Isochronous IN transaction complete (isoc_xfer_done):** This bit indicates that the isochronous IN transaction for this endpoint is complete. The application can use this information to program the isochronous IN data for the next microframe. This bit is used only in Slave-Only mode.
22:11	0x0 RO	**Receive Packet Size (rx_pkt_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
10	0x0 RW/1C	**Transmit DMA Completion (tdc):** Indicates the transmit DMA has completed transferring a descriptor chain data to the Tx FIFO. After servicing the interrupt, the application must clear this bit.
9	0x0 RW/1C	**Error response on the host bus (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	0x0 RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RW/1C	**IN token (in_tok):** An IN token has been received by this endpoint. After servicing the interrupt, application must clear this bit.
5:4	0x0 RO	**OUT token (out_tok):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.21 IN Endpoint 3 Buffer Size Register (ep3_in_bufsize_udc_reg)—Offset 68h

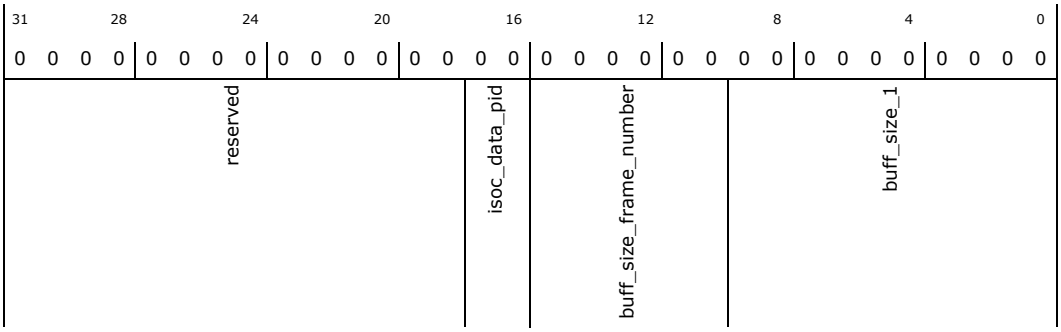
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 68h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RW	**Isochronous IN Transaction PID (isoc_data_pid):** Initial data PID to be sent for a high-bandwidth isochronous IN transaction. This field is used only in Slave-Only mode. 00: DATA0 PID is sent 01: DATA0 PID is sent 10: DATA1 PID is sent 11: DATA2 PID is sent
15:10	0x0 RO	**Buffer Size (buff_size_frame_number):** These bits are reserved and should be set to zero.
9:0	0x0 RW	**Buffer Size (buff_size_1):** Buffer size required for this endpoint. The application can program this field to make each endpoints buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Transmit FIFO. NOTE: the maximum size of the TxFIFO is 1024 32-bit word entries.

16.6.1.22 IN Endpoint 3 Maximum Packet Size Register (ep3_in_mpkt_sz_reg)—Offset 6Ch

This register also specifies the maximum packet size an endpoint should support. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the USB Device Controller register space.

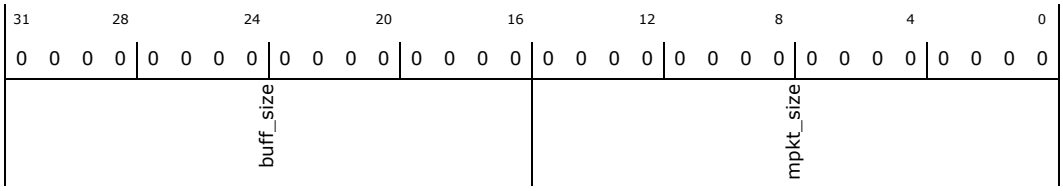
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 6Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RO	**Buffer Size (buff_size):** This is an OUT endpoint field only. These bits are reserved and should be set to zero.
15:0	0x0 RO	**Max Packet Size (mpkt_size):** Maximum packet size for the endpoint in bytes. This maximum size is used to calculate whether the Receive FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.23 IN Endpoint 3 Data Descriptor Pointer Register (ep3_in_desptr_udc_reg)—Offset 74h

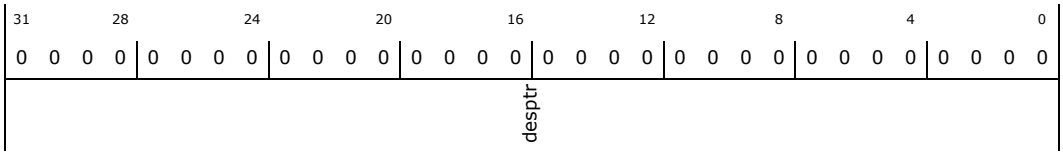
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 74h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointers.

16.6.1.24 IN Endpoint 3 Write Confirmation register (for Slave-Only mode) (ep3_wr_cfrm_udc_reg)—Offset 7Ch

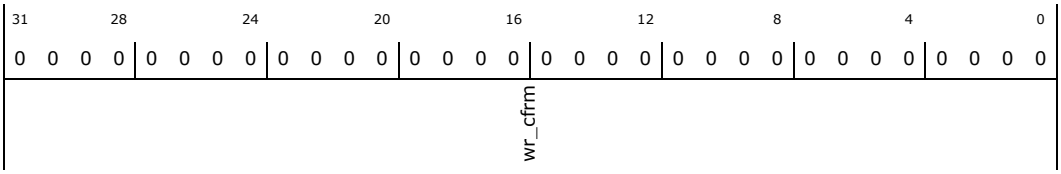
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 7Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Write Confirmation (for Slave-Only mode) (wr_cfrm):** Writing to this register confirms the IN data into the TxFIFO.

16.6.1.25 OUT Endpoint 0 Control Register (ep0_out_ctrl_udc_reg)—Offset 200h

This register is used to program the endpoint as required by the application.

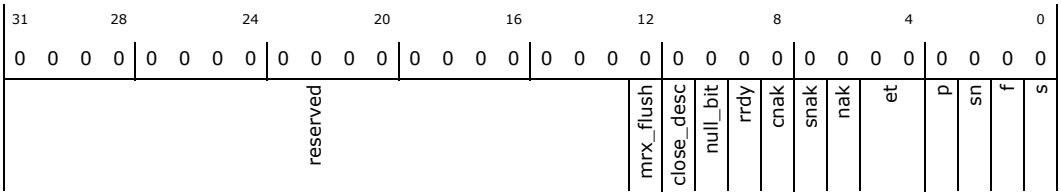
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 200h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 WO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** When the application wants to flush the endpoint receive FIFO, it must first set the SNAK bit in the Endpoint Control register. If the receive DMA is in progress, then the core will finish the current descriptor, terminate the DMA, and flush the data in the endpoint receive FIFO and clear this bit. The application must clear this bit after the EP receive FIFO is empty, by checking the MRXFIFO EMPTY bit in the Endpoint Status register.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RO	**Send Null Packet (null_bit):** This is an IN endpoint field only. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
9	0x0 RW	**Receive Ready (rrdy):** If this bit is set by the application, on receiving an OUT packet, the DMA sends the packet to system memory. This bit is deasserted at the end of packet if the Descriptor Update bit is set in the Device Control register. This bit is deasserted at the end of payload if the Descriptor Update bit is deasserted. This bit can be set by the application at any time. The application cannot clear this bit if the DMA is busy transferring the data.
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK. (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Handshake Enable (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: The NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** Endpoint Type (2-bit). The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RO	**Poll Demand (p):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
2	0x0 RW	**Snoop Mode (sn):** Configures the endpoint for Snoop mode. In this mode, the subsystem does not check the correctness of OUT packets before transferring them to application memory.
1	0x0 RO	**Flush TxFIFO (f):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.26 OUT Endpoint 0 Status Register (ep0_out_sts_udc_reg)—Offset 204h

The Endpoint Status register indicates the endpoint status.

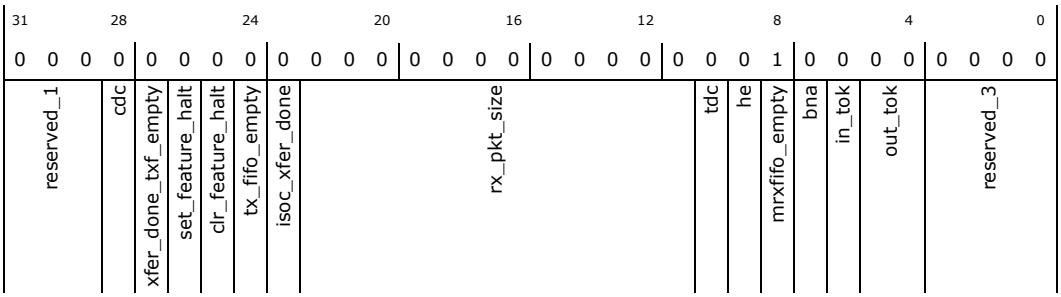
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 204h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000100h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RO	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint .To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature sta command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RO	**Transmit FIFO Empty detected (tx_fifo_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
23	0x0 RO	**Isochronous IN transaction complete (isoc_xfer_done):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
22:11	0x0 RW	**Receive Packet Size (rx_pkt_size):** Indicates the number of bytes in the current receive packet the RxFIFO is receiving. Because the USB host always sends 8 bytes of SETUP data, these bits do not indicate the receipt of 8 bytes of SETUP data for a SETUP packet. Rather, these bits indicate the configuration status (Configuration number [22:19], Interface number [18:15], and Alternate Setting number [14:11]). This field is used in slave mode only. In DMA mode, the application must check the status from the endpoint data descriptor.
10	0x0 RO	**Transmit DMA Completion (tdc):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW/1C	**System Host Error (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.

Bit Range	Default & Access	Field Name (ID): Description
8	1h RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This bit indicates the empty status of the endpoint receive address FIFO. This bit is set by the core after the DMA transfers data to system memory, and there are no new packets received from the USB. This bit is cleared by the core after receiving a valid packet from the USB. 1: EP RXFIFO is empty 0: EP RXFIFO is not empty
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RO	**IN token (in_tok):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
5:4	0x0 RW/1C	**OUT token. (out_tok):** An OUT packet has been received by this endpoint. The encoding of these two bits indicates the type of data received. The possible options are: 00: None 01: Received data 10: Received SETUP data (8 bytes) 11: Reserved (The application must write the same values to clear these bits.)
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.27 OUT Endpoint 0 Receive Packet Frame Number Register (ep0_out_rpf_udc_reg)—Offset 208h

This register contains the frame number in which the packet is received. This frame number information is useful when handling isochronous traffic.

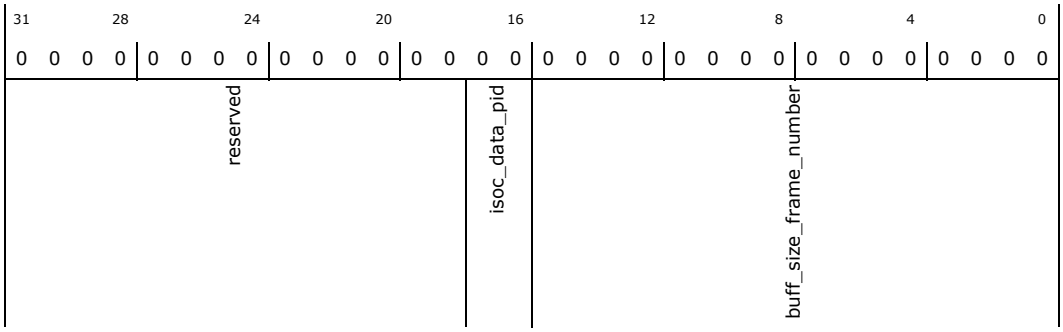
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 208h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
17:16	0x0 RO	**Isochronous OUT Transaction PID (isoc_data_pid):** Data PID received for a high-bandwidth isochronous OUT transaction. This field indicates that the data PID for the current packet is available in the Receive FIFO. This field is used only in Slave-Only mode. 00: DATA0 PID is received 01: DATA1 PID is received 10: DATA2 PID is received 11: MDATA PID is received
15:0	0x0 RO	**Buffer Size/Frame Number (buff_size_frame_number):** Frame number in which the packet is received. For high-speed operation: [15:14] Reserved [13:3] Millisecond frame number [2:0] Microframe number For full-speed operation: [15:11] Reserved [10:0] Millisecond frame number

16.6.1.28 OUT Endpoint 0 Buffer Size Register (ep0_out_bufsize_udc_reg)—Offset 20Ch

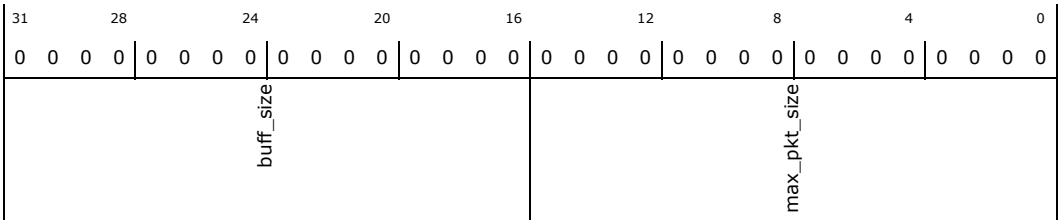
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 20Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RW	**Packet Size (buff_size):** Buffer size required for this endpoint. The application can program this field to make each endpoint buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Receive FIFO. This value cannot be changed dynamically during operation. NOTE: the maximum size of the RxFIFO is 512 32-bit word entries.
15:0	0x0 RW	**Max Packet Size (max_pkt_size):** Maximum packet size for the endpoint. This is the value in bytes. This maximum size is used to calculate whether the Transmit FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.29 OUT Endpoint 0 SETUP Buffer Pointer Register (ep0_subptr_udc_reg)—Offset 210h

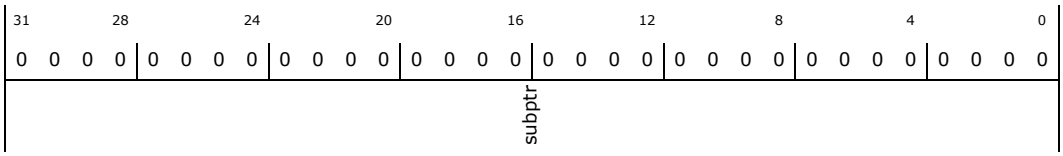
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 210h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**SETUP Buffer Pointer (subptr):** Endpoint SETUP buffer pointers are used for SETUP commands. The Endpoint SETUP Buffer Pointer register is used only in DMA mode. NOTE: This is applicable only to control endpoints. For all other endpoints this register is reserved.

16.6.1.30 OUT Endpoint 0 Data Descriptor Pointer Register (ep0_out_desptr_udc_reg)—Offset 214h

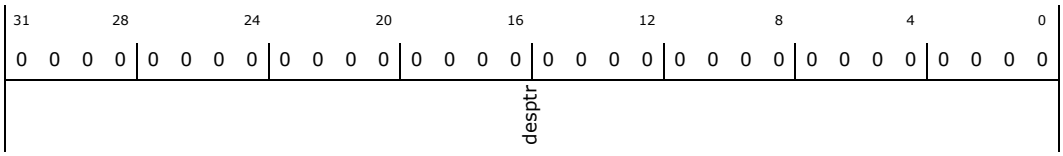
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 214h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointer.

16.6.1.31 OUT Endpoint 0 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep0_rd_cfrm_udc_reg)—Offset 21Ch

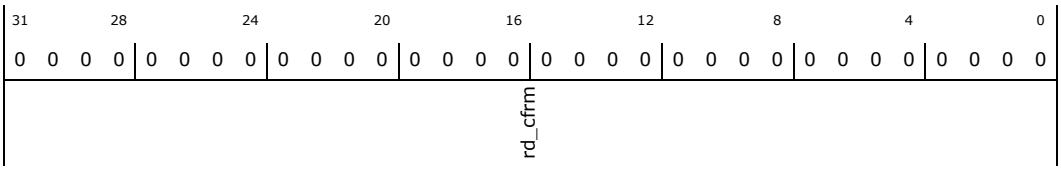
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 21Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Read Confirmation for zero-length OUT data (for Slave-Only mode) (rd_cfrm):** For zero-length OUT data, the application must perform a dummy read from this register

16.6.1.32 OUT Endpoint 1 Control Register (ep1_out_ctrl_udc_reg)—Offset 220h

This register is used to program the endpoint as required by the application.

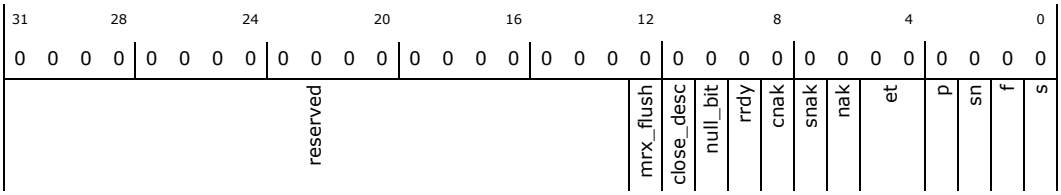
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 220h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 WO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** When the application wants to flush the endpoint receive FIFO, it must first set the SNAK bit in the Endpoint Control register. If the receive DMA is in progress, then the core will finish the current descriptor, terminate the DMA, and flush the data in the endpoint receive FIFO and clear this bit. The application must clear this bit after the EP receive FIFO is empty, by checking the MRXFIFO EMPTY bit in the Endpoint Status register.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RO	**Send Null Packet (null_bit):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW	**Receive Ready (rrdy):** If this bit is set by the application, on receiving an OUT packet, the DMA sends the packet to system memory. This bit is deasserted at the end of packet if the Descriptor Update bit is set in the Device Control register. This bit is deasserted at the end of payload if the Descriptor Update bit is deasserted. This bit can be set by the application at any time. The application cannot clear this bit if the DMA is busy transferring the data.

Bit Range	Default & Access	Field Name (ID): Description
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.
7	0x0 WO	**Set NAK. (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Handshake Enable (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: The NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** Endpoint Type (2-bit). The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RO	**Poll Demand (p):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
2	0x0 RW	**Snoop Mode (sn):** Configures the endpoint for Snoop mode. In this mode, the subsystem does not check the correctness of OUT packets before transferring them to application memory.
1	0x0 RO	**Flush TxFIFO (f):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.33 OUT Endpoint 1 Status Register (ep1_out_sts_udc_reg)—Offset 224h

The Endpoint Status register indicates the endpoint status.

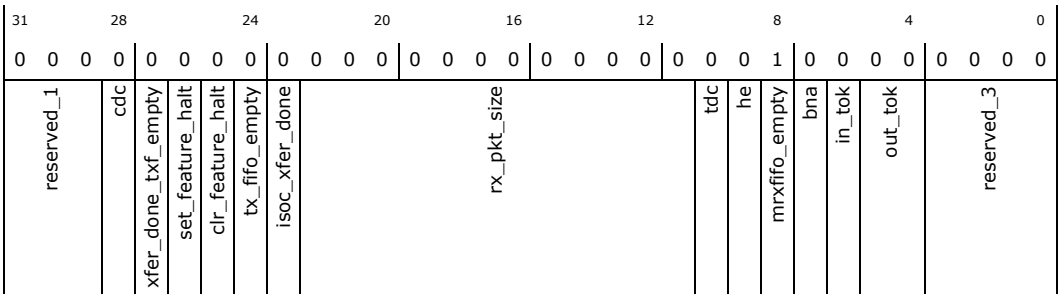
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 224h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000100h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RO	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint .To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature sta command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RO	**Transmit FIFO Empty detected (tx_fifo_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
23	0x0 RO	**Isochronous IN transaction complete (isoc_xfer_done):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
22:11	0x0 RW	**Receive Packet Size (rx_pkt_size):** Indicates the number of bytes in the current receive packet the RxFIFO is receiving. Because the USB host always sends 8 bytes of SETUP data, these bits do not indicate the receipt of 8 bytes of SETUP data for a SETUP packet. Rather, these bits indicate the configuration status (Configuration number [22:19], Interface number [18:15], and Alternate Setting number [14:11]). This field is used in slave mode only. In DMA mode, the application must check the status from the endpoint data descriptor.
10	0x0 RO	**Transmit DMA Completion (tdc):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW/1C	**System Host Error (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	1h RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This bit indicates the empty status of the endpoint receive address FIFO. This bit is set by the core after the DMA transfers data to system memory, and there are no new packets received from the USB. This bit is cleared by the core after receiving a valid packet from the USB. 1: EP RXFIFO is empty 0: EP RXFIFO is not empty

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.
6	0x0 RO	**IN token (in_tok):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
5:4	0x0 RW/1C	**OUT token. (out_tok):** An OUT packet has been received by this endpoint. The encoding of these two bits indicates the type of data received. The possible options are: 00: None 01: Received data 10: Received SETUP data (8 bytes) 11: Reserved (The application must write the same values to clear these bits.)
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.34 OUT Endpoint 1 Receive Packet Frame Number Register (ep1_out_rpf_udc_reg)—Offset 228h

This register contains the frame number in which the packet is received. This frame number information is useful when handling isochronous traffic.

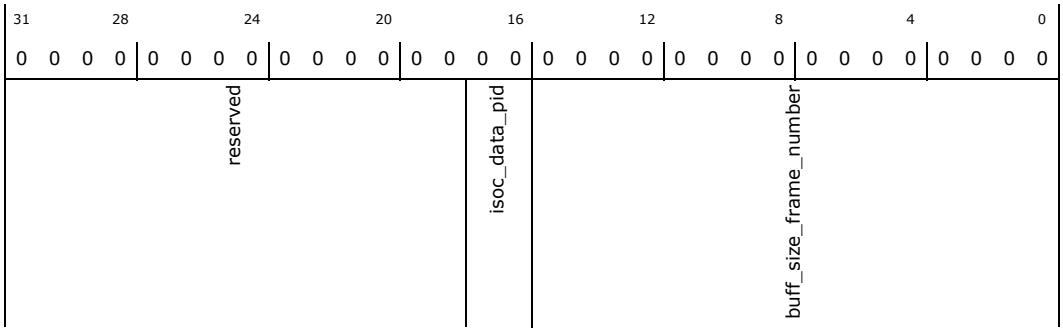
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 228h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RO	**Isochronous OUT Transaction PID (isoc_data_pid):** Data PID received for a high-bandwidth isochronous OUT transaction. This field indicates that the data PID for the current packet is available in the Receive FIFO. This field is used only in Slave-Only mode. 00: DATA0 PID is received 01: DATA1 PID is received 10: DATA2 PID is received 11: MDATA PID is received

Bit Range	Default & Access	Field Name (ID): Description
15:0	0x0 RO	**Buffer Size/Frame Number (buff_size_frame_number):** Frame number in which the packet is received. For high-speed operation: [15:14] Reserved [13:3] Millisecond frame number [2:0] Microframe number For full-speed operation: [15:11] Reserved [10:0] Millisecond frame number

16.6.1.35 OUT Endpoint 1 Buffer Size Register (ep1_out_bufsize_udc_reg)—Offset 22Ch

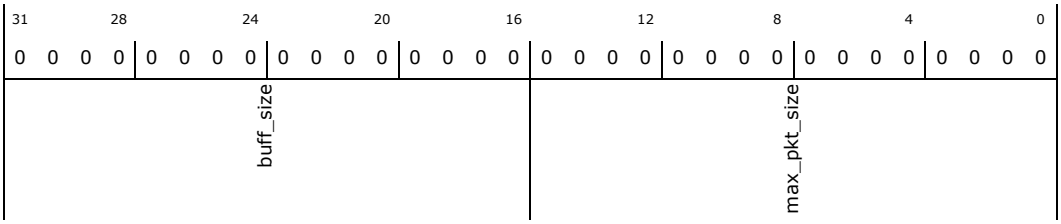
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 22Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RW	**Packet Size (buff_size):** Buffer size required for this endpoint. The application can program this field to make each endpoint buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Receive FIFO. This value cannot be changed dynamically during operation. NOTE: the maximum size of the RxFIFO is 512 32-bit word entries.
15:0	0x0 RW	**Max Packet Size (max_pkt_size):** Maximum packet size for the endpoint. This is the value in bytes. This maximum size is used to calculate whether the Transmit FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.36 OUT Endpoint 1 SETUP Buffer Pointer Register (ep1_subptr_udc_reg)—Offset 230h

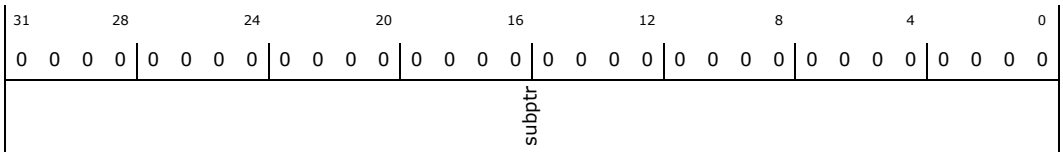
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 230h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**SETUP Buffer Pointer (subptr):** Endpoint SETUP buffer pointers are used for SETUP commands. The Endpoint SETUP Buffer Pointer register is used only in DMA mode. NOTE: This is applicable only to control endpoints. For all other endpoints this register is reserved.

16.6.1.37 OUT Endpoint 1 Data Descriptor Pointer Register (ep1_out_desptr_udc_reg)—Offset 234h

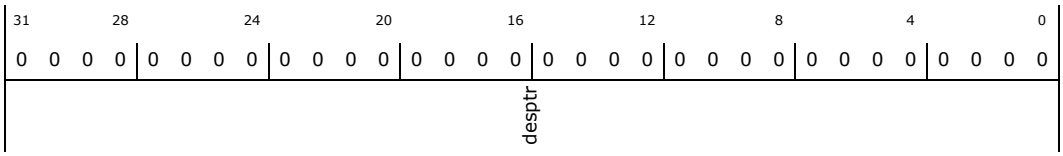
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 234h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointer.

16.6.1.38 OUT Endpoint 1 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep1_rd_cfrm_udc_reg)—Offset 23Ch

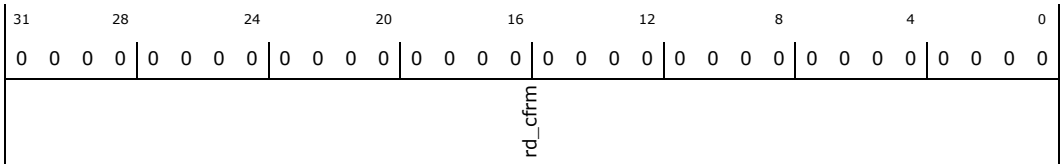
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 23Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Read Confirmation for zero-length OUT data (for Slave-Only mode) (rd_cfrm):** For zero-length OUT data, the application must perform a dummy read from this register

16.6.1.39 OUT Endpoint 2 Control Register (ep2_out_ctrl_udc_reg)—Offset 240h

This register is used to program the endpoint as required by the application.

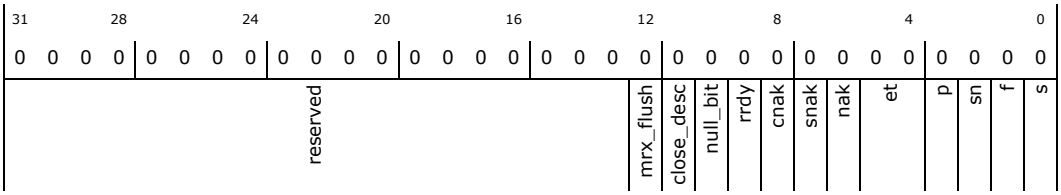
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 240h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 WO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** When the application wants to flush the endpoint receive FIFO, it must first set the SNAK bit in the Endpoint Control register. If the receive DMA is in progress, then the core will finish the current descriptor, terminate the DMA, and flush the data in the endpoint receive FIFO and clear this bit. The application must clear this bit after the EP receive FIFO is empty, by checking the MRXFIFO EMPTY bit in the Endpoint Status register.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RO	**Send Null Packet (null_bit):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW	**Receive Ready (rrdy):** If this bit is set by the application, on receiving an OUT packet, the DMA sends the packet to system memory. This bit is deasserted at the end of packet if the Descriptor Update bit is set in the Device Control register. This bit is deasserted at the end of payload if the Descriptor Update bit is deasserted. This bit can be set by the application at any time. The application cannot clear this bit if the DMA is busy transferring the data.
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 WO	**Set NAK. (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Handshake Enable (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: The NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** Endpoint Type (2-bit). The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RO	**Poll Demand (p):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
2	0x0 RW	**Snoop Mode (sn):** Configures the endpoint for Snoop mode. In this mode, the subsystem does not check the correctness of OUT packets before transferring them to application memory.
1	0x0 RO	**Flush TxFIFO (f):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.40 OUT Endpoint 2 Status Register (ep2_out_sts_udc_reg)—Offset 244h

The Endpoint Status register indicates the endpoint status.

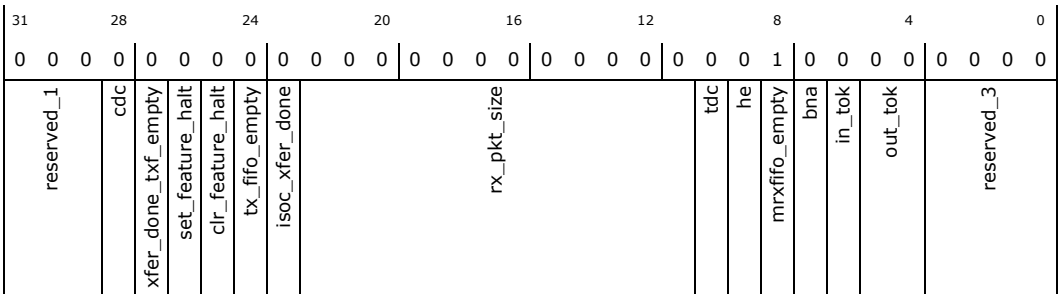
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 244h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000100h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RO	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint .To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature sta command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RO	**Transmit FIFO Empty detected (tx_fifo_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
23	0x0 RO	**Isochronous IN transaction complete (isoc_xfer_done):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
22:11	0x0 RW	**Receive Packet Size (rx_pkt_size):** Indicates the number of bytes in the current receive packet the RxFIFO is receiving. Because the USB host always sends 8 bytes of SETUP data, these bits do not indicate the receipt of 8 bytes of SETUP data for a SETUP packet. Rather, these bits indicate the configuration status (Configuration number [22:19], Interface number [18:15], and Alternate Setting number [14:11]). This field is used in slave mode only. In DMA mode, the application must check the status from the endpoint data descriptor.
10	0x0 RO	**Transmit DMA Completion (tdc):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW/1C	**System Host Error (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	1h RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This bit indicates the empty status of the endpoint receive address FIFO. This bit is set by the core after the DMA transfers data to system memory, and there are no new packets received from the USB. This bit is cleared by the core after receiving a valid packet from the USB. 1: EP RXFIFO is empty 0: EP RXFIFO is not empty
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.

Bit Range	Default & Access	Field Name (ID): Description
6	0x0 RO	**IN token (in_tok):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
5:4	0x0 RW/1C	**OUT token. (out_tok):** An OUT packet has been received by this endpoint. The encoding of these two bits indicates the type of data received. The possible options are: 00: None 01: Received data 10: Received SETUP data (8 bytes) 11: Reserved (The application must write the same values to clear these bits.)
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.41 OUT Endpoint 2 Receive Packet Frame Number Register (ep2_out_rpf_udc_reg)—Offset 248h

This register contains the frame number in which the packet is received. This frame number information is useful when handling isochronous traffic.

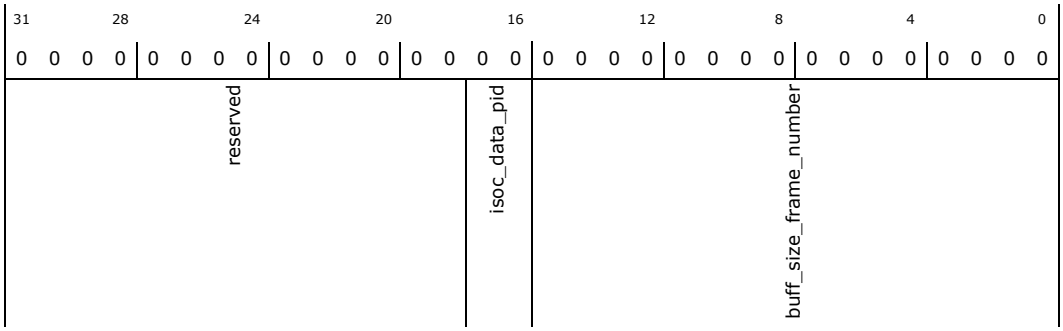
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 248h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RO	**Isochronous OUT Transaction PID (isoc_data_pid):** Data PID received for a high-bandwidth isochronous OUT transaction. This field indicates that the data PID for the current packet is available in the Receive FIFO. This field is used only in Slave-Only mode. 00: DATA0 PID is received 01: DATA1 PID is received 10: DATA2 PID is received 11: MDATA PID is received

Bit Range	Default & Access	Field Name (ID): Description
15:0	0x0 RO	**Buffer Size/Frame Number (buff_size_frame_number):** Frame number in which the packet is received. For high-speed operation: [15:14] Reserved [13:3] Millisecond frame number [2:0] Microframe number For full-speed operation: [15:11] Reserved [10:0] Millisecond frame number

16.6.1.42 OUT Endpoint 2 Buffer Size Register (ep2_out_bufsize_udc_reg)—Offset 24Ch

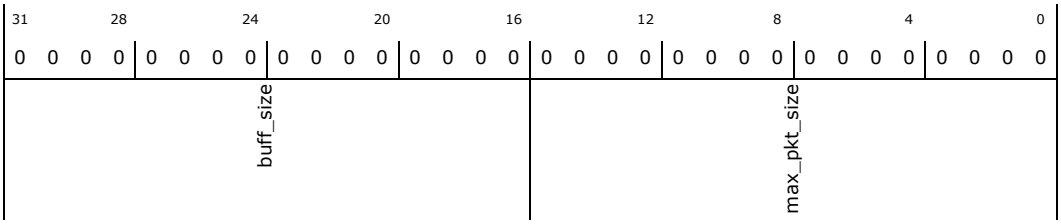
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 24Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RW	**Packet Size (buff_size):** Buffer size required for this endpoint. The application can program this field to make each endpoint buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Receive FIFO. This value cannot be changed dynamically during operation. NOTE: the maximum size of the RxFIFO is 512 32-bit word entries.
15:0	0x0 RW	**Max Packet Size (max_pkt_size):** Maximum packet size for the endpoint. This is the value in bytes. This maximum size is used to calculate whether the Transmit FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.43 OUT Endpoint 2 SETUP Buffer Pointer Register (ep2_subptr_udc_reg)—Offset 250h

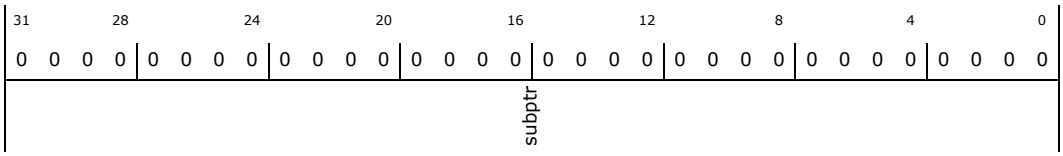
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 250h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**SETUP Buffer Pointer (subptr):** Endpoint SETUP buffer pointers are used for SETUP commands. The Endpoint SETUP Buffer Pointer register is used only in DMA mode. NOTE: This is applicable only to control endpoints. For all other endpoints this register is reserved.

16.6.1.44 OUT Endpoint 2 Data Descriptor Pointer Register (ep2_out_desptr_udc_reg)—Offset 254h

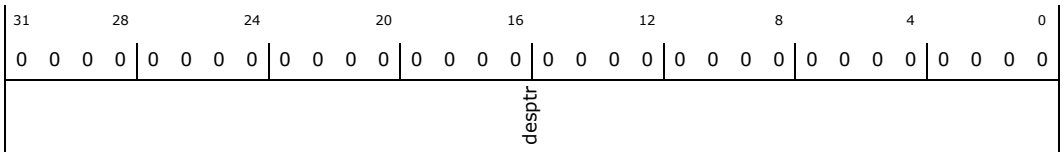
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 254h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointer.

16.6.1.45 OUT Endpoint 2 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep2_rd_cfrm_udc_reg)—Offset 25Ch

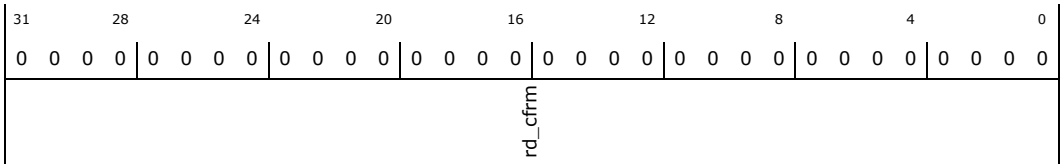
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 25Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Read Confirmation for zero-length OUT data (for Slave-Only mode) (rd_cfrm):** For zero-length OUT data, the application must perform a dummy read from this register

16.6.1.46 OUT Endpoint 3 Control Register (ep3_out_ctrl_udc_reg)—Offset 260h

This register is used to program the endpoint as required by the application.

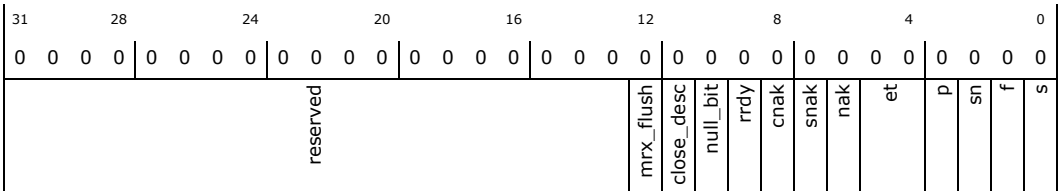
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 260h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
12	0x0 WO	**Receive FIFO Flush for Multiple Receive FIFO (mrx_flush):** When the application wants to flush the endpoint receive FIFO, it must first set the SNAK bit in the Endpoint Control register. If the receive DMA is in progress, then the core will finish the current descriptor, terminate the DMA, and flush the data in the endpoint receive FIFO and clear this bit. The application must clear this bit after the EP receive FIFO is empty, by checking the MRXFIFO EMPTY bit in the Endpoint Status register.
11	0x0 RO	**Close Descriptor Channel (close_desc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
10	0x0 RO	**Send Null Packet (null_bit):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW	**Receive Ready (rrdy):** If this bit is set by the application, on receiving an OUT packet, the DMA sends the packet to system memory. This bit is deasserted at the end of packet if the Descriptor Update bit is set in the Device Control register. This bit is deasserted at the end of payload if the Descriptor Update bit is deasserted. This bit can be set by the application at any time. The application cannot clear this bit if the DMA is busy transferring the data.
8	0x0 WO	**Clear NAK (cnak):** Used by the application to clear the NAK bit (bit 6). After the subsystem sets bit 6 (NAK), the application must clear it with a write of 1 to the CNAK bit. (For example, after the application has decoded the SETUP packet and determined it is not an invalid command, the application must set the CNAK bit of the control endpoint to 1 to clear the NAK bit.)The application also must clear the NAK bit whenever the subsystem sets it. (The subsystem sets it due to the application setting the Stall bit.)The application can clear this bit only when the RxFIFO is empty (for single RxFIFO implementation) or when the RxFIFO corresponding to the same logical endpoint is empty (Multiple RxFIFO implementation). The application can write CNAK any time without waiting for a RxFIFO empty condition. The NAK bit is cleared immediately upon write to CNAK bit. No polling is necessary.

Bit Range	Default & Access	Field Name (ID): Description
7	0x0 WO	**Set NAK. (snak):** Used by the application to set the NAK bit (bit 6 of this register). The application must not set the NAK bit for an IN endpoint until it has received an IN token interrupt indicating that the TxFIFO is empty.
6	0x0 RO	**NAK Handshake Enable (nak):** After a SETUP packet, which is decoded by the application, is received by the core, the core sets the NAK bit for all control IN/OUT endpoints. NAK is also set after a STALL response for the endpoint (the STALL bit is set in Endpoint Control register bit 0). 1: The endpoint responds to the USB host with a NAK handshake. 0: The endpoint responds normally. Note 1: A SETUP packet is sent to the application regardless of whether the NAK bit is set. Note 2: The NAK set on ISOC OUT endpoint rejects the ISOC out data packet.
5:4	0x0 RW	**Endpoint Type (et):** Endpoint Type (2-bit). The possible options are: 00: Control endpoint 01: Isochronous endpoint 10: Bulk endpoint 11: Interrupt endpoint
3	0x0 RO	**Poll Demand (p):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
2	0x0 RW	**Snoop Mode (sn):** Configures the endpoint for Snoop mode. In this mode, the subsystem does not check the correctness of OUT packets before transferring them to application memory.
1	0x0 RO	**Flush TxFIFO (f):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
0	0x0 RW	**Stall Handshake (s):** On successful reception of a SETUP packet (decoded by the application), the subsystem clears both IN and OUT Stall bits, and sets both the IN and OUT NAK bits. The application must check for RxFIFO emptiness before setting the IN and OUT STALL bit. For non-SETUP packets, the subsystem clears either IN or out STALL bits only if a STALL handshake is returned to the USB host, then sets the corresponding NAK bit. The subsystem returns a STALL handshake for the subsequent transactions of the stalled endpoint until the USB host issues a Clear_Feature command to clear it. Once this bit is set, if the subsystem has already returned a STALL handshake to the USB host, the application firmware cannot clear the S bit to stop the subsystem from sending the STALL handshake on the endpoint. The host must instead send one of the following commands to clear the endpoint Halt status: - Clear Feature (Halt) - SetConfiguration - SetInterface.

16.6.1.47 OUT Endpoint 3 Status Register (ep3_out_sts_udc_reg)—Offset 264h

The Endpoint Status register indicates the endpoint status.

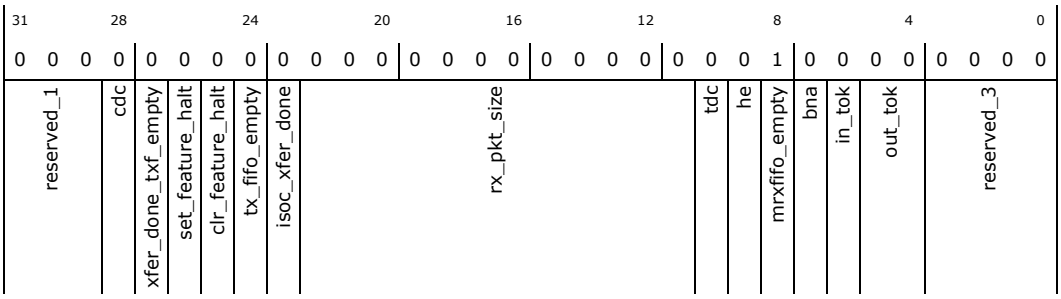
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 264h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000100h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
28	0x0 RO	**Close Descriptor Channel (cdc):** Close Descriptor Enhancement is not supported. These bits are reserved and should be set to zero.
27	0x0 RO	**Transfer Done/Transmit FIFO Empty (xfer_done_txf_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
26	0x0 RW/1C	**Received Set Halt (set_feature_halt):** This bit indicates that the Set Feature (EP HALT) command is received for this endpoint .To stall this endpoint, the application can set the S bit in Endpoint Control Register. After this, the application clears this RSS bit to acknowledge the reception of Set Feature sta command. Once this RSS bit is cleared, core sends the zero-length packet for the Status IN phase of Set Feature command. Received Set Stall indication is applicable only for Bulk and Interrupt transactions.
25	0x0 RW/1C	**Received Clear Halt (clr_feature_halt):** This bit indicates that the Clear Feature (EP HALT) command is received for this endpoint. To continue to stall the endpoint, the application must set the S bit Endpoint Control Register. After this, the application clears this RCS bit to acknowledge the reception of clear stall command. Once this bit is cleared, core sends the zero-length packet for the Status IN phase of clear stall command. Received Clear Stall indication is applicable only for Bulk and Interrupt transactions.
24	0x0 RO	**Transmit FIFO Empty detected (tx_fifo_empty):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
23	0x0 RO	**Isochronous IN transaction complete (isoc_xfer_done):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
22:11	0x0 RW	**Receive Packet Size (rx_pkt_size):** Indicates the number of bytes in the current receive packet the RxFIFO is receiving. Because the USB host always sends 8 bytes of SETUP data, these bits do not indicate the receipt of 8 bytes of SETUP data for a SETUP packet. Rather, these bits indicate the configuration status (Configuration number [22:19], Interface number [18:15], and Alternate Setting number [14:11]). This field is used in slave mode only. In DMA mode, the application must check the status from the endpoint data descriptor.
10	0x0 RO	**Transmit DMA Completion (tdc):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
9	0x0 RW/1C	**System Host Error (he):** Error response on the host bus (AHB) when doing a data transfer, descriptor fetch, or descriptor update for this particular endpoint. After servicing the interrupt, the application must clear this bit.
8	1h RO	**Receive Address FIFO Empty Status (mrxfifo_empty):** This bit indicates the empty status of the endpoint receive address FIFO. This bit is set by the core after the DMA transfers data to system memory, and there are no new packets received from the USB. This bit is cleared by the core after receiving a valid packet from the USB. 1: EP RXFIFO is empty 0: EP RXFIFO is not empty
7	0x0 RW/1C	**Buffer Not Available (bna):** The subsystem sets this bit when the descriptor status is either Host Busy or DMA Done to indicate that the descriptor was not ready at the time the DMA tried to access it. After servicing the interrupt, the application must clear this bit.

Bit Range	Default & Access	Field Name (ID): Description
6	0x0 RO	**IN token (in_tok):** This is an IN endpoint field only. These bits are reserved and should be set to zero.
5:4	0x0 RW/1C	**OUT token. (out_tok):** An OUT packet has been received by this endpoint. The encoding of these two bits indicates the type of data received. The possible options are: 00: None 01: Received data 10: Received SETUP data (8 bytes) 11: Reserved (The application must write the same values to clear these bits.)
3:0	0x0 RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

16.6.1.48 OUT Endpoint 3 Receive Packet Frame Number Register (ep3_out_rpf_udc_reg)—Offset 268h

This register contains the frame number in which the packet is received. This frame number information is useful when handling isochronous traffic.

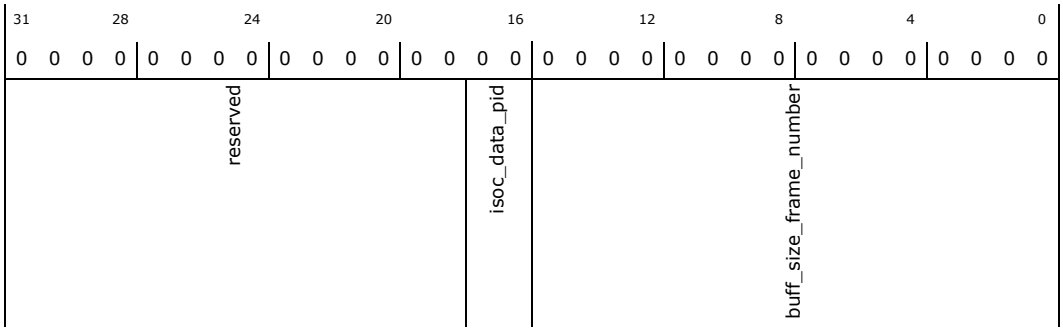
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 268h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
17:16	0x0 RO	**Isochronous OUT Transaction PID (isoc_data_pid):** Data PID received for a high-bandwidth isochronous OUT transaction. This field indicates that the data PID for the current packet is available in the Receive FIFO. This field is used only in Slave-Only mode. 00: DATA0 PID is received 01: DATA1 PID is received 10: DATA2 PID is received 11: MDATA PID is received

Bit Range	Default & Access	Field Name (ID): Description
15:0	0x0 RO	**Buffer Size/Frame Number (buff_size_frame_number):** Frame number in which the packet is received. For high-speed operation: [15:14] Reserved [13:3] Millisecond frame number [2:0] Microframe number For full-speed operation: [15:11] Reserved [10:0] Millisecond frame number

16.6.1.49 OUT Endpoint 3 Buffer Size Register (ep3_out_bufsize_udc_reg)—Offset 26Ch

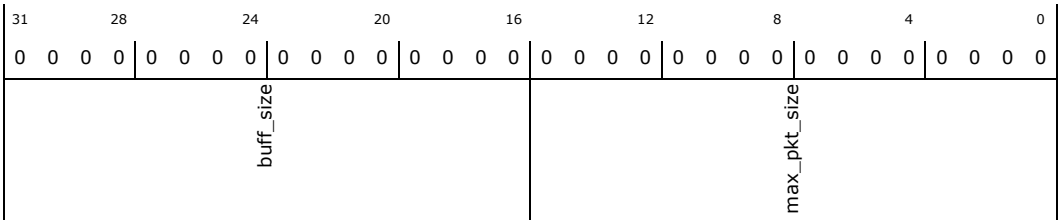
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 26Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RW	**Packet Size (buff_size):** Buffer size required for this endpoint. The application can program this field to make each endpoint buffers adaptive, providing flexibility in buffer size when the interface or configuration is changed. This value is in 32-bit words, and indicates the number of 32-bit word entries in the Receive FIFO. This value cannot be changed dynamically during operation. NOTE: the maximum size of the RxFIFO is 512 32-bit word entries.
15:0	0x0 RW	**Max Packet Size (max_pkt_size):** Maximum packet size for the endpoint. This is the value in bytes. This maximum size is used to calculate whether the Transmit FIFO has sufficient space to accept a packet. When changing the maximum packet size for a specific endpoint, the user must also program the corresponding Physical Endpoint Register.

16.6.1.50 OUT Endpoint 3 SETUP Buffer Pointer Register (ep3_subptr_udc_reg)—Offset 270h

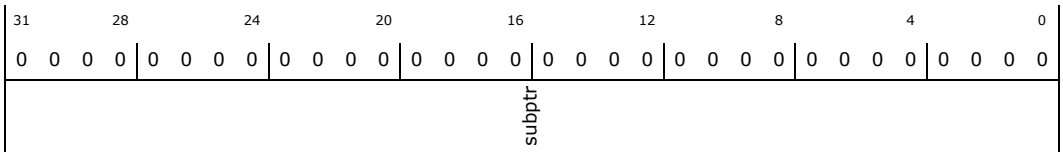
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 270h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**SETUP Buffer Pointer (subptr):** Endpoint SETUP buffer pointers are used for SETUP commands. The Endpoint SETUP Buffer Pointer register is used only in DMA mode. NOTE: This is applicable only to control endpoints. For all other endpoints this register is reserved.

16.6.1.51 OUT Endpoint 3 Data Descriptor Pointer Register (ep3_out_desptr_udc_reg)—Offset 274h

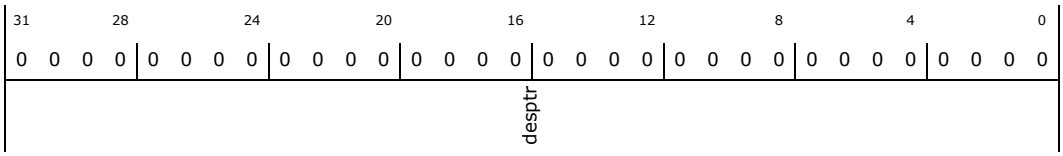
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 274h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Descriptor Pointer (desptr):** This register contains the data descriptor pointer.

16.6.1.52 OUT Endpoint 3 Read Confirmation Register for zero-length OUT data (for Slave-Only mode) (ep3_rd_cfrm_udc_reg)—Offset 27Ch

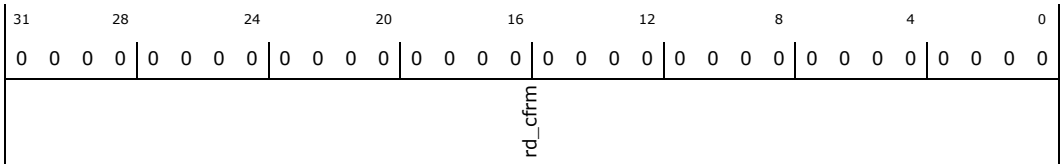
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 27Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 WO	**Read Confirmation for zero-length OUT data (for Slave-Only mode) (rd_cfrm):** For zero-length OUT data, the application must perform a dummy read from this register

16.6.1.53 Device Configuration Register (d_cfg_udc_reg)—Offset 400h

This register configures the device. It is only set during initial configuration or when there is a change in the configuration.

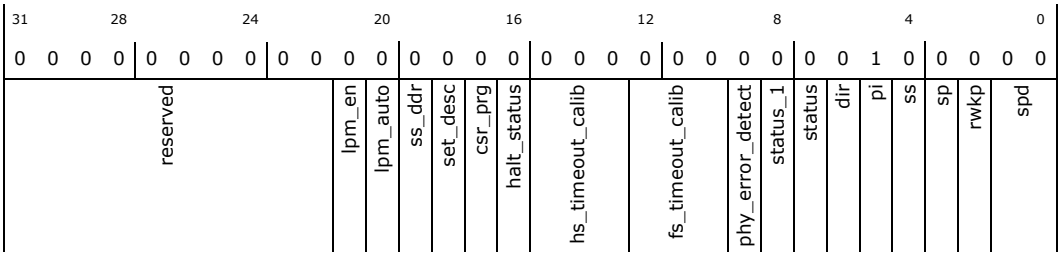
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 400h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000020h



Bit Range	Default & Access	Field Name (ID): Description
31:22	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
21	0x0 RO	**Link Power Mode Enable (lpm_en):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
20	0x0 RO	**Link Power Mode Automatic (lpm_auto):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
19	0x0 RO	**Double Data Rate (ss_ddr):** ULPI PHY interface is not supported. These bits are reserved and should be set to zero.
18	0x0 RW	**Set Descriptor Request Enable (set_desc):** Indicates that the device supports Set Descriptor requests. 0: The USB Device Controller returns a STALL handshake to the USB host. 1: The SETUP packet for the Set Descriptor request passes to the application.
17	0x0 RW	**Dynamic Register Programming (csr_prg):** The application can program the USB Device Controller registers dynamically whenever it has received an interrupt for either a Set Configuration or a Set Interface request. If this bit is set to 1, the USB Device Controller returns a NAK handshake during the status IN stage of both the Set Configuration and Set Interface requests until the application has written 1 to the CSR_DONE bit 13 of the Device Control Register.
16	0x0 RW	**Halt Status (halt_status):** This bit indicates whether the USB Device Controller must respond with a STALL or an ACK handshake when the USB host has issued a Clear_Feature (ENDPOINT_HALT) request for Endpoint 0. Options are: 0: ACK 1: STALL
15:13	0x0 RW	**HS Timeout Counter (hs_timeout_calib):** Number of PHY Clocks to the USB Device Controller Timeout Counter(for HS).The application uses these bits to increase the timeout value (736 to 848 bit times in high-speed operation), which depends on the PHY delay in generating a line state condition. The default timeout value is 736 bit times.

Bit Range	Default & Access	Field Name (ID): Description
12:10	0x0 RW	**FS Timeout Counter (fs_timeout_calib):** Number of PHY Clocks to the USB Device Controller Timeout Counter(for FS).These three bits indicate the number of PHY clocks to the USB Device Controller timeout counter. The application uses these bits to increase the timeout value (16 to 18 bit times in full-speed operation), which depends on the PHY delay in generating line state condition. The default timeout value is 16 bit times.
9	0x0 RW	**PHY Error Detect (phy_error_detect):** If the application sets this bit, the device detects the phy_rxvalid or phy_rxactive input signal to be continuously asserted for 2 ms, indicating PHY error.
8	0x0 RW	**Non-zero Control Handshake 1 (status_1):** This bit, together with STATUS Bit 7, provides an option for the USB Device Controller to respond to the USB host with a STALL or ACK handshake if the USB host has issued a non-zero-length data packet during the STATUS-OUT stage of a CONTROL transfer.
7	0x0 RW	**Non-zero Control Handshake (status):** This bit, together with STATUS Bit 8, provides an option for the USB Device Controller to respond to the USB host with a STALL or ACK handshake if the USB host has issued a non-zero-length data packet during the STATUS-OUT stage of a CONTROL transfer.
6	0x0 RO	**UTMI Data Bus Direction (dir):** This bit indicates if the UTMI data bus interface has to support a unidirectional or bidirectional interface. 0: Unidirectional interface 1: Bidirectional interface
5	1h RW	**PHY Interface Width (pi):** Indicates if the UTMI PHY supports an 8-bit or 16-bit interface. 0: 16-bit 1: 8-bit. NOTE: even if this field is writable, only 8bit interface is supported. Writing 0 will lead to undefined behavior.
4	0x0 RW	**Sync Frame Support (ss):** Indicates that the Device Supports Sync Frame.
3	0x0 RW	**Self-Powered Device (sp):** Indicates that the Device is Self-Powered.
2	0x0 RW	**Remote Wake Up Capable (rwkp):** Indicates that the device is remote wake up capable.
1:0	0x0 RW	**Device Speed (spd):** This is the expected speed the application programs to the subsystem. The actual speed the subsystem operates depends on the enumeration speed (ENUM SPD) of the Device Status register. 00: HS (PHY clock = 60 MHz) 01: FS (PHY clock = 60 MHz) 10: Reserved 11: Reserved

16.6.1.54 Device Control Register (d_ctrl_udc_reg)—Offset 404h

This register is set at runtime and controls the device after device configuration.

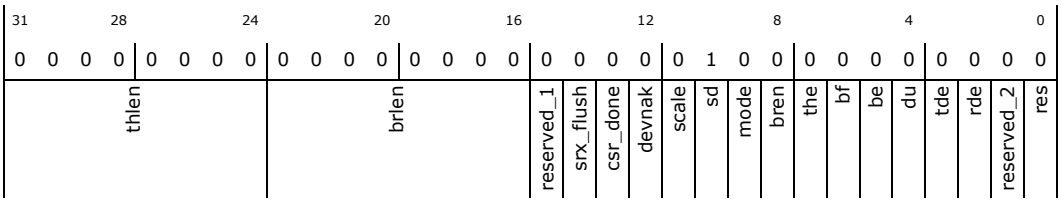
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 404h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000400h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0x0 RW	**Threshold Length (thlen):** Indicates the number (THLEN + 1) of 32-bit entries in the RxFIFO before the DMA can start data transfer.
23:16	0x0 RW	**Burst Length (brlen):** Indicates the length, in 32-bit transfers, of a single burst on the AHB. The subsystem sends number of 32-bit transfers equal to (BRLEN + 1).
15	0x0 RO	**reserved_1:** Reserved bits. These bits are reserved and should be set to zero.
14	0x0 RO	**Receive FIFO Flush for Single Receive FIFO (srx_flush):** Multiple receive FIFOs are implemented. These bits are reserved and should be set to zero.
13	0x0 WO	**Dynamic Register Programming Done (csr_done):** The application uses this bit to notify the USB Device Controller that the application has completed programming all required USB Device Controller registers, and the Subsystem can acknowledge the current Set Configuration or Set Interface command.
12	0x0 RW	**Device NAK (devnak):** When the application sets this bit, the Subsystem core returns a NAK handshake to all OUT endpoints. By writing 1 to this bit, the application does not need to write 1 to the SNAK bit 7 of each Endpoint Control register.
11	0x0 RW	**Scale Down (scale):** This bit reduces the timer values inside the USB Device Controller when running gate-level simulation only. When this bit is set to 1, timer values are scaled down to reduce simulation time. In Scale-Down mode, the USB Device Controller detects a USB reset within 150 PHY clock cycles (60-MHz PHY clock, 8-bit UTMI). Reset this bit to 0 for normal operation.
10	1h RW	**Soft Disconnect (sd):** The application software uses this bit to signal the USB Device Controller to soft-disconnect. When set to 1, this bit causes the device to enter the disconnected state.
9	0x0 RW	**DMA/Slave-Only Mode (mode):** Enables the application to dictate the subsystem operation in either DMA mode (1) or Slave-Only mode (0) operation.
8	0x0 RW	**Burst Enable (bren):** When this bit is set, transfers on the AHB are split into bursts.
7	0x0 RW	**Threshold Enable (the):** When this bit is set, a number of quadlets equivalent to the threshold value is transferred from the RxFIFO to the memory.
6	0x0 RW	**Buffer Fill Mode (bf):** The DMA is in Buffer Fill mode and transfers data into contiguous locations pointed to by the buffer address.
5	0x0 RW	**System Endianness (be):** A value of 1 indicates a big endian system.
4	0x0 RW	**Descriptor Update (du):** When this bit is set, the DMA updates the descriptor at the end of each packet processed.
3	0x0 RW	**Transmit DMA Enable (tde):** 0: disabled 1: enabled
2	0x0 RW	**Receive DMA Enable (rde):** 0: disabled 1: enabled
1	0x0 RO	**reserved_2:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
0	0x0 RW	**Remote Wakeup Resume (res):** To perform a remote wakeup resume the application sets this bit to 1, then resets it to 0 after 1 ms. The USB Device Controller signals the USB host to resume the USB bus. However: The application must first set RWKP bit 2 in the Device Configuration Register, indicating that the subsystem supports the Remote Wakeup feature. The host must already have issued a Set Feature request to enable the device Remote Wakeup feature.

16.6.1.55 Device Status Register (d_sts_udc_reg)—Offset 408h

This register reflects status information needed to service some of the interrupts.

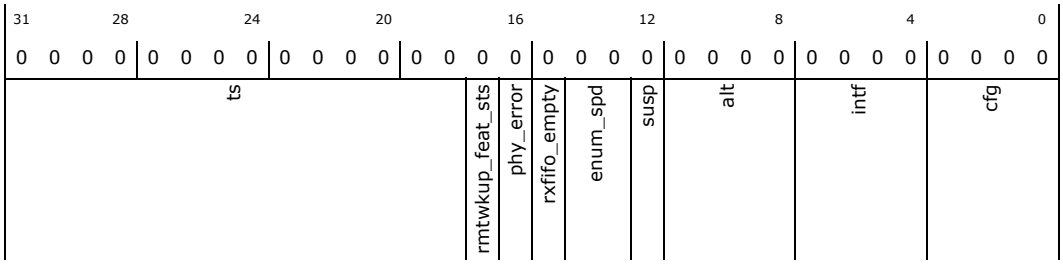
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 408h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0x0 RO	**Received SOF Frame Number (ts):** For high-speed operation: [31:21]: Millisecond frame number [20:18]: Microframe number For full-speed operation: [31:29]: Reserved [28:18]: Millisecond frame number
17	0x0 RO	**Remote Wakeup Status (rmtwkup_feat_sts):** Status of Remote wakeup feature due to Set/Clear Feature (Remotewakeup) command from the host. A value of 1 indicates a Set Feature (Remotewakeup) has been received. A value of 0 indicates Clear Feature (Remotewakeup) has been received. Any change to this bit sets an interrupt in bit 7 of Device Interrupt register, if not masked.
16	0x0 RO	**PHY Error (phy_error):** Either the phy_rxvalid or phy_rxactive input signal is detected to be continuously asserted for 2 ms, indicating PHY error. The USB Device Controller goes to the Suspend state as a result. When the application serves the early suspend interrupt (ES bit 2 of the Device Interrupt register) it also must check this bit to determine if the early suspend interrupt was generated due to PHY error detection.
15	0h RO	**Receive Address FIFO Empty Status (rxfifo_empty):** Multiple receive FIFOs are implemented. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
14:13	0x0 RO	**Enumerated Speed (enum_spd):** These bits hold the speed at which the subsystem comes up after the speed enumeration. Possible options are: 00 HS: If the SPD is high speed and the subsystem connects to a 2.0 host controller, then after Speed Enumeration, these bits indicate that the subsystem is operating in high speed mode. 01 FS: If the expected speed (SPD of the Device Configuration register) is high speed and the subsystem connects to a 1.1 host controller, then after Speed Enumeration, these bits indicate that the subsystem is operating in full speed mode. 10 : Reserved 110: Reserved
12	0x0 RO	**Suspend Status (susp):** This bit is set as long as a Suspend condition is detected on the USB.
11:8	0x0 RO	**Alternate Setting (alt):** This 4-bit field represents the alternate setting to which the above interface is switched.
7:4	0x0 RO	**SetInterface Command (intf):** This 4-bit field reflects the interface set by the SetInterface command.
3:0	0x0 RO	**SetConfiguration Command (cfg):** This 4-bit field reflects the configuration set by the SetConfiguration command.

16.6.1.56 Device Interrupt Register (d_intr_udc_reg)—Offset 40Ch

Device interrupts are set when there are system-level events. Interrupts are used by the application to make system-level decisions. After checking the register, the application must clear the interrupt by writing a 1?b1 to the correct bit.

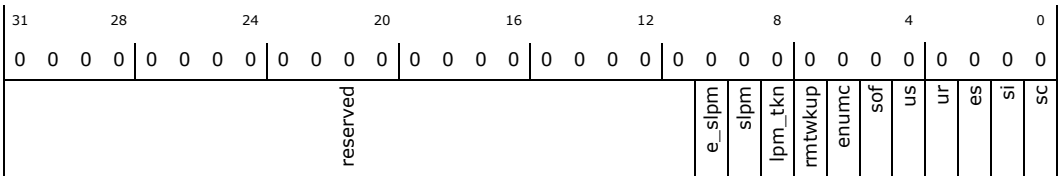
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 40Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
10	0x0 RO	**LPM Early Sleep (e_slpm):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
9	0x0 RO	**LPM Sleep (slpm):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
8	0x0 RO	**LPM Transaction (lpm_tkn):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
7	0x0 RW/1C	**Remote Wakeup (rmtwkup):** A Set/Clear Feature (Remote Wakeup) is received by the core. This bit is set by the core whenever bit 17 of the Device Status Register changes: HIGH to LOW or LOW to HIGH.

Bit Range	Default & Access	Field Name (ID): Description
6	0x0 RW/1C	**Speed Enumeration Completed (enumc):** Speed enumeration is complete.
5	0x0 RW/1C	**SOF Token Detected (sof):** An SOF token is detected on the USB.
4	0x0 RW/1C	**Suspend State (us):** A suspend state is detected on the USB for a duration of 3 milliseconds, following the 3 ms Idle State interrupt activity due to an idle state.
3	0x0 RW/1C	**USB Reset Detected (ur):** NOTE: If the application has not served this interrupt, the USB Device Controller returns a NAK handshake for all transactions except the 8 SETUP packet bytes from the USB host.
2	0x0 RW/1C	**Idle State Detected (es):** An idle state is detected on the USB for a duration of 3 ms. This interrupt bit is used for the application firmware to finish its job before the subsystem generates a true suspend (US) interrupt (another 3 ms after the ES interrupt)
1	0x0 RW/1C	**Set_Interface Command Received (si):** The device has received a Set_Interface command. NOTE: If the application has not served this interrupt, the USB Device Controller returns a NAK handshake for all transactions except the 8 SETUP packet bytes from the USB host.
0	0x0 RW/1C	**Set_Configuration Command Received (sc):** NOTE: If the application has not served this interrupt, the USB Device Controller returns a NAK handshake for all transactions except the 8 SETUP packet bytes from the USB host.

16.6.1.57 Device Interrupt Mask Register (d_intr_msk_udc_reg)—Offset 410h

The device interrupt mask can be set for system-level interrupts using this register. Programming 1'b1 in the appropriate bit position in the Interrupt Mask register masks the designated interrupt. Once masked, an interrupt signal does not reach the application, nor does its interrupt bit get set.

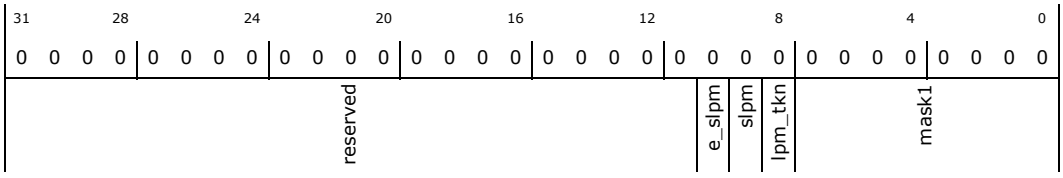
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 410h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
10	0x0 RO	**LPM Early Sleep (e_slpm):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
9	0x0 RO	**LPM Sleep (slpm):** Link Power Mode is not supported. These bits are reserved and should be set to zero.
8	0x0 RO	**LPM Transaction (lpm_tkn):** Link Power Mode is not supported. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
7:0	0x0 RW	**Device Interrupt Mask (mask1):** Masks equivalent device interrupt bit in the Device Interrupt Register.

16.6.1.58 Endpoints Interrupt Register (ep_intr_udc_reg)—Offset 414h

The Endpoint Interrupt register is used to set endpoint-level interrupts. Since all 4 endpoints are bidirectional, each endpoint has two interrupt bits (one for each direction). The application needs to clear the interrupt by writing a 1'b1 to the correct bit after checking the register.

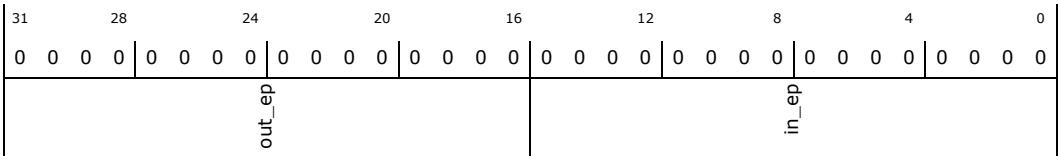
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 414h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RW/1C	**OUT Endpoint Interrupt (out_ep):** One bit is associated to one of the 4 supported OUT endpoint, set when there is an event on that endpoint. bit0: OUT ED0 bit1: OUT ED1 bit2: OUT ED2 bit3: OUT ED3 bit4-bit15: Reserved
15:0	0x0 RW/1C	**IN Endpoint Interrupt (in_ep):** One bit is associated to one of the 4 supported IN endpoint, set when there is an event on that endpoint. bit0: IN ED0 bit1: IN ED1 bit2: IN ED2 bit3: IN ED3 bit4-bit15: Reserved

16.6.1.59 Endpoints Interrupt Mask Register (ep_intr_msk_udc_reg)—Offset 418h

This register is used to mask endpoint interrupts. A write of 1'b1 to any bit in this register masks the corresponding endpoint for any possible interrupts. Once masked, an interrupt signal does not reach the application, nor does its interrupt bit get set.

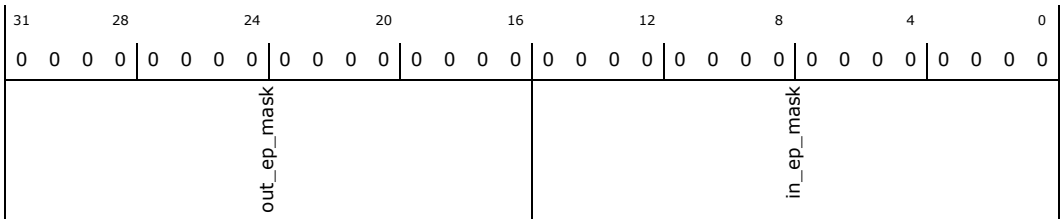
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 418h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RW	**OUT Endpoint Interrupt Mask (out_ep_mask):** Masks the OUT Endpoint Interrupt Register bits of the equivalent OUT endpoint. bit0: OUT ED0 bit1: OUT ED1 bit2: OUT ED2 bit3: OUT ED3 bit4-bit15: Reserved
15:0	0h RW	**IN Endpoint Interrupt Mask (in_ep_mask):** Masks the IN Endpoint Interrupt Register bits of the equivalent IN endpoint. bit0: IN ED0 bit1: IN ED1 bit2: IN ED2 bit3: IN ED3 bit4-bit15: Reserved

16.6.1.60 Test Mode Register (test_mode_udc_reg)—Offset 41Ch

In Test mode, the application can use the AHB to read from a TxFIFO or write to an RxFIFO using AHB read/write cycles. Test mode is supported only in the Slave-Only operational mode. In Test mode, only single- DWORD transactions are supported: byte and word transactions on the AHB are not supported. In non-Test modes, reading from a TxFIFO or writing to an RxFIFO results in an AHB error response. The application must never read an empty TxFIFO or write a full RxFIFO even though an AHB error response is not provided. NOTE: Writing to the RxFIFO in Test mode and enabling the DMA to transfer the data are not supported.

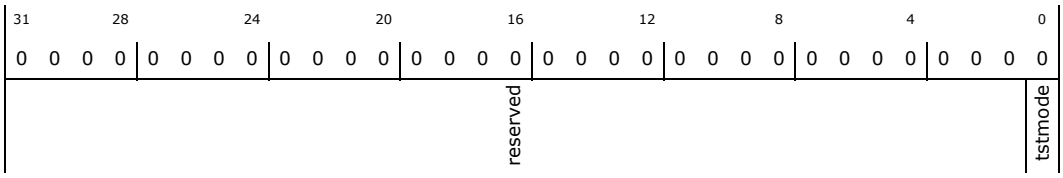
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 41Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0x0 RO	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
0	0x0 RW	**Test Mode indicator (tstmode):** 0: Normal mode 1: Test mode

16.6.1.61 Product Release Number Register (revision_udc_reg)—Offset 420h

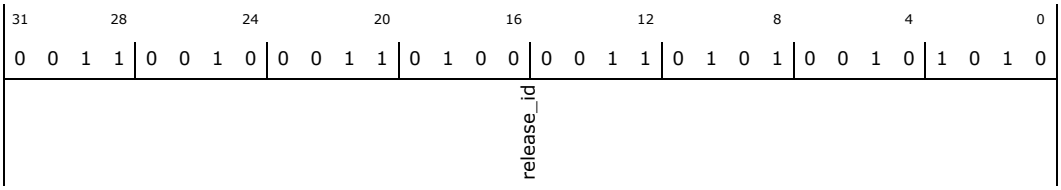
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 420h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 3234352Ah



Bit Range	Default & Access	Field Name (ID): Description
31:0	3234352Ah RO	**Product Release Number (release_id):** This field indicates the ASCII characters of the four-digit release number in hexadecimal format. For example, 32_34_35_ 2A represents 2.45* in ASCII character, where * is an alphabetic character (for example, a, b, or c) that represents a update to the release, which does not impact the RTL source.

16.6.1.62 SETUP command address pointer register (udc_desc_addr_udc_reg)—Offset 500h

NOTE: The SETUP command address pointer register is not writable and returns 0 when read because the SETUP command address pointer is hardcoded to xFFF0.

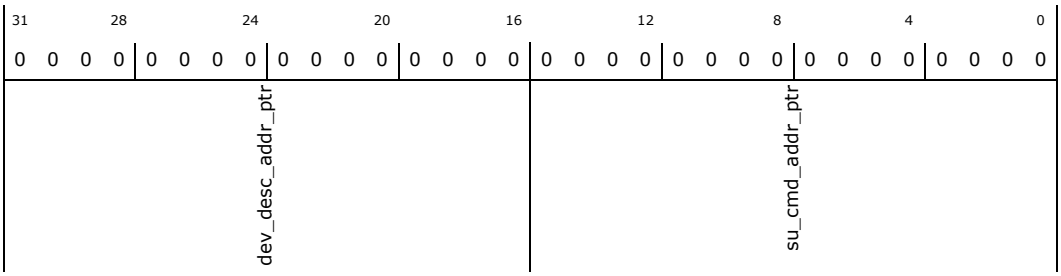
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 500h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0x0 RO	**Device Descriptor Address Pointer (dev_desc_addr_ptr):** Device Descriptor Address Pointer
15:0	0x0 RO	**SETUP Command Address Pointer (su_cmd_addr_ptr):** SETUP Command Address Pointer

16.6.1.63 Physical Endpoint 0 Register (udc_ep_ne_udc_reg_0)—Offset 504h

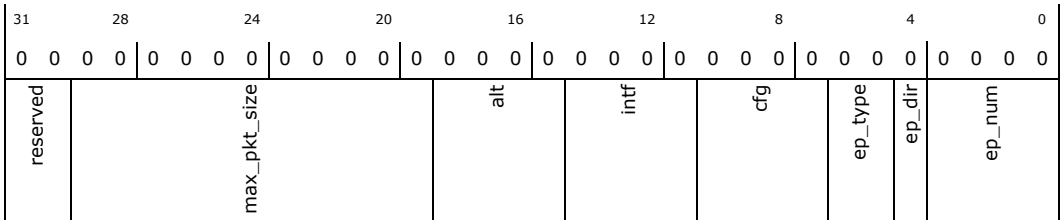
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 504h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.64 Physical Endpoint 1 Register (udc_ep_ne_udc_reg_1)—Offset 508h

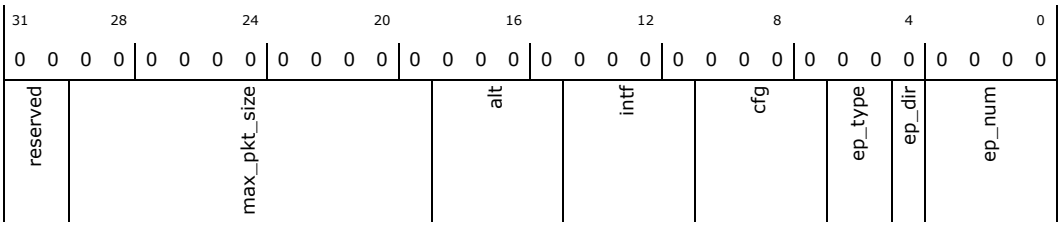
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 508h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.65 Physical Endpoint 2 Register (udc_ep_ne_udc_reg_2)—Offset 50Ch

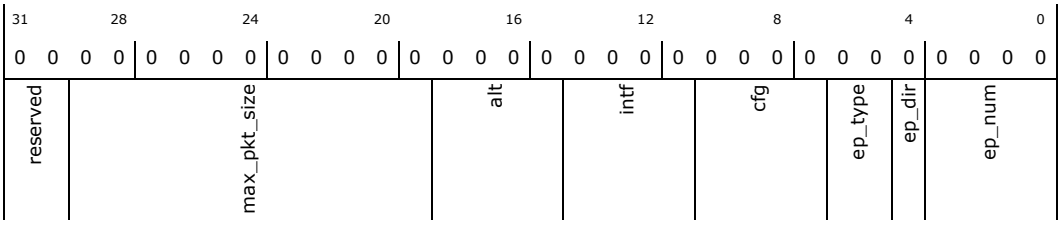
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 50Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.66 Physical Endpoint 3 Register (udc_ep_ne_udc_reg_3)—Offset 510h

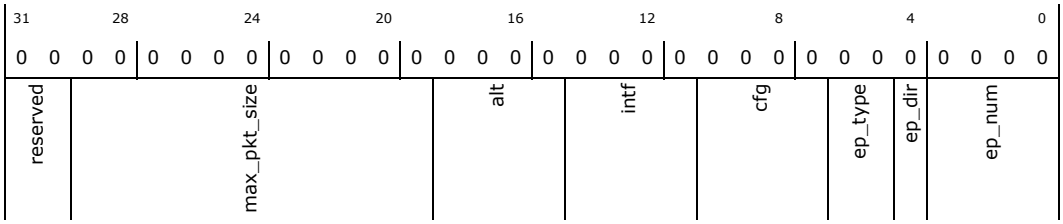
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 510h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs

Bit Range	Default & Access	Field Name (ID): Description
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.67 Physical Endpoint 4 Register (udc_ep_ne_udc_reg_4)—Offset 514h

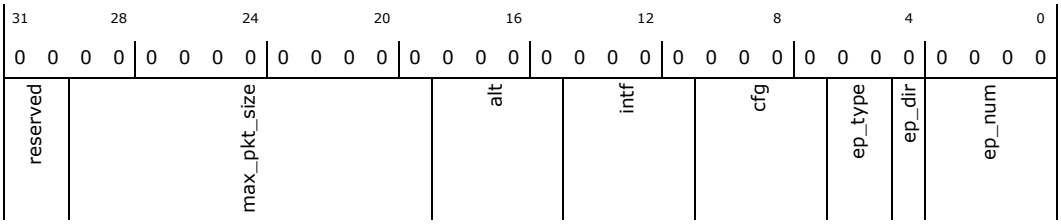
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 514h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.68 Physical Endpoint 5 Register (udc_ep_ne_udc_reg_5)—Offset 518h

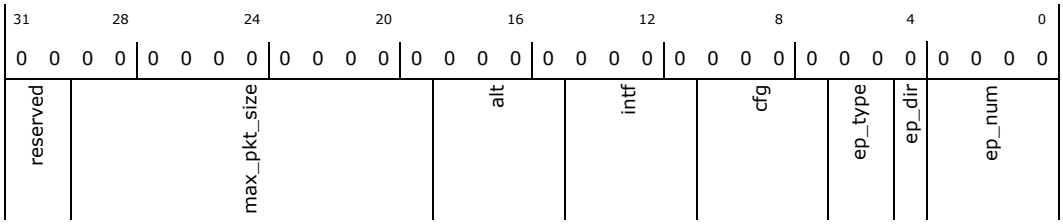
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 518h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.69 Physical Endpoint 6 Register (udc_ep_ne_udc_reg_6)—Offset 51Ch

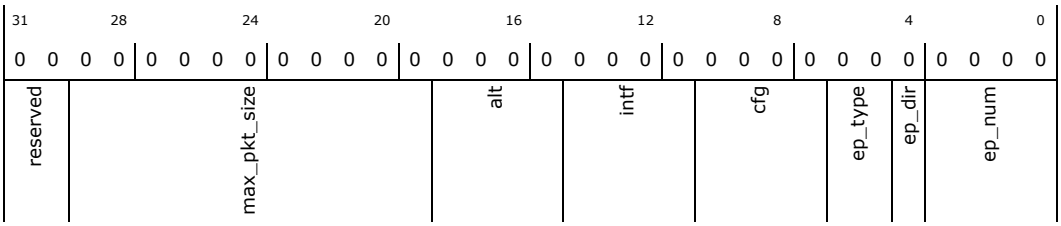
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 51Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0x0 RW	**reserved:** Reserved bits. These bits are reserved and should be set to zero.
29:19	0x0 RW	**Maximum Packet Size (max_pkt_size):** NOTE: In full-speed mode operation, application firmware must program the maximum packet size of default endpoint 0 in accordance with the value defined in USB 2.0 specification
18:15	0x0 RW	**Alternate Setting (alt):** Alternate setting to which this endpoint belongs.
14:11	0x0 RW	**Interface Number (intf):** Interface number to which this endpoint belongs.
10:7	0x0 RW	**Configuration Number (cfg):** Configuration number to which this endpoint belongs
6:5	0x0 RW	**Endpoint Type (ep_type):** The possible options are: 00: Control 01: Isochronous 10: Bulk 11: Interrupt
4	0x0 RW	**Endpoint Direction (ep_dir):** 0: OUT 1: IN
3:0	0x0 RW	**Logical Endpoint Number (ep_num):** Logical Endpoint Number

16.6.1.70 RxFIFO Array[0-511] (udc_rx_fifo_reg_array[0-511])—Offset 800h, Count 512, Stride 4h

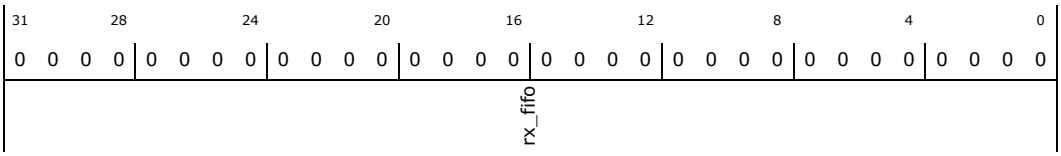
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-511]: [BAR0] + 800h + [0-511]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Receive FIFO (rx_fifo):** Receive FIFO

16.6.1.71 TxFIFO 0 Array[0-255] (udc_tx_fifo_reg_0_array[0-255])—Offset 1000h, Count 256, Stride 4h

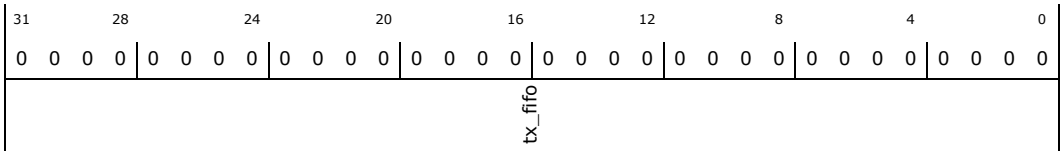
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-255]: [BAR0] + 1000h + [0-255]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Transmit FIFO 0 (tx_fifo):** Transmit FIFO 0

16.6.1.72 TxFIFO 1 Array[0-255] (udc_tx_fifo_reg_1_array[0-255])—Offset 1400h, Count 256, Stride 4h

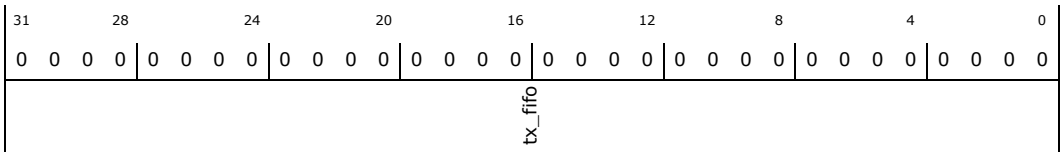
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-255]: [BAR0] + 1400h + [0-255]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Transmit FIFO 1 (tx_fifo):** Transmit FIFO 1

16.6.1.73 TxFIFO 2 Array[0-255] (udc_tx_fifo_reg_2_array[0-255])—Offset 1800h, Count 256, Stride 4h

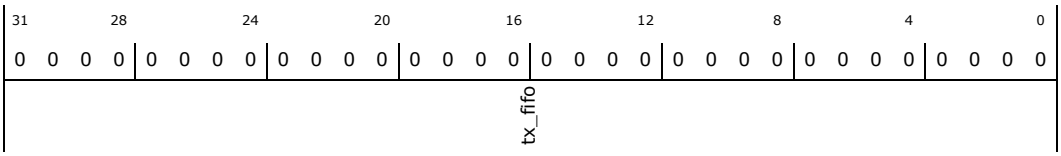
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-255]: [BAR0] + 1800h + [0-255]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Transmit FIFO 2 (tx_fifo):** Transmit FIFO 2

16.6.1.74 TxFIFO 3 Array[0-255] (udc_tx_fifo_reg_3_array[0-255])—Offset 1C00h, Count 256, Stride 4h

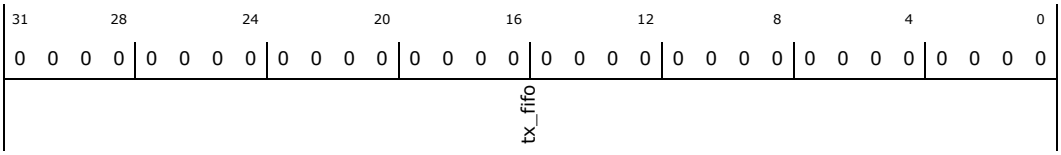
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-255]: [BAR0] + 1C00h + [0-255]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0x0 RW	**Transmit FIFO 3 (tx_fifo):** Transmit FIFO 3

16.6.2 USB EHCI

Table 90. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Host Controller Interface Version Number and Capability Registers Length (HCCAPBASE)—Offset 0h" on page 536	01000010h
4h	7h	"Host Controller Structural Parameters (HCSPARAMS)—Offset 4h" on page 536	00001212h
8h	Bh	"Host Controller Capability Parameters (HCCPARAMS)—Offset 8h" on page 538	0000C012h
10h	13h	"USB Command (USBCMD)—Offset 10h" on page 539	00080000h
14h	17h	"USB Status (USBSTS)—Offset 14h" on page 541	00001000h
18h	1Bh	"USB Interrupt Enable (USBINTR)—Offset 18h" on page 543	00000000h
1Ch	1Fh	"USB Frame Index (FRINDEX)—Offset 1Ch" on page 544	00000000h
20h	23h	"4 Gigabyte Memory Segment Selector (CTRLDSSEGMENT)—Offset 20h" on page 545	00000000h
24h	27h	"Periodic Frame List Base Address (PERIODICLISTBASE)—Offset 24h" on page 545	00000000h

Table 90. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
28h	2Bh	"Asynchronous List Address (ASYNCLISTADDR)—Offset 28h" on page 546	00000000h
50h	53h	"Configure Flag (CONFIGFLAG)—Offset 50h" on page 546	00000000h
54h + [0-1]*4h	57h	"Port Status/Control[0-1] (PORTSC[0-1])—Offset 54h, Count 2, Stride 4h" on page 547	00002000h
90h	93h	"Programmable Microframe Base Value (INSNREG00)—Offset 90h" on page 550	00000000h
94h	97h	"Programmable Packet Buffer OUT/IN Thresholds (INSNREG01)—Offset 94h" on page 551	00200020h
98h	9Bh	"Programmable Packet Buffer Depth (INSNREG02)—Offset 98h" on page 551	00000080h
9Ch	9Fh	"Programmable Controller Settings (INSNREG03)—Offset 9Ch" on page 552	00002001h
A0h	A3h	"Programmable Controller Settings (INSNREG04)—Offset A0h" on page 553	00000000h
A4h	A7h	"UTMI Configuration (INSNREG05)—Offset A4h" on page 554	00001000h

16.6.2.1 Host Controller Interface Version Number and Capability Registers Length (HCCAPBASE)—Offset 0h

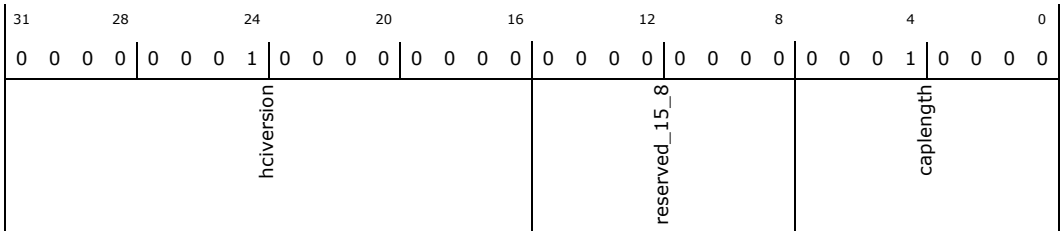
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 01000010h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0100h RO	**Host Controller Interface Version Number (hciversion):** This is a two-byte register containing a BCD encoding of the EHCI revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision.
15:8	0h RO	**reserved_15_8:** Reserved bits. These bits are reserved and should be set to zero.
7:0	010h RO	**Capability Registers Length (caplength):** This register is used as an offset to add to register base to find the beginning of the Operational Register Space.

16.6.2.2 Host Controller Structural Parameters (HCSPARAMS)—Offset 4h

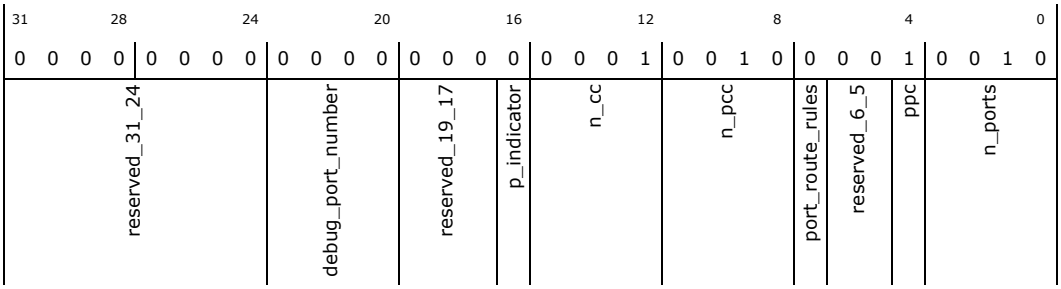
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00001212h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	**reserved_31_24:** Reserved bits. These bits are reserved and should be set to zero.
23:20	0h RO	**Debug Port Number (debug_port_number):** This register identifies which of the host controller ports is the debug port. The value is the port number (one-based) of the debug port. A non-zero value in this field indicates the presence of a debug port. The value in this register must not be greater than N_PORTS. NOTE: Debug Port is not supported
19:17	000b RO	**reserved_19_17:** Reserved bits. These bits are reserved and should be set to zero.
16	0b RO	**Port Indicator (p_indicator):** This bit indicates whether the ports support port indicator control. When this bit is a one, the port status and control registers include a read/writeable field for controlling the state of the port indicator. NOTE: Port Indicator is not supported
15:12	1h RO	**Number of Companion Controllers (n_cc):** This field indicates the number of companion controllers associated with this USB 2.0 host controller. A zero in this field indicates there are no companion host controllers. Port-ownership hand-off is not supported. Only high-speed devices are supported on the host controller root ports. A value larger than zero in this field indicates there are companion USB 1.1 host controller(s). Port-ownership hand-offs are supported. High, Full- and Low-speed devices are supported on the host controller root ports.
11:8	2h RO	**Number of Ports per Companion Controller (n_pcc):** This field indicates the number of ports supported per companion host controller. It is used to indicate the port routing configuration to system software. For example, if N_PORTS has a value of 6 and N_CC has a value of 2 then N_PCC could have a value of 3. The convention is that the first N_PCC ports are assumed to be routed to companion controller 1, the next N_PCC ports to companion controller 2, etc. In the previous example, the N_PCC could have been 4, where the first 4 are routed to companion controller 1 and the last two are routed to companion controller 2. The number in this field must be consistent with N_PORTS and N_CC.
7	0b RO	**Port Routing Rules (port_route_rules):** This field indicates the method used by this implementation for how all ports are mapped to companion controllers. The value of this field has the following interpretation: 0: The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on. 1: The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTROUTE array.
6:5	00b RO	**reserved_6_5:** Reserved bits. These bits are reserved and should be set to zero.
4	1b RO	**Port Power Control (ppc):** This field indicates whether the host controller implementation includes port power control. A one in this bit indicates the ports have port power switches. A zero in this bit indicates the port do not have port power switches. The value of this field affects the functionality of the Port Power field in each port status and control register. NOTE: Port Power Control is supported.

Bit Range	Default & Access	Field Name (ID): Description
3:0	2h RO	**Number of Physical Downstream Ports (n_ports):** This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 1 to F. A zero in this field is undefined.

16.6.2.3 Host Controller Capability Parameters (HCCPARAMS)—Offset 8h

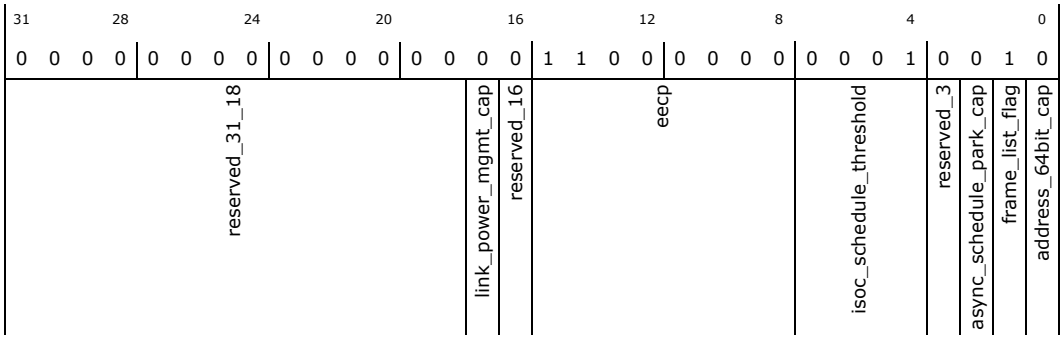
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 0000C012h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0h RO	**reserved_31_18:** Reserved bits. These bits are reserved and should be set to zero.
17	0h RO	**Link Power Management Capability (link_power_mgmt_cap):** NOTE: Link Power Management is not supported.
16	0h RO	**reserved_16:** Reserved bits. These bits are reserved and should be set to zero.
15:8	C0h RO	**EHCI Extended Capabilities Pointer (eecp):** This optional field indicates the existence of a capabilities list. A value of 00h indicates no extended capabilities are implemented. A non-zero value in this register indicates the offset in PCI configuration space of the first EHCI extended capability. The pointer value must be 40 or greater if implemented to maintain the consistency of the PCI header defined for this class of device. NOTE: EHCI Extended Capabilities is supported.
7:4	1h RO	**Isochronous Scheduling Threshold (isoc_schedule_threshold):** This field indicates, relative to the current position of the executing host controller, where software can reliably update the isochronous schedule. When bit [7] is zero, the value of the least significant 3 bits indicates the number of micro-frames a host controller can hold a set of isochronous data structures (one or more) before flushing the state. When bit [7] is a one, then host software assumes the host controller may cache an isochronous data structure for an entire frame.
3	0h RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RO	**Async Schedule Park Capability (async_schedule_park_cap):** If this bit is set to a one, then the host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule and he feature can be disabled or enabled and set to a specific level by using the Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count fields in the USBCMD register. NOTE: Async Schedule Park capability is not supported.
1	1h RO	**Programmable Frame List Flag (frame_list_flag):** If this bit is set to a zero, then system software must use a frame list length of 1024 elements with this host controller and the USBCMD register Frame List Size field is a read-only register. If set to a one, then system software can specify and use a smaller frame list and configure the host controller via the USBCMD register Frame List Size field. The frame list must always be aligned on a 4K page boundary. This requirement ensures that the frame list is always physically contiguous. NOTE: Programmable Frame List Flag is supported.
0	0h RO	**64-bit Addressing Capability (address_64bit_cap):** This field documents the addressing range capability of this implementation. The value of this field determines whether software should use the data structures. Values for this field have the following interpretation: 0: data structures using 32-bit address memory pointers 1: data structures using 64-bit address memory pointers NOTE: 64-bit Addressing is not supported

16.6.2.4 USB Command (USBCMD)—Offset 10h

This register indicates the command to be executed by the serial bus host controller. Writing to the register causes a command to be executed.

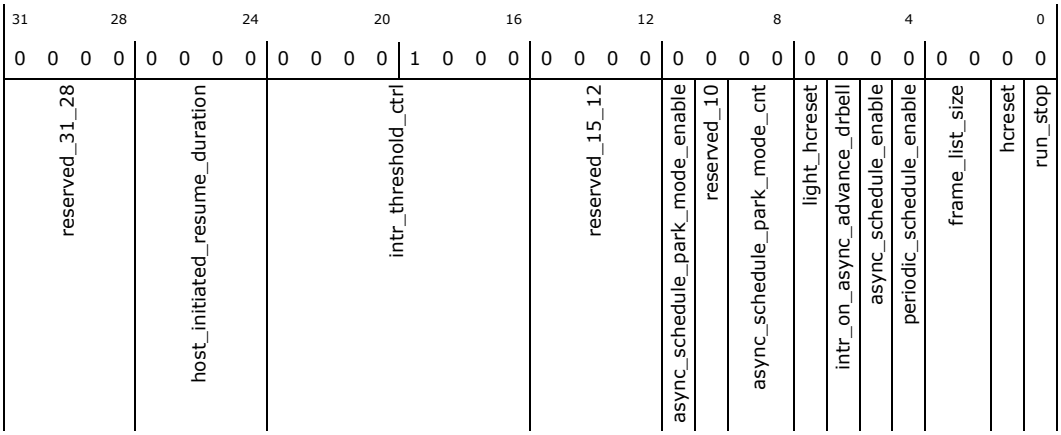
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00080000h



Bit Range	Default & Access	Field Name (ID): Description
31:28	0h RO	**reserved_31_28:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
27:24	0h RO	**Host Initiated Resume Duration (host_initiated_resume_duration):** If the Link Power Management Capability bit in the HCCPARAMS register is set to one then this bit is R/W; otherwise RO and not functional. NOTE: Link Power Management is not supported.
23:16	08h RW	**Interrupt Threshold Control (intr_threshold_ctrl):** This field is used by system software to select the maximum rate at which the host controller will issue interrupts. If software writes an invalid value to this register, the results are undefined. Allowed values are: 00: Reserved 01: 1 micro-frame 02: 2 micro-frames 04: 4 micro-frames 08: 8 micro-frames (default, equates to 1ms) 10: 16 micro-frames (2ms) 20: 32 micro-frames (4ms) 40: 64 micro-frames (8ms) Any other value in this register yields undefined results. Software modifications to this bit while HCHalted bit is equal to zero results in undefined behavior.
15:12	0h RO	**reserved_15_12:** Reserved bits. These bits are reserved and should be set to zero.
11	0h RO	**Asynchronous Schedule Park Mode Enable (async_schedule_park_mode_enable):** If the Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this bit defaults to a 1 and is R/W. Otherwise the bit must be a zero and is RO. Software uses this bit to enable or disable Park mode. When this bit is one, Park mode is enabled. When this bit is a zero, Park mode is disabled. NOTE: Asynchronous Park Capability is not supported.
10	0h RO	**reserved_10:** Reserved bits. These bits are reserved and should be set to zero.
9:8	0h RO	**Asynchronous Schedule Park Mode Count (async_schedule_park_mode_cnt):** If the Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this field defaults to 3 and is R/W. Otherwise it defaults to zero and is RO. It contains a count of the number of successive transactions the host controller is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. Valid values are 1h to 3h. Software must not write a zero to this bit when Park Mode Enable is a one as this will result in undefined behavior. NOTE: Asynchronous Park Capability is not supported.
7	0h RW	**Light Host Controller Reset (light_hcreset):** This control bit allows the driver to reset the EHCI controller without affecting the state of the ports or the relationship to the companion host controllers. For example, the PORSTC registers should not be reset to their default values and the CF bit setting should not go to zero (retaining port ownership relationships).A host software read of this bit as zero indicates the Light Host Controller Reset has completed and it is safe for host software to re-initialize the host controller. A host software read of this bit as a one indicates the Light Host Controller Reset has not yet completed. If not implemented a read of this field will always return a zero. NOTE: this control bit is supported and resets the EHCI List Processor Master Controller Unit.
6	0h RW	**Interrupt on Async Advance Doorbell (intr_on_async_advance_drbell):** This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Software must write a 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS register. If the Interrupt on Async Advance Enable bit in the USBINTR register is a one then the host controller will assert an interrupt at the next interrupt threshold. The host controller sets this bit to a zero after it has set the Interrupt on Async Advance status bit in the USBSTS register to a one. Software should not write a one to this bit when the asynchronous schedule is disabled. Doing so will yield undefined results.
5	0h RW	**Asynchronous Schedule Enable (async_schedule_enable):** This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean: 0: Do not process the Asynchronous Schedule 1: Use the ASYNCLISTADDR register to access the Asynchronous Schedule

Bit Range	Default & Access	Field Name (ID): Description
4	0h RW	**Periodic Schedule Enable (periodic_schedule_enable):** This bit controls whether the host controller skips processing the Periodic Schedule. Values mean: 0: Do not process the Periodic Schedule 1: Use the PERIODICLISTBASE register to access the Periodic Schedule
3:2	0h RW	**Frame List Size (frame_list_size):** This field is RW only if Programmable Frame List Flag in the HCCPARAMS registers is set to a one. This field specifies the size of the frame list. The size the frame list controls which bits in the Frame Index Register should be used for the Frame List Current index. Values mean: 00: 1024 elements (4096) Default value 01: 512 elements (2048) 10: 256 elements (1024) for resource-constrained environments 11: Reserved
1	0h RW	**Host Controller Reset (hcreset):** This control bit is used by software to reset host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset. When software writes a one to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports. PCI Configuration registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s). This bit is set to zero by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a zero to this register. Software should not set this bit to a one when the HCHalted bit in the USBSTS register is a zero. Attempting to reset an actively running host controller will result in undefined behavior.
0	0h RW	**Run/Stop bit (run_stop):** 1: Run 0: Stop When set to a 1, the Host Controller proceeds with execution of the schedule. The Host Controller continues execution as long as this bit is set to a 1. When this bit is set to 0, the Host Controller completes the current and any actively pipelined transactions on the USB and then halts. The Host Controller must halt within 16 micro-frames after software clears the Run bit. The HC Halted bit in the status register indicates when the Host Controller has finished its pending pipelined transactions and has entered the stopped state. Software must not write a one to this field unless the host controller is in the Halted state (i.e. HCHalted in the USBSTS register is a one). Doing so will yield undefined results.

16.6.2.5 USB Status (USBSTS)—Offset 14h

This register indicates pending interrupts and various states of the Host Controller. The status resulting from a transaction on the serial bus is not indicated in this register.

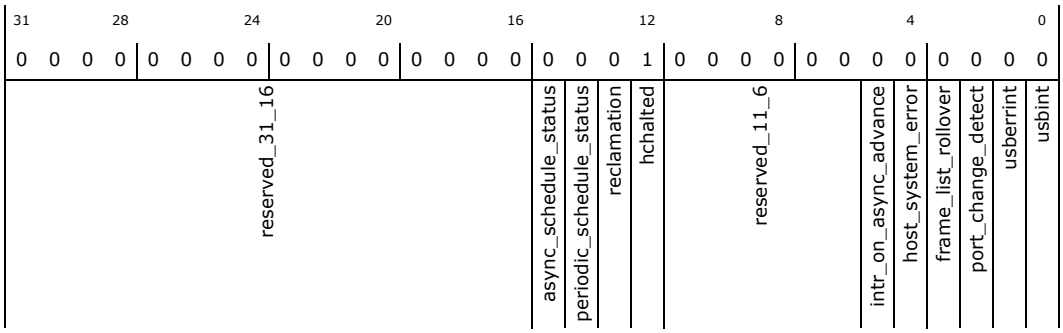
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00001000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	**reserved_31_16:** Reserved bits. These bits are reserved and should be set to zero.
15	0h RO	**Asynchronous Schedule Status (async_schedule_status):** The bit reports the current real status of the Asynchronous Schedule. If this bit is a zero then the status of the Asynchronous Schedule is disabled. If this bit is a one then the status of the Asynchronous Schedule is enabled. The Host Controller is not required to immediately disable or enable the Asynchronous Schedule when software transitions the Asynchronous Schedule Enable bit in the USBCMD register. When this bit and the Asynchronous Schedule Enable bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).
14	0h RO	**Periodic Schedule Status (periodic_schedule_status):** The bit reports the current real status of the Periodic Schedule. If this bit is a zero then the status of the Periodic Schedule is disabled. If this bit is a one then the status of the Periodic Schedule is enabled. The Host Controller is not required to immediately disable or enable the Periodic Schedule when software transitions the Periodic Schedule Enable bit in the USBCMD register. When this bit and the Periodic Schedule Enable bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).
13	0h RO	**Reclamation (reclamation):** This bit is used to detect an empty asynchronous schedule.
12	1h RO	**HcHalted (hchalted):** This bit is a zero whenever the Run/Stop bit is a one. The Host Controller sets this bit to one after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller hardware (e.g. internal error).
11:6	0h RO	**reserved_11_6:** Reserved bits. These bits are reserved and should be set to zero.
5	0h RW/1C	**Interrupt on Async Advance (intr_on_async_advance):** System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a one to the Interrupt on Async Advance Doorbell bit in the USBCMD register. This status bit indicates the assertion of that interrupt source.
4	0h RW/1C	**Host System Error (host_system_error):** The Host Controller sets this bit to 1 when a serious error occurs during a host system access involving the Host Controller module. In a PCI system, conditions that set this bit to 1 include PCI Parity error, PCI Master Abort, and PCI Target Abort. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs.
3	0h RW/1C	**Frame List Rollover (frame_list_rollover):** The Host Controller sets this bit to a one when the Frame List Index rolls over from its maximum value to zero. The exact value at which the rollover occurs depends on the frame list size. For example, if the frame list size (as programmed in the Frame List Size field of the USBCMD register) is 1024, the Frame Index Register rolls over every time FRINDEX[13] toggles. Similarly, if the size is 512, the Host Controller sets this bit to a one every time FRINDEX[12] toggles.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW/1C	**Port Change Detect (port_change_detect):** The Host Controller sets this bit to a one when any port for which the Port Owner bit is set to zero has a change bit transition from a zero to a one or a Force Port Resume bit transition from a zero to a one as a result of a J-K transition detected on a suspended port. This bit will also be set as a result of the Connect Status Change being set to a one after system software has relinquished ownership of a connected port by writing a one to a port's Port Owner bit. On a D3 to D0 transition of the EHCI HC device, this bit is loaded with the OR of all of the PORTSC change bits (including: Force port resume, over-current change, enable/ disable change and connect status change).
1	0h RW/1C	**USB Error Interrupt (usberrint):** The Host Controller sets this bit to 1 when completion of a USB transaction results in an error condition (e.g., error counter underflow). If the TD on which the error interrupt occurred also had its IOC bit set, both this bit and USBINT bit are set. Refer to EHCI Specification for a list of the USB errors that will result in this bit being set to a one.
0	0h RW/1C	**USB Interrupt (usbint):** The Host Controller sets this bit to 1 on the completion of a USB transaction, which results in the retirement of a Transfer Descriptor that had its IOC bit set. The Host Controller also sets this bit to 1 when a short packet is detected (actual number of bytes received was less than the expected number of bytes).

16.6.2.6 USB Interrupt Enable (USBINTR)—Offset 18h

This register enables and disables reporting of the corresponding interrupt to the software. When a bit is set and the corresponding interrupt is active, an interrupt is generated to the host. Interrupt sources that are disabled in this register still appear in the USBSTS to allow the software to poll for events.

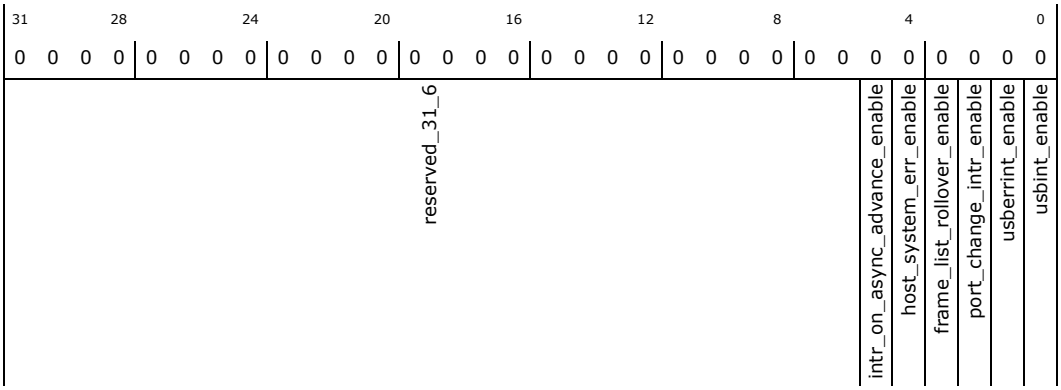
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:6	0h RO	**reserved_31_6:** Reserved bits. These bits are reserved and should be set to zero.
5	0h RW	**Interrupt on Async Advance Enable (intr_on_async_advance_enable):** When this bit is a one, and the Interrupt on Async Advance bit in the USBSTS register is a one, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.

Bit Range	Default & Access	Field Name (ID): Description
4	0h RW	**Host System Error Enable (host_system_err_enable):** When this bit is a one, and the Host System Error Status bit in the USBSTS register is a one, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.
3	0h RW	**Frame List Rollover Enable (frame_list_rollover_enable):** When this bit is a one, and the Frame List Rollover bit in the USBSTS register is a one, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit
2	0h RW	**Port Change Interrupt Enable (port_change_intr_enable):** When this bit is a one, and the Port Change Detect bit in the USBSTS register is a one, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Change Detect bit.
1	0h RW	**USB Error Interrupt Enable (usberrint_enable):** When this bit is a one, and the USBERRINT bit in the USBSTS register is a one, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBERRINT bit.
0	0h RW	**USB Interrupt Enable (usbint_enable):** When this bit is a one, and the USBINT bit in the USBSTS register is a one, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBINT bit.

16.6.2.7 USB Frame Index (FRINDEX)—Offset 1Ch

This register is used by the host controller to index into the periodic frame list.

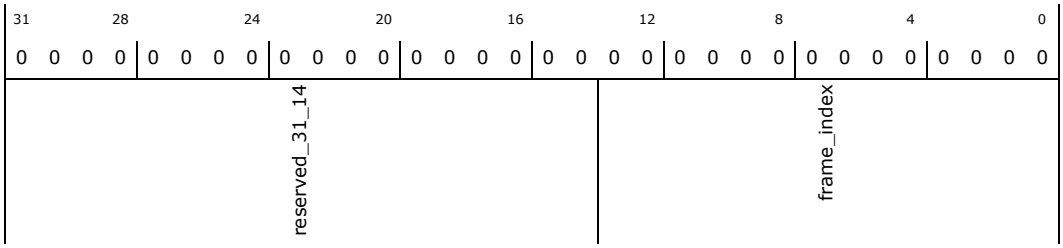
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	**reserved_31_14:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
13:0	0h RW	**Frame Index (frame_index):** The value of this register increments at the end of each time frame (e.g. micro-frame). Bits [N:3] are used for the Frame List current index. This means that each location of the frame list is accessed 8 times (frames or micro-frames) before moving to the next index. The following illustrates values of N based on the value of the Frame List Size field in the USBCMD register. USBCMD[Frame List Size] Number Elements N USBCMD[Frame List Size] = 00 (1024), N = 12 USBCMD[Frame List Size] = 01 (512), N = 11 USBCMD[Frame List Size] = 10 (256), N = 10 USBCMD[Frame List Size] = 11 (Reserved) This register must be written as a DWord. Byte writes produce undefined results. This register cannot be written unless the Host Controller is in the Halted state as indicated by the HCHalted bit in USBSTS register. A write to this register while the Run/Stop bit of USBCMD register is set to a one produces undefined results. Writes to this register also affect the SOF value.

16.6.2.8 4 Gigabyte Memory Segment Selector (CTRLDSSEGMENT)—Offset 20h

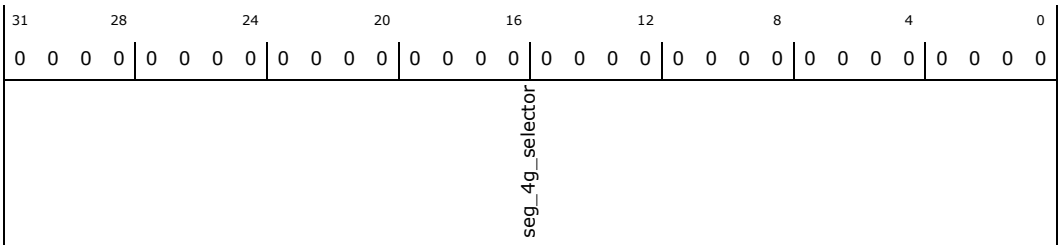
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO	**4 Gigabyte Memory Segment Selector (seg_4g_selector):** This 32-bit register corresponds to the most significant address bits [63:32] for all EHCI data structures. If the 64-bit Addressing Capability field in HCCPARAMS is a zero, then this register is not used. Software cannot write to it and a read from this register will return zeros. If the 64-bit Addressing Capability field in HCCPARAMS is a one, then this register is used with the link pointers to construct 64-bit addresses to EHCI control data structures. This register allows the host software to locate all control data structures within the same 4 Gigabyte memory segment. NOTE: 64-bit Addressing is not supported

16.6.2.9 Periodic Frame List Base Address (PERIODICLISTBASE)—Offset 24h

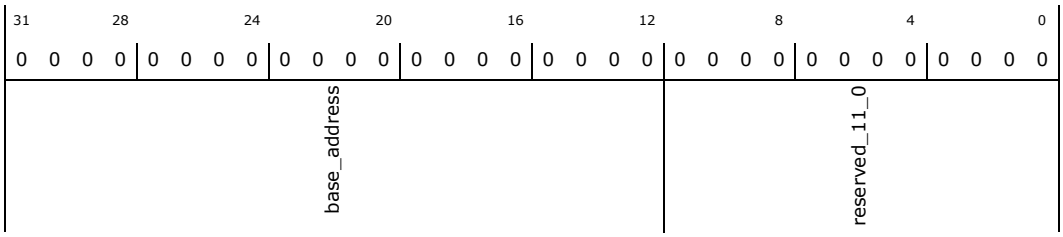
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 24h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RW	**Base Address (base_address):** This field contains bits [31:12]of the 32 bit address of the Periodic Frame List in the system memory. System software loads this register prior to starting the schedule execution by the Host Controller. The memory structure referenced by this physical memory pointer is assumed to be 4-Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence. Writes must be DWord Writes.
11:0	0h RO	**reserved_11_0:** Reserved bits. Must be written as 0s. During runtime the value of these bits is undefined.

16.6.2.10 Asynchronous List Address (ASYNCLISTADDR)—Offset 28h

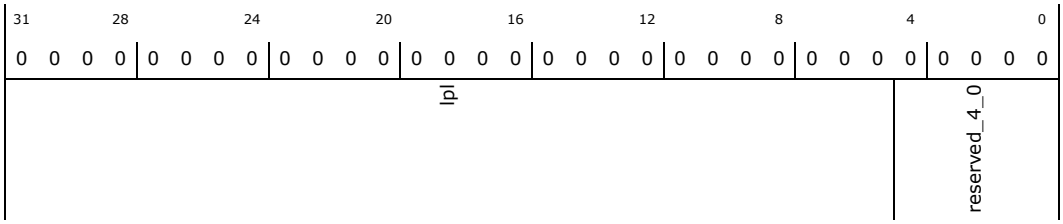
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 28h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RW	**Link Pointer Low (lpl):** This field contains bit [31:5] of the address of the next asynchronous queue head to be executed. The memory structure referenced by this physical memory pointer is assumed to be 4-Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence. This field may only reference a Queue Head (QH).
4:0	0h RO	**reserved_4_0:** Reserved bits. These bits are reserved and their value has no effect on operation.

16.6.2.11 Configure Flag (CONFIGFLAG)—Offset 50h

This register is in the auxiliary power well. It is only reset by hardware when the auxiliary power is initially applied or in response to a host controller reset.

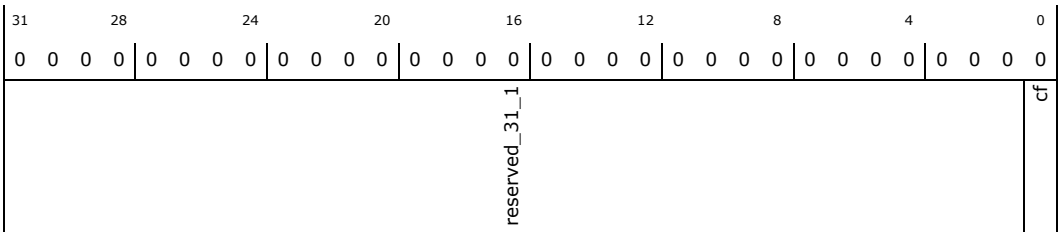
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 50h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0h RO	**reserved_31_1:** Reserved bits. These bits are reserved and should be set to zero.
0	0h RW	**Configure Flag (cf):** Host software sets this bit as the last action in its process of configuring the Host Controller. This bit controls the default port-routing control logic. Bit values and side-effects are listed below. 0: Port routing control logic default-routes each port to an implementation dependent classic host controller. 1: Port routing control logic default-routes all ports to this host controller

16.6.2.12 Port Status/Control[0-1] (PORTSC[0-1])—Offset 54h, Count 2, Stride 4h

This register is in the auxiliary power well. It is only reset by hardware when the auxiliary power is initially applied or in response to a host controller reset.

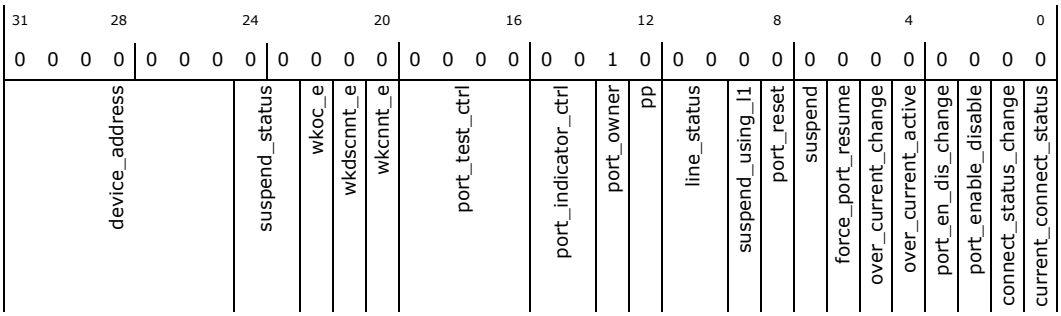
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset[0-1]: [BAR0] + 54h + [0-1]*4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00002000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0h RO	**Device Address (device_address):** The 7-bit USB device address for the device attached to and immediately downstream of the associated root port. A value of zero indicates no device is present or support for this feature is not present. NOTE: This field is not supported.

Bit Range	Default & Access	Field Name (ID): Description
24:23	0h RO	**Suspend Status (suspend_status):** These two bits are used by software to determine whether the most recent L1 suspend request was successful. NOTE: This field is not supported.
22	0h RW	**Wake on Over-current Enable (wkoc_e):** Writing this bit to a one enables the port to be sensitive to over-current conditions as wake-up events. This field is zero if Port Power is zero.
21	0h RW	**Wake on Disconnect Enable (wkdscnnt_e):** Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events. This field is zero if Port Power is zero.
20	0h RW	**Wake on Connect Enable (wkcnnt_e):** Writing this bit to a one enables the port to be sensitive to device connects as wake-up events. This field is zero if Port Power is zero.
19:16	0h RW	**Port Test Control (port_test_ctrl):** When this field is zero, the port is NOT operating in a test mode. A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value. The encoding of the test mode bits are (0110 - 1111 are reserved): 0000: Test mode not enabled 0001: Test J_STATE 0010: Test K_STATE 0011: Test SE0_NAK 0100: Test Packet 0101: Test FORCE_ENABLE
15:14	00h RW	**Port Indicator Control (port_indicator_ctrl):** Writing to these bits has no effect if the Port Indicator (P_INDICATOR) bit in the HCSPARAMS register is a zero. If P_INDICATOR bit is a one, then the bit encodings are: 00: Port indicators are off 01: Amber 10: Green 11: Undefined This field is zero if Port Power is zero. NOTE: Port Indicator is not supported.
13	1h RW	**Port Owner (port_owner):** This bit unconditionally goes to a 0 when the Configured bit in the CONFIGFLAG register makes a 0b to 1b transition. This bit unconditionally goes to 1 whenever the Configured bit is zero. System software uses this field to release ownership of the port to a selected host controller (in the event that the attached device is not a high-speed device). Software writes a one to this bit when the attached device is not a high-speed device. A one in this bit means that a companion host controller owns and controls the port.
12	0h RW	**Port Power (pp):** The function of this bit depends on the value of the Port Power Control (PPC) field in the HCSPARAMS register. The behavior is as follows: If PPC=0, PP=1 and RO: Host controller does not have port power control switches. Each port is hard-wired to power. If PPC=0, PP=1 or 0 and R/W: Host controller has port power control switches. This bit represents the current setting of the switch (0 = off, 1 = on). When power is not available on port (i.e. PP equals a 0),the port is non-functional and will not report attaches, detaches, etc. When an over-current condition is detected on a powered port and PPC is a one, the PP bit in each affected port may be transitioned by the host controller from a 1 to 0 (removing power from the port). NOTE: Per Port Power control is supported (PPC=1)
11:10	0h RO	**Line Status (line_status):** These bits reflect the current logical levels of the D+ (bit 11) and D- (bit 10) signal lines. These bits are used for detection of low-speed USB devices prior to the port reset and enable sequence. This field is valid only when the port enable bit is zero and the current connect status bit is set to a one. The encoding of the bits are: Bits[11:10] USB State Interpretation 00: USB State is SE0, Not Low-speed device, perform EHCI reset 10: USB State is J-state, Not Low-speed device, perform EHCI reset 01: USB State is K-state, Low-speed device, release ownership of port 11: USB State is Undefined, Not Low-speed device, perform EHCI reset The value of this field is undefined if Port Power is zero.
9	0h RO	**Suspend using l1 (suspend_using_l1):** NOTE: suspend using l1 is not supported as part of LPM.

Bit Range	Default & Access	Field Name (ID): Description
8	0h RW	**Port Reset (port_reset):** 1: Port is in Reset 0: Port is not in Reset When software writes a one to this bit (from a zero), the bus reset sequence as defined in the USB Specification Revision 2.0 is started. Software writes a zero to this bit to terminate the bus reset sequence. Software must keep this bit at a one long enough to ensure the reset sequence, as specified in the USB Specification Revision 2.0, completes. When software writes this bit to a one, it must also write a zero to the Port Enable bit. Note that when software writes a zero to this bit there may be a delay before the bit status changes to a zero. The bit status will not read as a zero until after the reset has completed. If the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (e.g. set the Port Enable bit to a one). A host controller must terminate the reset and stabilize the state of the port within 2ms of software transitioning this bit from a one to a zero. For example: if the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state within 2ms of software writing this bit to a zero. The HCHalted bit in the USBSTS register should be a zero before software attempts to use this bit. The host controller may hold Port Reset asserted to a one when the HCHalted bit is a one. This field is zero if Port Power is zero.
7	0h RW	**Suspend (suspend):** 1: Port in suspend state 0: Port not in suspend state Port Enabled Bit and Suspend bit of this register define the port states as follows: Bit [Port Enabled, Suspend]: Port State 0X: Disable 10: Enable 11: Suspend When in suspend state, downstream propagation of data is blocked on this port, except for port reset. The blocking occurs at the end of the current transaction, if a transaction was in progress when this bit was written to 1. In the suspend state, the port is sensitive to resume detection. Note that the bit status does not change until the port is suspended and that there may be a delay in suspending a port if there is a transaction currently in progress on the USB. A write of zero to this bit is ignored by the host controller. The host controller will unconditionally set this bit to a zero when: Software sets the Force Port Resume bit to a zero (from a one). Software sets the Port Reset bit to a one (from a zero). If host software sets this bit to a one when the port is not enabled (i.e. Port enabled bit is a zero) the results are undefined. This field is zero if Port Power is zero.
6	0h RW	**Force Port Resume (force_port_resume):** 1: Resume detected/driven on port 0: No resume (K-state) detected/driven on port This functionality defined for manipulating this bit depends on the value of the Suspend bit. For example, if the port is not suspended (Suspend and Enabled bits are a one) and software transitions this bit to a one, then the effects on the bus are undefined. Software sets this bit to a 1 to drive resume signaling. The Host Controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a one because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register is also set to a one. If software sets this bit to a one, the host controller must not set the Port Change Detect bit. Note that when the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB Specification Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this bit remains a one. Software must appropriately time the Resume and set this bit to a zero when the appropriate amount of time has elapsed. Writing a zero (from one) causes the port to return to high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a one until the port has switched to the high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to a zero. This field is zero if Port Power is zero.
5	0h RW/1C	**Over Current Change (over_current_change):** This bit gets set to a one when there is a change to Over-current Active. Software clears this bit by writing a one to this bit position.
4	0h RO	**Over Current Active (over_current_active):** 1: This port currently has an over-current condition 0: This port does not have an over-current condition This bit will automatically transition from a one to a zero when the over current condition is removed.

Bit Range	Default & Access	Field Name (ID): Description
3	0h RW/1C	**Port Enable/Disable Change (port_en_dis_change):** 1: Port enabled/disabled status has changed 0: No change For the root hub, this bit gets set to a one only when a port is disabled due to the appropriate conditions existing at the EOF2 point Software clears this bit by writing a 1 to it. This field is zero if Port Power is zero.
2	0h RW	**Port Enable/Disable (port_enable_disable):** 1: Enable 0: Disable Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a one to this field. The host controller will only set this bit to a one when the reset sequence determines that the attached device is a high-speed device. Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by host software. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. When the port is disabled (0) downstream propagation of data is blocked on this port, except for reset. This field is zero if Port Power is zero.
1	0h RW/1C	**Connect Status Change (connect_status_change):** 1: Change in Current Connect Status 0: No change Indicates a change has occurred in the ports Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. For example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be setting an already set bit. Software sets this bit to 0 by writing a 1 to it. This field is zero if Port Power is zero.
0	0h RO	**Current Connect Status (current_connect_status):** 1: Device is present on port 0: No device is present This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change bit (Bit 1) to be set. This field is zero if Port Power is zero.

16.6.2.13 Programmable Microframe Base Value (INSNREG00)—Offset 90h

This register allows changing the microframe length value (default is microframe SOF = 125ms) to reduce the simulation time.

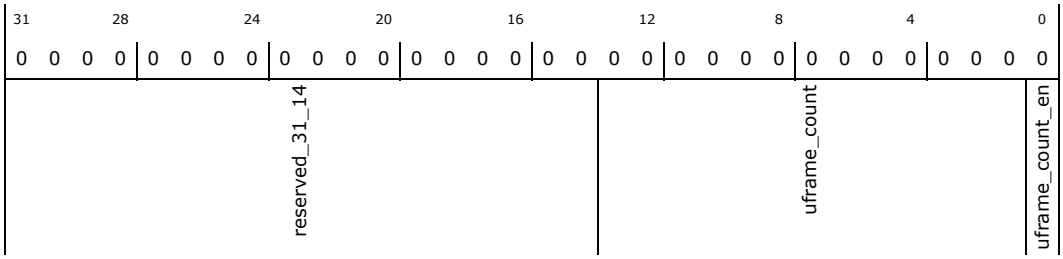
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 90h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	**reserved_31_14:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
13:1	0h RW	**uframe_count:** 1-microframe counter with byte interface (8-bits).
0	0b RW	**uframe_count_en:** 1-microframe counter is enabled when this bit is set to 1.

16.6.2.14 Programmable Packet Buffer OUT/IN Thresholds (INSNREG01)—Offset 94h

This register allows setting the packet buffer OUT/IN thresholds. The value specified by the thresholds is in number of DWORDs (32-bit entries). The minimum threshold amount that can be programmed is the highest possible INCRX burst value (INCR16) i.e the minimum OUT and IN threshold value should be 64 bytes (16 DWords). With the implemented packet buffer depth of 512 bytes, OUT and IN threshold values can be equal to the packet buffer depth only when isochronous/interrupt transactions are not initiated by the host controller.

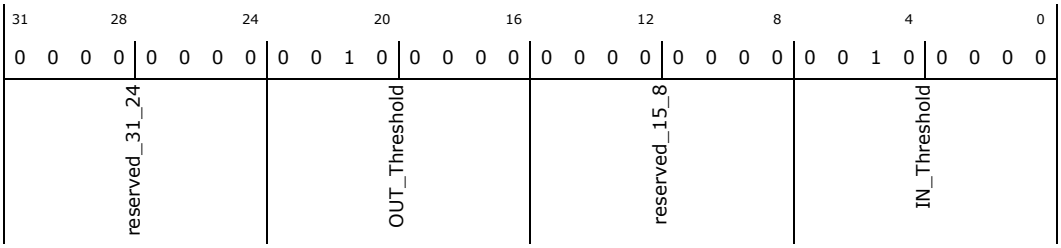
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 94h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00200020h



Bit Range	Default & Access	Field Name (ID): Description
31:24	00h RO	**reserved_31_24:** Reserved bits. These bits are reserved and should be set to zero.
23:16	20h RW	**Out Threshold (OUT_Threshold):** The OUT threshold is used to start the USB transfer as soon as the OUT threshold amount of data is fetched from system memory. It is also used to disconnect the data fetch, if the threshold amount of space is not available in the Packet Buffer.
15:8	00h RO	**reserved_15_8:** Reserved bits. These bits are reserved and should be set to zero.
7:0	20h RW	**IN Threshold (IN_Threshold):** The IN threshold is used to start the memory transfer as soon as the IN threshold amount of data is available in the Packet Buffer. It is also used to disconnect the data write, if the threshold amount of data is not available in the Packet Buffer.

16.6.2.15 Programmable Packet Buffer Depth (INSNREG02)—Offset 98h

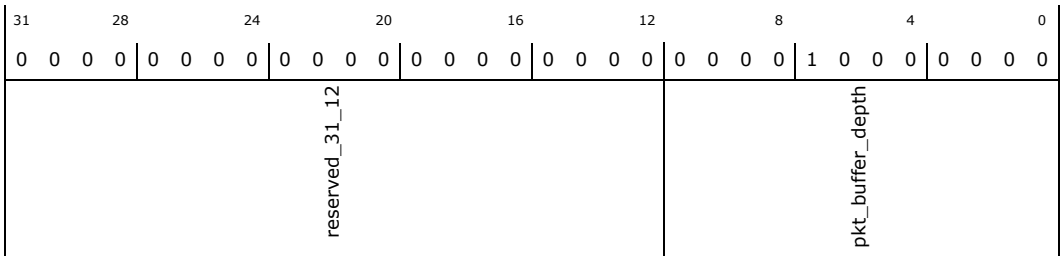
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 98h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000080h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	**reserved_31_12:** Reserved bits. These bits are reserved and should be set to zero.
11:0	80h RW	**Programmable Packet Buffer Depth (pkt_buffer_depth):** The value specified here is the number of DWORDs (32-bit entries).

16.6.2.16 Programmable Controller Settings (INSNREG03)—Offset 9Ch

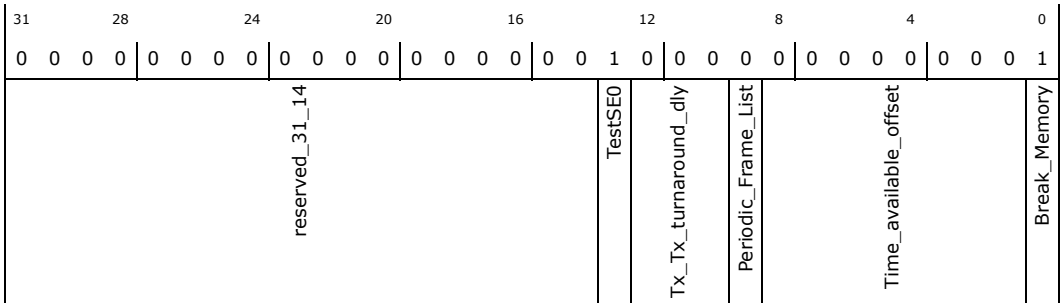
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 9Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00002001h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	**reserved_31_14:** Reserved bits. These bits are reserved and should be set to zero.
13	1h RW	**TestSE0 NAK (TestSE0):** When set to 1 (default), the core ignores the line state checking when transmitting SOF during the SE0_NAK test mode. When Set to 0, the port state machine disables the port if it does not find the line state to be in SE0 when transmitting SOF during the SE0_NAK testing. While doing impedance measurement during the SE0_NAK testing, the line state could go to non SE0 forcing the core to disable the port. This bit is used to control the port behavior during this.

Bit Range	Default & Access	Field Name (ID): Description
12:10	0h RW	**Transmit to Transmit turnaround delay (Tx_Tx_turnaround_dly):** This field specifies the extra delays in phy_clks to be added to the Transmit to Transmit turnaround delay value maintained in the core. The default value of this register field is 0. This default value of 0 is sufficient for most PHYs. But for some PHYs which puts wait states during the token packet, it may be required to program a value greater than 0 to meet the transmit to transmit minimum turnaround time. The recommendation to use the default value of 0 and change it only if there is an issue with minimum transmit-to-transmit turnaround time. This value should be programmed during core initialization and should not be changed afterwards.
9	0h RW	**Periodic Frame List Fetch (Periodic_Frame_List):** Setting this bit will force the host controller to fetch the periodic frame list in every microframe of a frame. If not set, then the periodic frame list will be fetched only in microframe 0 of every frame. The default is 0 (not set). This bit can be changed only during core initialization and should not be changed afterwards.
8:1	0h RW	**Time Available Offset (Time_available_offset):** This value indicates the additional number of bytes to be accommodated for the time-available calculation. The USB traffic on the bus can be started only when sufficient time is available to complete the packet within the EOF1 point. This time-available calculation is done in the hardware, and can be further offset by programming a value in this location. Note that time-available calculation is added for future flexibility and the application is not required to program this field by default.
0	1h RO	**Break Memory Transfer (Break_Memory):** 1: Enables this function 0: Disables this function Used in conjunction with INSNREG01 to enable breaking memory transactions into chunks once the OUT/IN threshold value is reached.

16.6.2.17 Programmable Controller Settings (INSNREG04)—Offset A0h

Bits [2:0] are used only for debug purposes.

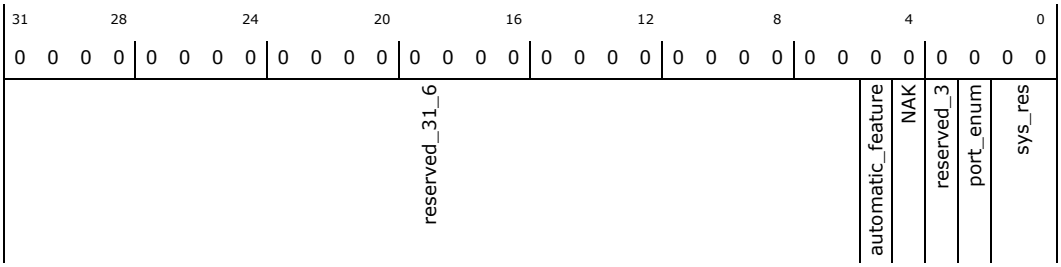
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + A0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:6	0h RO	**reserved_31_6:** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Field Name (ID): Description
5	0h RW	**Automatic Suspend Feature (automatic_feature):** 0: enables the automatic feature. The Suspend signal is deasserted (logic level 1) when run/stop is reset by software, but the hchalted bit is not yet set. 1: disables the automatic feature, which takes all ports out of suspend when software clears the run/stop bit. This is for backward compatibility. For systems where the host is halted without waking up all ports out of suspend, the port can become stuck because the PHYCLK is not running when the halt is programmed. To avoid this, the DWC H20AHB host core automatically pulls ports out of suspend when the host is halted by software. This bit is used to disable this automatic function. This is a functional bit.
4	0h RW	**NAK reload fix enabled (NAK):** When 1 NAK reload fix disabled. This is a functional bit for backward compatibility with Synopsys USB 2.0 Host-AHB core Release 2.40c.
3	0h RO	**reserved_3:** Reserved bits. These bits are reserved and should be set to zero.
2	0h RW	**Port Enumeration Time (port_enum):** Scales down port enumeration time. This is a debug bit.
1:0	0h RW	**System Reset (sys_res):** When sys_res[1]=1, HCCPARAMS bits 17,15:4 become writable. When sys_res[0]=1, HCCPARAMS register become writable Upon system reset, this field is 0. This is a debug bit.

16.6.2.18 UTMI Configuration (INSNREG05)—Offset A4h

This register gives access to implementation specific PHY vendor control and status registers. NOTE: Vendor control and status registers are not implemented. Read from Vendor Status will always return 0. Write to Vendor Control will have no effect.

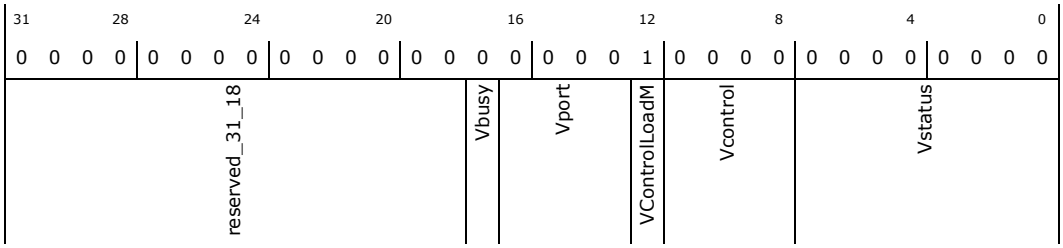
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + A4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:3] + 10h

Default: 00001000h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0h RO	**reserved_31_18:** Reserved bits. These bits are reserved and should be set to zero.
17	0h RO/V	**Vbusy:** Hardware indicator that a write to this register has occurred and the hardware is currently processing the operation defined by the data written. When processing is finished, this bit is cleared.

Bit Range	Default & Access	Field Name (ID): Description
16:13	0h RW	**Vport:** Port number from/to the PHY Vendor Status and Control value is read/written in Vstatus field. Vport meaningful values depend on the number of ports implemented. Allowed values are 4'h1 and 4'h2, any other value will return Vstatus=8'h0. Once software writes to VPort, from that write onwards, any writes to this register is ignored.
12	1h RW	**Vendor Control Load (VControlLoadM):** 0: Load. 1: NOP.
11:8	0h RW	**Vendor control register (Vcontrol):** Not implemented - write to Vendor Control will have no effect.
7:0	0h RO	**Vendor Status register (Vstatus):** Not implemented - read from Vendor Status will always return 0.

16.6.3 USB OHCI

Table 91. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"OHCI Revision (HCREVISION)—Offset 0h" on page 556	00000010h
4h	7h	"Host Controller Control (HCCONTROL)—Offset 4h" on page 556	00000000h
8h	Bh	"Host Controller Command Status (HCCMDSTATUS)—Offset 8h" on page 557	00000000h
Ch	Fh	"Host Controller Interrupt Status (HCINTRSTATUS)—Offset Ch" on page 559	00000000h
10h	13h	"Host Controller Interrupt Enable (HCINTRENABLE)—Offset 10h" on page 560	00000000h
14h	17h	"Host Controller Interrupt Disable (HCINTRDISABLE)—Offset 14h" on page 561	00000000h
18h	1Bh	"Host Controller Communication Area (HCHCCA)—Offset 18h" on page 562	00000000h
1Ch	1Fh	"Host Controller Current Isochronous or Interrupt Endpoint (HCPRDCURED)—Offset 1Ch" on page 563	00000000h
20h	23h	"Host Controller Current First Control Endpoint (HCCTRLHEADED)—Offset 20h" on page 563	00000000h
24h	27h	"Host Controller Current Control Endpoint (HCCTRLCURED)—Offset 24h" on page 564	00000000h
28h	2Bh	"Host Controller First Bulk Endpoint (HCBULKHEADED)—Offset 28h" on page 564	00000000h
2Ch	2Fh	"Host Controller Current Bulk Endpoint (HCBULKCURED)—Offset 2Ch" on page 565	00000000h
30h	33h	"Host Controller Last Completed Descriptor (HCDONEHEAD)—Offset 30h" on page 566	00000000h
34h	37h	"Host Controller Frame Interval (HCFMINTERVAL)—Offset 34h" on page 566	00002EDFh
38h	3Bh	"Host Controller Remaining Frame (HCFMREMAINING)—Offset 38h" on page 567	00000000h
3Ch	3Fh	"Host Controller Frame Number (HCFMNUMBER)—Offset 3Ch" on page 568	00000000h
40h	43h	"Host Controller Periodic List Start (HCPERIODICSTART)—Offset 40h" on page 569	00000000h
44h	47h	"Host Controller LS Threshold (HCLSTHRESHOLD)—Offset 44h" on page 569	00000628h
48h	4Bh	"Host Controller Root Hub Descriptor A (HCRHDESPA)—Offset 48h" on page 570	02000902h
4Ch	4Fh	"Host Controller Root Hub Descriptor B (HCRHDESPB)—Offset 4Ch" on page 571	00000000h
50h	53h	"Host Controller Root Hub Status (HCRHSTATUS)—Offset 50h" on page 572	00000000h
54h	57h	"Host Controller Root Hub Port Status (HCRHPORTSTS)—Offset 54h" on page 573	00000000h

16.6.3.1 OHCI Revision (HCREVISION)—Offset 0h

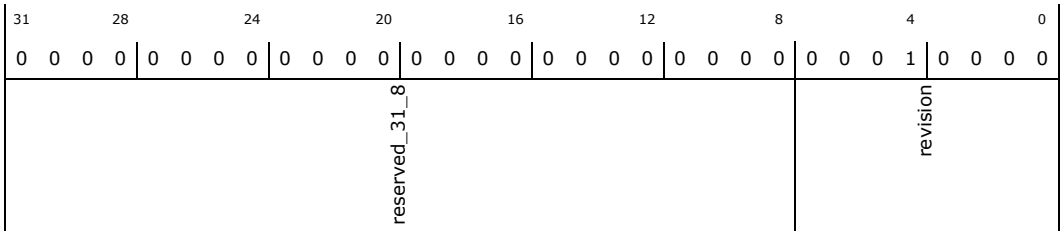
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCREVISION: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000010h



Bit Range	Default & Access	Description
31: 8	000000h RO	**reserved_31_8 (reserved_31_8):** Reserved bits. These bits are reserved and should be set to zero.
7: 0	10h RO	**Revision (revision):** This read-only field contains the BCD representation of the version of the HCI specification that is implemented by this HC.

16.6.3.2 Host Controller Control (HCCONTROL)—Offset 4h

The HCCONTROL register defines the operating modes for the Host Controller (HC). Most of the fields in this register are modified only by the Host Controller Driver (HCD), except hc_function_state and rmtwkup_connected.

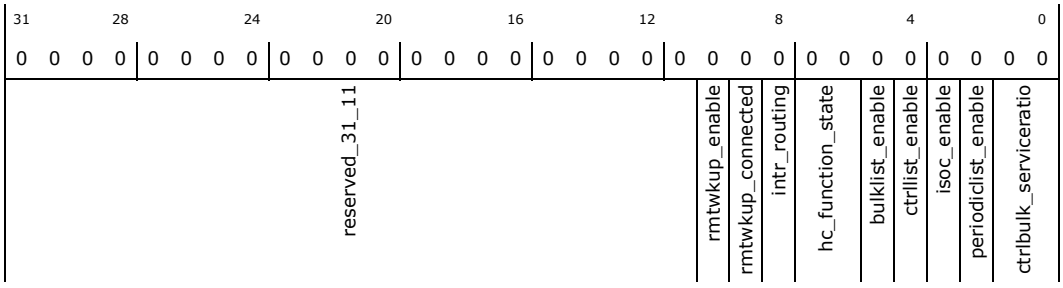
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

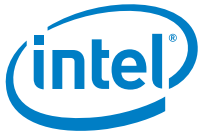
HCCONTROL: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RO	**reserved_31_11 (reserved_31_11):** Reserved bits. These bits are reserved and should be set to zero.



Bit Range	Default & Access	Description
10	0h RW	**Remote Wakeup Enable (rmtwkup_enable):** This bit is used by HCD to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the resume_detected bit in HCINTRSTATUS is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupts.
9	0h RW	**Remote Wakeup Connected (rmtwkup_connected):** This bit indicates whether HC supports remote wakeup signaling. If remote wakeup is supported and used by the system it is the responsibility of system firmware to set this bit during POST. HC clears the bit upon a hardware reset but does not alter it upon a software reset.
8	0h RW	**Interrupt Routing (intr_routing):** This bit determines the routing of interrupts generated by events registered in HCINTRSTATUS. If clear, all interrupts are routed to the normal host bus interrupt mechanism. If set, interrupts are routed to the System Management Interrupt. HCD clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. HCD uses this bit as a tag to indicate the ownership of HC.
7: 6	0h RW	**Host Controller Functional State for USB (hc_function_state):** 00: USBRESET 01: USBRESUME 10: USBOPERATIONAL 11: USBSUSPEND A transition to USBOPERATIONAL from another state causes SOF generation to begin 1ms later. HCD may determine whether HC has begun sending SOFs by reading the start_of_frame field of HCINTRSTATUS. This field may be changed by HC only when in the USBSUSPEND state. HC may move from the USBSUSPEND state to the USBRESUME state after detecting the resume signaling from a downstream port. HC enters USBSUSPEND after software reset, whereas it enters USBRESET after a hardware reset. The latter also resets the Root Hub and asserts subsequent reset signaling to downstream ports.
5	0h RW	**Bulk List Enable (bulklist_enable):** This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If HCBULKCURED is pointing to an ED (Endpoint Descriptor) to be removed, HCD must advance the pointer by updating HCBULKCURED before re-enabling processing of the list.
4	0h RW	**Control List Enable (ctrllist_enable):** This bit is set to enable the processing of the Control list in the next Frame. If cleared by HCD, processing of the Control list does not occur after the next SOF. HC must check this bit whenever it determines to process the list. When disabled, HCD may modify the list. If HCCTRLCURED is pointing to an ED to be removed, HCD must advance the pointer by updating HCCTRLCURED before re-enabling processing of the list.
3	0h RW	**Isochronous Enable (isoc_enable):** This bit is used by HCD to enable/disable processing of isochronous ED's. While processing the periodic list in a Frame, HC checks the status of this bit when it finds an Isochronous ED (F=1). If set (enabled), HC continues processing the ED's. If cleared (disabled), HC halts processing of the periodic list (which now contains only isochronous ED's) and begins processing the Bulk/Control lists. Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).
2	0h RW	**Periodic List Enable (periodiclist_enable):** This bit is set to enable the processing of the periodic list in the next Frame. If cleared by HCD, processing of the periodic list does not occur after the next SOF. HC must check this bit before it starts processing the list.
1: 0	0h RW	**Control Bulk Service Ratio (ctrlbulk_serviceratio):** This specifies the service ratio between Control and Bulk ED's. Before processing any of the nonperiodic lists, HC must compare the ratio specified with its internal count on how many nonempty Control ED's have been processed, in determining whether to continue serving another Control ED or switching to Bulk ED's. The internal count will be retained when crossing the frame boundary. In case of reset, HCD is responsible for restoring this value.

16.6.3.3 Host Controller Command Status (HCCMDSTATUS)—Offset 8h

The HCCMDSTATUS register is used by the Host Controller to receive commands issued by the Host Controller Driver, as well as reflecting the current status of the Host Controller. To the Host Controller Driver, it appears to be a 'write to set' register. The Host Controller must ensure that bits written as 1 become set in the register while bits

written as 0 remain unchanged in the register. The Host Controller Driver may issue multiple distinct commands to the Host Controller without concern for corrupting previously issued commands. The Host Controller Driver has normal read access to all bits. The sch_overrun_cnt field indicates the number of frames with which the Host Controller has detected the scheduling overrun error. This occurs when the Periodic list does not complete before EOF. When a scheduling overrun error is detected, the Host Controller increments the counter and sets the sch_overrun field in the HCINTRSTATUS register.

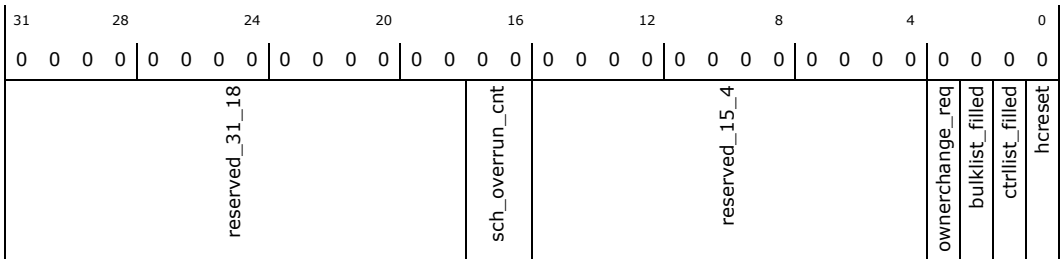
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCCMDSTATUS: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 18	0h RO	**reserved_31_18 (reserved_31_18):** Reserved bits. These bits are reserved and should be set to zero.
17: 16	0h RO	**Scheduling Overrun Count (sch_overrun_cnt):** These bits are incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if sch_overrun in HCINTRSTATUS has already been set. This is used by HCD to monitor any persistent scheduling problems
15: 4	0h RO	**reserved_15_4 (reserved_15_4):** Reserved bits. These bits are reserved and should be set to zero.
3	0h RW/1C	**Ownership Change Request (ownerchange_req):** This bit is set by an OS HCD to request a change of control of the HC. When set HC will set the ownerchange_req field in HCINTRSTATUS. After the changeover, this bit is cleared and remains so until the next request from OS HCD.
2	0h RW	**BulkListFilled (bulklist_filled):** This bit is used to indicate whether there are any TDs on the Bulk list. It is set by HCD whenever it adds a TD to an ED in the Bulk list. When HC begins to process the head of the Bulk list, it checks bulklist_filled. As long as bulklist_filled is 0, HC will not start processing the Bulk list. If bulklist_filled is 1, HC will start processing the Bulk list and will set bulklist_filled to 0. If HC finds a TD on the list, then HC will set bulklist_filled to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set bulklist_filled, then bulklist_filled will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop.
1	0h RW	**Control List Filled (ctrllist_filled):** This bit is used to indicate whether there are any TDs on the Control list. It is set by HCD whenever it adds a TD to an ED in the Control list. When HC begins to process the head of the Control list, it checks ctrllist_filled. As long as ctrllist_filled is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set ctrllist_filled to 0. If HC finds a TD on the list, then HC will set ctrllist_filled to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if the HCD does not set ctrllist_filled, then ctrllist_filled will still be 0 when HC completes processing the Control list and Control list processing will stop.

Bit Range	Default & Access	Description
0	0h RW	**Host Controller Reset (hcreset):** This bit is set by HCD to initiate a software reset of HC. Regardless of the functional state of HC, it moves to the USBSUSPEND state in which most of the operational registers are reset except those stated otherwise; e.g., the intr_routing field of HCCONTROL, and no Host bus accesses are allowed. This bit is cleared by HC upon the completion of the reset operation. The reset operation must be completed within 10ms. This bit, when set, should not cause a reset to the Root Hub and no subsequent reset signaling should be asserted to its downstream ports.

16.6.3.4 Host Controller Interrupt Status (HCINTRSTATUS)—Offset Ch

This register provides status on various events that cause hardware interrupts. When an event occurs, Host Controller sets the corresponding bit in this register. When a bit becomes set, a hardware interrupt is generated if the interrupt is enabled in the HCINTRENABLE register and the mstr_intr_enable bit is set. The Host Controller Driver may clear specific bits in this register by writing 1 to bit positions to be cleared. The Host Controller Driver may not set any of these bits. The Host Controller will never clear the bit.

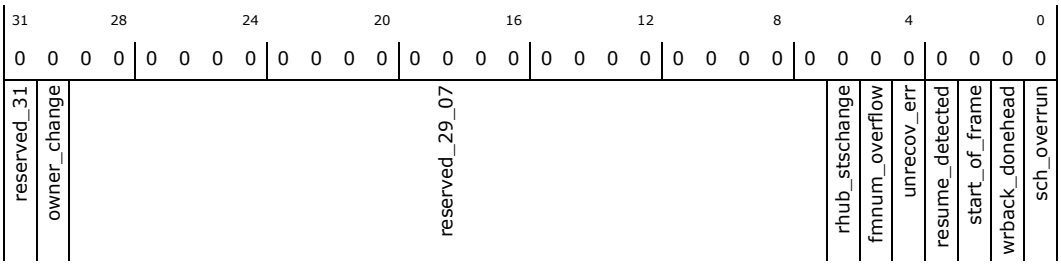
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCINTRSTATUS: [BAR0] + Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0h RO	**reserved_31 (reserved_31):** Reserved bits. These bits are reserved and should be set to zero.
30	0h RW/1C	**Ownership Change (owner_change):** This bit is set by HC when HCD sets the ownerchange_req field in HCCMDSTATUS. This event, when unmasked, will always generate a System Management Interrupt (SMI) immediately. This bit is tied to 0 when the SMI pin is not implemented. NOTE: OHCI SMIs are ignored by the system.
29: 7	0h RO	**reserved_29_07 (reserved_29_07):** Reserved bits. These bits are reserved and should be set to zero.
6	0h RW/1C	**Root Hub Status Change (rhub_stschange):** This bit is set when the content of HCRHSTATUS or the content of any of HCRHPORTSTS[NumberofDownstreamPort] has changed.
5	0h RW/1C	**Frame Number Overflow (fmnum_overflow):** This bit is set when the MSb of HCFMNUMBER (bit 15) changes value, from 0 to 1 or from 1 to 0, and after HccaFrameNumber has been updated.
4	0h RW/1C	**Unrecoverable Error (unrecov_err):** This bit is set when HC detects a system error not related to USB. HC should not proceed with any processing or signaling before the system error has been corrected. HCD clears this bit after HC has been reset.

Bit Range	Default & Access	Description
3	0h RW/1C	**Resume Detected (resume_detected):** This bit is set when HC detects that a device on the USB is asserting resume signaling. It is the transition from no resume signaling to resume signaling causing this bit to be set. This bit is not set when HCD sets the USBRESUME state.
2	0h RW/1C	**Start of Frame (start_of_frame):** This bit is set by HC at each start of a frame and after the update of HccaFrameNumber. HC also generates a SOF token at the same time.
1	0h RW/1C	**Writeback Done Head (wrback_donehead):** This bit is set immediately after HC has written HCDONEHEAD to HccaDoneHead. Further updates of the HccaDoneHead will not occur until this bit has been cleared. HCD should only clear this bit after it has saved the content of HccaDoneHead.
0	0h RW/1C	**Scheduling Overrun (sch_overrun):** This bit is set when the USB schedule for the current Frame overruns and after the update of HccaFrameNumber. A scheduling overrun will also cause the sch_overrun_cnt of HCCMDSTATUS to be incremented.

16.6.3.5 Host Controller Interrupt Enable (HCINTRENABLE)—Offset 10h

Each enable bit in the HCINTRENABLE register corresponds to an associated interrupt bit in the HCINTRSTATUS register. The HCINTRENABLE register is used to control which events generate a hardware interrupt. When a bit is set in the HCINTRSTATUS register AND the corresponding bit in the HCINTRENABLE register is set AND the mstr_intr_enable bit is set, then a hardware interrupt is requested on the host bus. Writing a '1' to a bit in this register sets the corresponding bit, whereas writing a '0' to a bit in this register leaves the corresponding bit unchanged. On read, the current value of this register is returned.

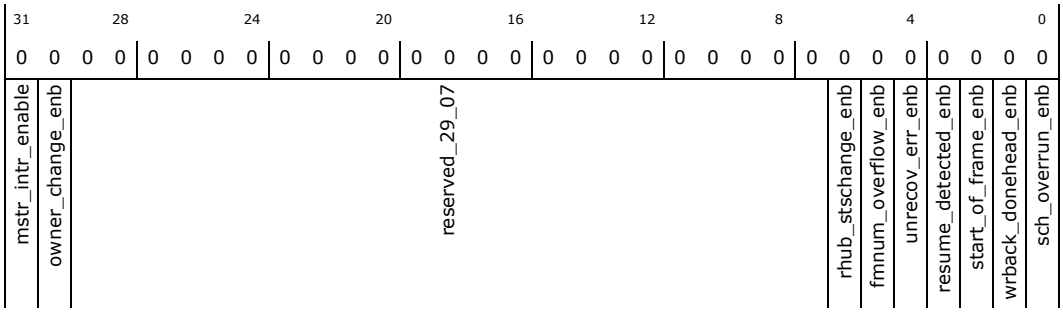
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCINTRENABLE: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0h RW/1S	**Master Interrupt Enable (mstr_intr_enable):** A 0 written to this field is ignored by HC. A 1 written to this field enables interrupt generation due to events specified in the other bits of this register. This is used by HCD as a Master Interrupt Enable.
30	0h RW/1S	**Owner Change Enable (owner_change_enb):** 0: Ignore 1: Enable interrupt generation due to Ownership Change.
29: 7	0h RO	**reserved_29_07 (reserved_29_07):** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Description
6	0h RW/1S	**Root Hub Status Change Enable (rhub_stschange_enb):** 0: Ignore 1: Enable interrupt generation due to Root Hub Status Change
5	0h RW/1S	**Frame Number Overflow Enable (fmnum_overflow_enb):** 0: Ignore 1: Enable interrupt generation due to Frame Number Overflow
4	0h RW/1S	**Unrecoverable Error Enable (unrecov_err_enb):** 0: Ignore 1: Enable interrupt generation due to Unrecoverable Error
3	0h RW/1S	**Resume Detected Enable (resume_detected_enb):** 0: Ignore 1: Enable interrupt generation due to Resume Detect
2	0h RW/1S	**Start of Frame Enable (start_of_frame_enb):** 0: Ignore 1: Enable interrupt generation due to Start of Frame
1	0h RW/1S	**HCDONEHEAD Writeback Enable (wrback_donehead_enb):** 0: Ignore 1: Enable interrupt generation due to HCDONEHEAD Writeback
0	0h RW/1S	**Scheduling Overrun Enable (sch_overrun_enb):** 0: Ignore 1: Enable interrupt generation due to Scheduling Overrun

16.6.3.6 Host Controller Interrupt Disable (HCINTRDISABLE)—Offset 14h

Each disable bit in the HCINTRDISABLE register corresponds to an associated interrupt bit in the HCINTRSTATUS register. The HCINTRDISABLE register is coupled with the HCINTRENABLE register. Thus, writing a '1' to a bit in this register clears the corresponding bit in the HCINTRENABLE register, whereas writing a '0' to a bit in this register leaves the corresponding bit in the HCINTRENABLE register unchanged. On read, the current value of the HCINTRENABLE register is returned.

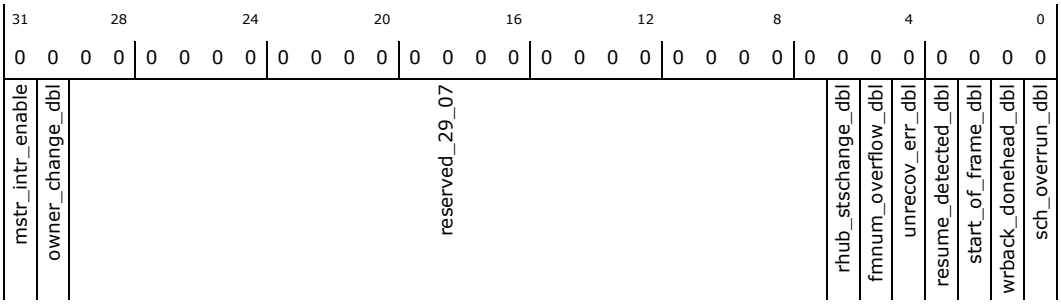
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCINTRDISABLE: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0h RW/1C	**Master Interrupt Enable (mstr_intr_enable):** A '0' written to this field is ignored by HC. A '1' written to this field disables interrupt generation due to events specified in the other bits of this register. This field is set after a hardware or software reset.
30	0h RW/1C	**Owner Change Disable (owner_change_dbl):** 0: Ignore 1: Disable interrupt generation due to Ownership Change
29: 7	0h RO	**reserved_29_07 (reserved_29_07):** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Description
6	0h RW/1C	**Root Hub Status Change Disable (rhub_stschange_dbl):** 0: Ignore 1: Disable interrupt generation due to Root Hub Status Change
5	0h RW/1C	**Frame Number Overflow Disable (fmnum_overflow_dbl):** 0: Ignore 1: Disable interrupt generation due to Frame Number Overflow
4	0h RW/1C	**Unrecoverable Error Disable (unrecov_err_dbl):** 0: Ignore 1: Disable interrupt generation due to Unrecoverable Error
3	0h RW/1C	**Resume Detected Disable (resume_detected_dbl):** 0: Ignore 1: Disable interrupt generation due to Resume Detect
2	0h RW/1C	**Start of Frame Disable (start_of_frame_dbl):** 0: Ignore 1: Disable interrupt generation due to Start of Frame
1	0h RW/1C	**HCDONEHEAD Writeback Disable (wrback_donehead_dbl):** 0: Ignore 1: Disable interrupt generation due to HCDONEHEAD Writeback
0	0h RW/1C	**Scheduling Overrun Disable (sch_overrun_dbl):** 0: Ignore 1: Disable interrupt generation due to Scheduling Overrun

16.6.3.7 Host Controller Communication Area (HCHCCA)—Offset 18h

The HCHCCA register contains the physical address of the Host Controller Communication Area. The Host Controller Driver determines the alignment restrictions by writing all 1s to HCHCCA and reading the content of HCHCCA. The alignment is evaluated by examining the number of zeroes in the lower order bits. The minimum alignment is 256 bytes; therefore, bits 0 through 7 must always return '0' when read. Detailed description can be found in Chapter 4. This area is used to hold the control structures and the Interrupt table that are accessed by both the Host Controller and the Host Controller Driver.

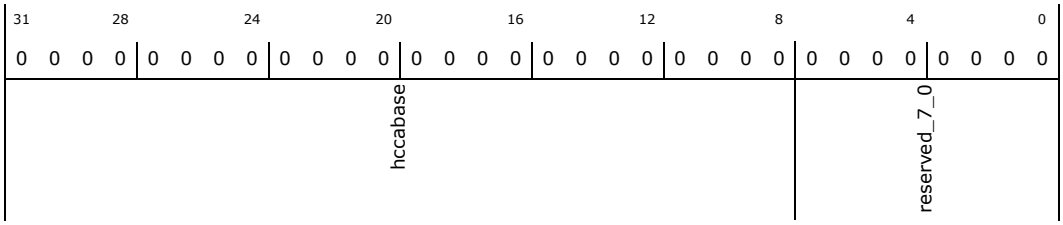
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCHCCA: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 8	0h RW	**Host Controller Communication Area Base (hccabase):** This is the base address of the Host Controller Communication Area
7: 0	0h RO	**reserved_7_0 (reserved_7_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.8 Host Controller Current Isochronous or Interrupt Endpoint (HCPRDCURED)—Offset 1Ch

The HcPeriodCurrentED register contains the physical address of the current Isochronous or Interrupt Endpoint Descriptor.

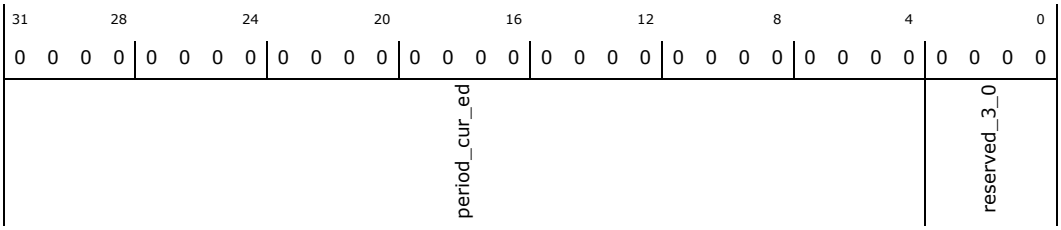
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCPRDCURED: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RO	**Period Current ED (period_cur_ed):** This is used by HC to point to the head of one of the Periodic lists which will be processed in the current Frame. The content of this register is updated by HC after a periodic ED has been processed. HCD may read the content in determining which ED is currently being processed at the time of reading.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.9 Host Controller Current First Control Endpoint (HCCTRLHEADED)—Offset 20h

The HCCTRLHEADED register contains the physical address of the first Endpoint Descriptor of the Control list.

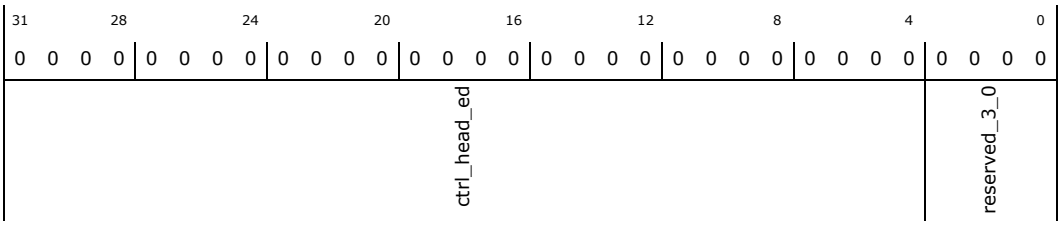
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCCTRLHEADED: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RW	**Control Head ED (ctrl_head_ed):** HC traverses the Control list starting with the HCCTRLHEADED pointer. The content is loaded from HCCA during the initialization of HC.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.10 Host Controller Current Control Endpoint (HCCTRLCURED)—Offset 24h

The HCCTRLCURED register contains the physical address of the current Endpoint Descriptor of the Control list.

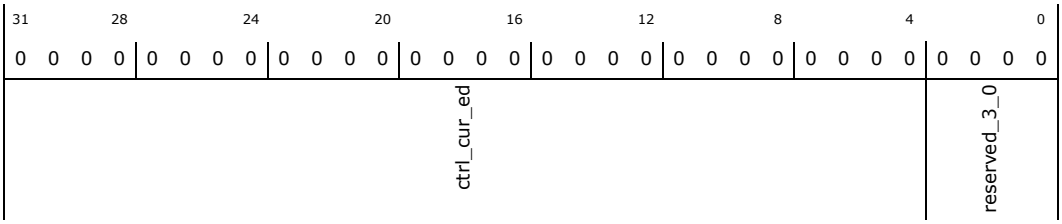
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCCTRLCURED: [BAR0] + 24h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RW	**Control Current ED (ctrl_cur_ed):** This pointer is advanced to the next ED after serving the present one. HC will continue processing the list from where it left off in the last Frame. When it reaches the end of the Control list, HC checks the ctrllist_filled of in HCCMDSTATUS. If set, it copies the content of HCCTRLHEADED to HCCTRLCURED and clears the bit. If not set, it does nothing. HCD is allowed to modify this register only when the ctrllist_enable of HCCONTROL is cleared. When set, HCD only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.11 Host Controller First Bulk Endpoint (HCBULKHEADED)—Offset 28h

The HCBULKHEADED register contains the physical address of the first Endpoint Descriptor of the Bulk list.

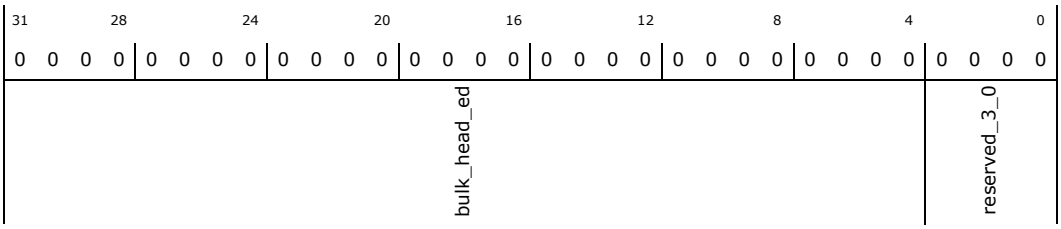
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCBULKHEADED: [BAR0] + 28h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RW	**Bulk Head ED (bulk_head_ed):** HC traverses the Bulk list starting with the HCBULKHEADED pointer. The content is loaded from HCCA during the initialization of HC.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.12 Host Controller Current Bulk Endpoint (HCBULKCURED)—Offset 2Ch

The HCBULKCURED register contains the physical address of the current endpoint of the Bulk list. As the Bulk list will be served in a round-robin fashion, the endpoints will be ordered according to their insertion to the list.

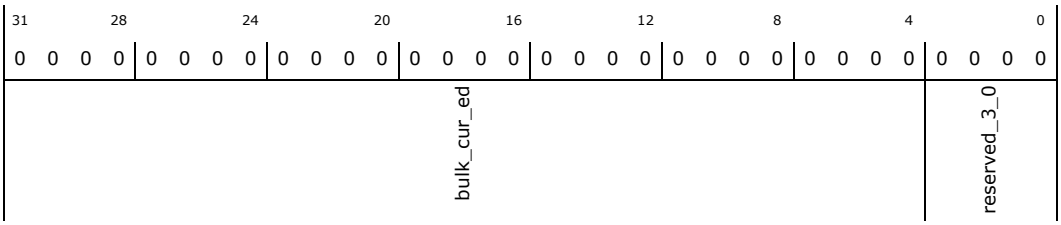
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCBULKCURED: [BAR0] + 2Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RW	**Bulk Current ED (bulk_cur_ed):** This is advanced to the next ED after the HC has served the present one. HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, HC checks the ctrllist_filled of HCCONTROL. If set, it copies the content of HCBULKHEADED to HCBULKCURED and clears the bit. If it is not set, it does nothing. HCD is only allowed to modify this register when the bulklist_enable of HCCONTROL is cleared. When set, the HCD only reads the instantaneous value of this register. This is initially set to zero to indicate the end of the Bulk list.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.13 Host Controller Last Completed Descriptor (HCDONEHEAD)—Offset 30h

The HCDONEHEAD register contains the physical address of the last completed Transfer Descriptor that was added to the Done queue. In normal operation, the Host Controller Driver should not need to read this register as its content is periodically written to the HCCA.

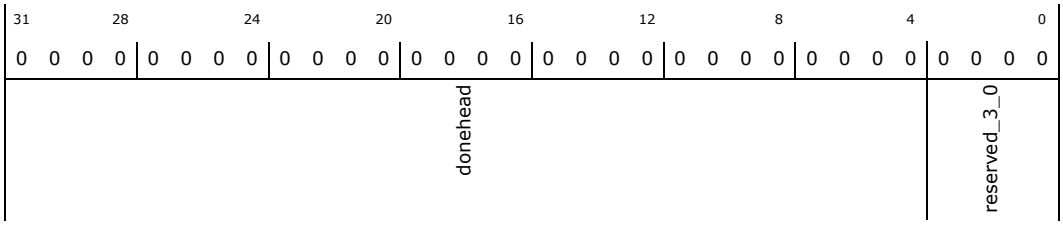
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCDONEHEAD: [BAR0] + 30h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 4	0h RO	**Done Head (donehead):** When a TD is completed, HC writes the content of HCDONEHEAD to the NextTD field of the TD. HC then overwrites the content of HCDONEHEAD with the address of this TD. This is set to zero whenever HC writes the content of this register to HCCA. It also sets the wrback_donehead of HCINTRSTATUS.
3: 0	0h RO	**reserved_3_0 (reserved_3_0):** Reserved bits. These bits are reserved and should be set to zero.

16.6.3.14 Host Controller Frame Interval (HCFMINTERVAL)—Offset 34h

The HCFMINTERVAL register contains a 14-bit value which indicates the bit time interval in a Frame, (i.e., between two consecutive SOFs), and a 15-bit value indicating the Full Speed maximum packet size that the Host Controller may transmit or receive without causing scheduling overrun. The Host Controller Driver may carry out minor adjustment on the fm_interval by writing a new value over the present one at each SOF. This provides the programmability necessary for the Host Controller to synchronize with an external clocking resource and to adjust any unknown local clock offset.

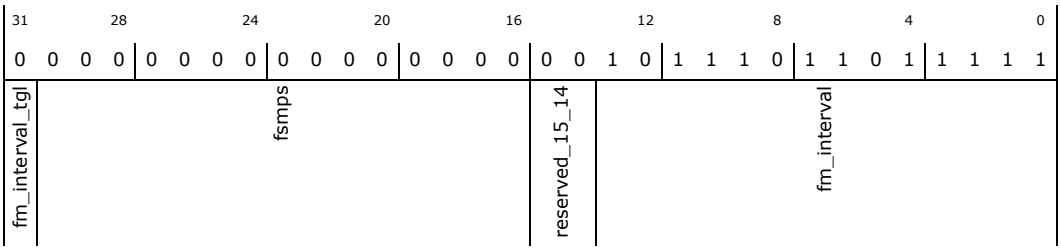
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCFMINTERVAL: [BAR0] + 34h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00002EDFh



Bit Range	Default & Access	Description
31	0h RW	**Frame Interval Toggle (fm_interval_tgl):** HCD toggles this bit whenever it loads a new value to fm_interval.
30: 16	0h RW	**FS Largest Data Packet (fsmps):** This field specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame. The counter value represents the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The field value is calculated by the HCD.
15: 14	0h RO	**reserved_15_14 (reserved_15_14):** Reserved bits. These bits are reserved and should be set to zero.
13: 0	2EDFh RW	**Frame Interval (fm_interval):** This specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999.HCD should store the current value of this field before resetting HC. By setting the hcreset field of HCCMDSTATUS as this will cause the HC to reset this field to its nominal value. HCD may choose to restore the stored value upon the completion of the Reset sequence.

16.6.3.15 Host Controller Remaining Frame (HCFMREMAINING)—Offset 38h

The HCFMREMAINING register is a 14-bit down counter showing the bit time remaining in the current Frame. NOTE: In OpenHCI 1.0a, the Frame Remaining and Frame Remaining Toggle bits in the HCFMREMAINING register are read-only to the Host Controller Driver. However, it is allowed to write to these bits for debugging purposes. Though these bits are writable, the Host Controller Driver must not write into these bits. Doing so yields undefined results.

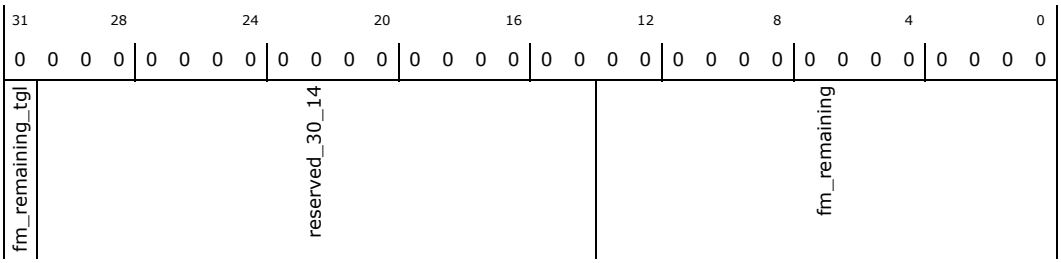
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCFMREMAINING: [BAR0] + 38h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0h RW	**Frame Remaining Toggle (fm_remaining_tgl):** This bit is loaded from the fm_interval_tgl field of fm_interval whenever fm_remaining reaches 0. This bit is used by HCD for the synchronization between fm_interval and fm_remaining. NOTE: Though these bits are writable, the Host Controller Driver must not write into these bits. Doing so yields undefined results.
30: 14	0h RO	**reserved_30_14 (reserved_30_14):** Reserved bits. These bits are reserved and should be set to zero.
13: 0	0h RW	**Frame Remaining (fm_remaining):** This counter is decremented at each bit time. When it reaches zero, it is reset by loading the fm_interval value specified in fm_interval at the next bit time boundary. When entering the USBOPERATIONAL state, HC re-loads the content with the fm_interval of HCFMINTERVAL and uses the updated value from the next SOF. NOTE: Though these bits are writable, the Host Controller Driver must not write into these bits. Doing so yields undefined results.

16.6.3.16 Host Controller Frame Number (HCFMNUMBER)—Offset 3Ch

The HCFMNUMBER register is a 16-bit counter. It provides a timing reference among events happening in the Host Controller and the Host Controller Driver. The Host Controller Driver may use the 16-bit value specified in this register and generate a 32-bit frame number without requiring frequent access to the register. NOTE: In OpenHCI 1.0a, the Frame Number bits in the HCFMNUMBER register are read-only to the Host Controller Driver. However, it is allowed to write to these bits for debugging purposes. Though these bits are writable, the Host Controller Driver must not write into these bits. Doing so yields undefined results.

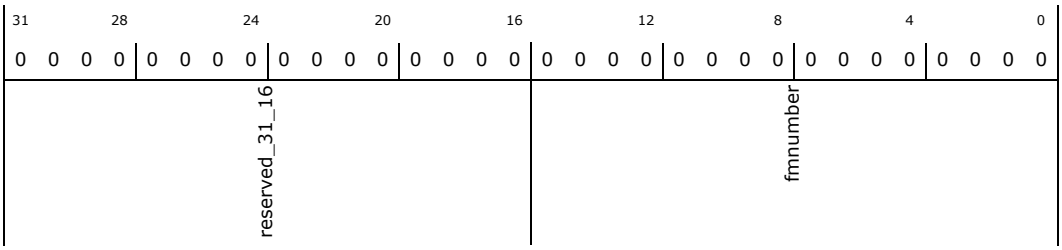
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCFMNUMBER: [BAR0] + 3Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 16	0h RO	**reserved_31_16 (reserved_31_16):** Reserved bits. These bits are reserved and should be set to zero.
15: 0	0h RW	**Frame Number (fmnumber):** This is incremented when fm_remaining is re-loaded. It will be rolled over to 0 after FFFF. When entering the USBOPERATIONAL state, this will be incremented automatically. The content will be written to HCCA after HC has incremented the FrameNumber at each frame boundary and sent a SOF but before HC reads the first ED in that Frame. After writing to HCCA, HC will set the start_of_frame in HCINTRSTATUS. NOTE: Though these bits are writable, the Host Controller Driver must not write into these bits. Doing so yields undefined results.

16.6.3.17 Host Controller Periodic List Start (HCPERIODICSTART)—Offset 40h

The HcPeriodicStart register has a 14-bit programmable value which determines when is the earliest time HC should start processing the periodic list.

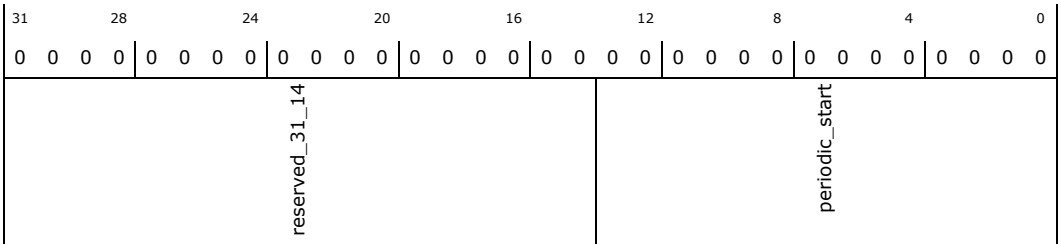
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCPERIODICSTART: [BAR0] + 40h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 14	0h RO	**reserved_31_14 (reserved_31_14):** Reserved bits. These bits are reserved and should be set to zero.
13: 0	0h RW	**Periodic Start (periodic_start):** After hardware reset, this field is cleared. This is then set by HCD during the HC initialization. The value is calculated roughly as 10% off from fm_interval. A typical value will be h3E67. When fm_remaining reaches the value specified, processing of the periodic lists will have priority over Control/Bulk processing. HC will therefore start processing the Interrupt list after completing the current Control or Bulk transaction that is in progress.

16.6.3.18 Host Controller LS Threshold (HCLSTHRESHOLD)—Offset 44h

The HcLSThreshold register contains an 11-bit value used by the Host Controller to determine whether to commit to the transfer of a maximum of 8-byte LS packet before EOF. Neither the Host Controller nor the Host Controller Driver is allowed to change this value.

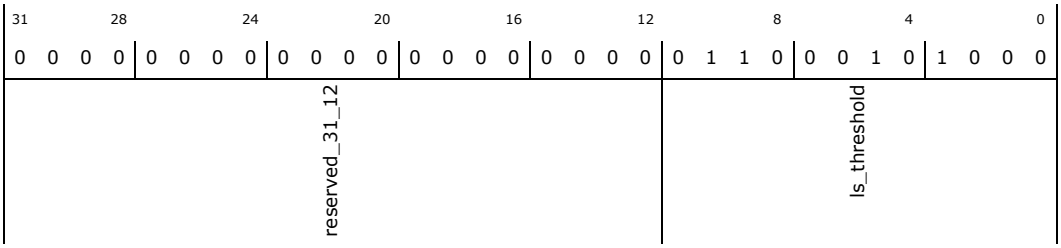
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCLSTHRESHOLD: [BAR0] + 44h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000628h



Bit Range	Default & Access	Description
31: 12	0h RO	**reserved_31_12 (reserved_31_12):** Reserved bits. These bits are reserved and should be set to zero.
11: 0	0628h RW	**LS Threshold (ls_threshold):** This field contains a value which is compared to the fm_remaining field prior to initiating a Low Speed transaction. The transaction is started only if fm_remaining this field. The value is calculated by HCD with the consideration of transmission and setup overhead. NOTE: Neither the Host Controller nor the Host Controller Driver are allowed to change this value.

16.6.3.19 Host Controller Root Hub Descriptor A (HCRHDESPA)—Offset 48h

The HCRHDESPA register is the first register of two describing the characteristics of the Root Hub.

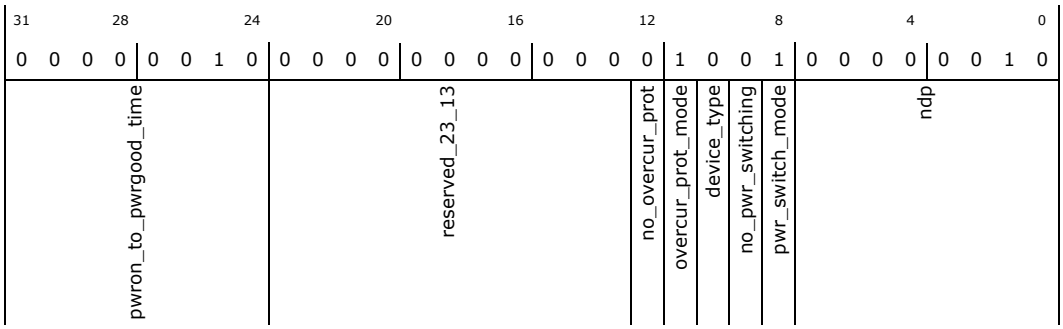
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCRHDESPA: [BAR0] + 48h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 02000902h



Bit Range	Default & Access	Description
31: 24	02h RW	**Power On To Power Good Time (pwron_to_pwrgood_time):** This byte specifies the duration HCD has to wait before accessing a powered-on port of the Root Hub. It is implementation-specific. The unit of time is 2ms. The duration is calculated as pwron_to_pwrgood_time*2ms.
23: 13	0h RO	**reserved_23_13 (reserved_23_13):** Reserved bits. These bits are reserved and should be set to zero.
12	0h RW	**No Over Current Protection (no_overcur_prot):** This bit describes how the overcurrent status for the Root Hub ports is reported. When this bit is cleared, the overcur_prot_mode field specifies global or per-port reporting. 0: Over-current status is reported collectively for all downstream ports 1: No overcurrent protection supported
11	1h RW	**Over Current Protection Mode (overcur_prot_mode):** This bit describes how the overcurrent status for the Root Hub ports is reported. At reset, this field should reflect the same mode as pwr_switch_mode. This field is valid only if the no_overcur_prot field is cleared. 0: over-current status is reported collectively for all downstream ports 1: over-current status is reported on a per-port basis
10	0h RO	**Device Type (device_type):** This bit specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read/write 0.

Bit Range	Default & Access	Description
9	0h RW	**No Power Switching (no_pwr_switching):** These bits are used to specify whether power switching is supported or port are always powered. It is implementation-specific. When this bit is cleared, the pwr_switch_mode specifies global or per-port switching. 0: Ports are power switched 1: Ports are always powered on when the HC is powered on
8	1h RW	**Power Switching Mode (pwr_switch_mode):** This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the no_pwr_switching field is cleared. 0: all ports are powered at the same time. 1: each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the port_pwr_ctrlmask bit is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ ClearGlobalPower).
7: 0	02h RO	**Number Downstream Ports (ndp):** These bits specify the number of downstream ports (ndp) supported by the Root Hub. It is implementation-specific. The minimum number of ports is 1.

16.6.3.20 Host Controller Root Hub Descriptor B (HCRHDESPB)—Offset 4Ch

The HCRHDESPB register is the second register of two describing the characteristics of the Root Hub. These fields are written during initialization to correspond with the system implementation.

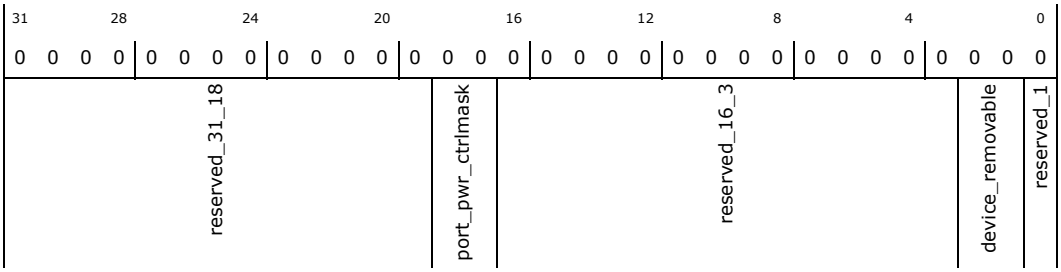
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCRHDESPB: [BAR0] + 4Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 19	0h RO	**reserved_31_18 (reserved_31_18):** Reserved bits. These bits are reserved and should be set to zero.
18: 17	0h RW	**Port Power Control Mask (port_pwr_ctrlmask):** Each bit indicates if a port is affected by a global power control command when pwr_switch_mode is set. When set, the port's power state is only affected by per-port power control (Set/ClearPortPower). When cleared, the port is controlled by the global power switch (Set/ClearGlobalPower). If the device is configured to global switching mode (pwr_switch_mode=0), this field is not valid. bit 0: Reserved bit 1: Ganged-power mask on Port #1 bit 2: Ganged-power mask on Port #2 NOTE: Per Port Power Control is supported
16: 3	0h RO	**reserved_16_3 (reserved_16_3):** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Description
2: 1	0h RW	**Device Removable (device_removable):** Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable. bit 0: Reserved bit 1: Device attached to Port #1 bit 2: Device attached to Port #2
0	0h RO	**reserved_1 (reserved_1):** Reserved bit. These bits are reserved and should be set to zero.

16.6.3.21 Host Controller Root Hub Status (HCRHSTATUS)—Offset 50h

The HCRHSTATUS register is divided into two parts. The lower word of a Dword represents the Hub Status field and the upper word represents the Hub Status Change field. Reserved bits should always be written '0'.

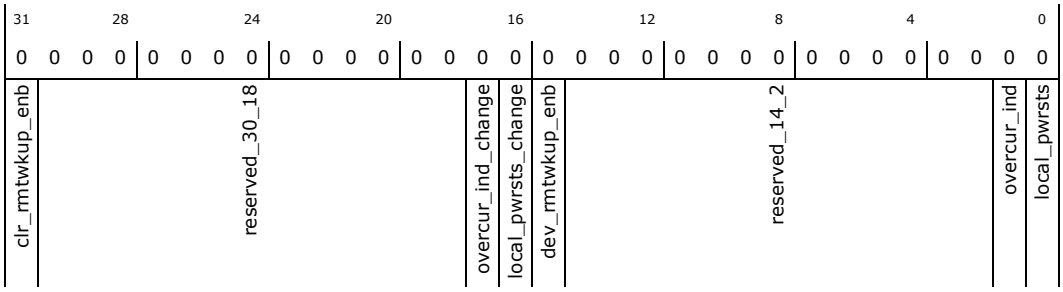
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCRHSTATUS: [BAR0] + 50h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0h WO	**Clear Remote Wakeup Enable (clr_rmtwkup_enb):** Writing a '1' clears dev_rmtwkup_enb. Writing a '0' has no effect.
30: 18	0h RO	**reserved_30_18 (reserved_30_18):** Reserved bits. These bits are reserved and should be set to zero.
17	0h RW/1S	**Over Current Indicator Change (overcur_ind_change):** This bit is set by hardware when a change has occurred to the OCI field of this register. The HCD clears this bit by writing a 1.Writing a 0 has no effect.
16	0h RW/1S	**Local Power Status Change / SetGlobalPower (local_pwrsts_change):** (read) Local Power Status Change. The Root Hub does not support the local power status feature and this bit is always read as 0. (write) SetGlobalPower. In global power mode (pwr_switch_mode=0), this bit is written to 1 to turn on power to all ports (clear pps). In per-port power mode, it sets pps only on ports whose port_pwr_ctrlmask bit is not set. Writing a 0 has no effect.

Bit Range	Default & Access	Description
15	0h RW/1S	**Device Remote Wakeup Enable / SetRemoteWakeupEnable (dev_rmtwkup_enb):** (read) Device Remote Wakeup Enable. This bit enables a csc bit as a resume event, causing a USBSUSPEND to USBRESUME state transition and setting the resume_detected interrupt. 0: csc is not a remote wakeup event. 1: csc is a remote wakeup event. (write) SetRemoteWakeupEnable. Writing a '1' sets dev_rmtwkup_enb. Writing a '0' has no effect.
14: 2	0h RO	**reserved_14_2 (reserved_14_2):** Reserved bits. These bits are reserved and should be set to zero.
1	0h RO	**Over Current Indicator (overcur_ind):** This bit reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal. If per-port overcurrent protection is implemented this bit is always 0
0	0h RW/1S	**Local Power Status / ClearGlobalPower (local_pwrsts):** (read)Local Power Status. The Root Hub does not support the local power status feature; thus, this bit is always read as 0. (write) ClearGlobalPower. In global power mode (pwr_switch_mode=0), this bit is written to 1 to turn off power to all ports (clear pps). In per-port power mode, it clears pps only on ports whose port_pwr_ctrlmask bit is not set. Writing a 0 has no effect.

16.6.3.22 Host Controller Root Hub Port Status (HCRHPORTSTS)—Offset 54h

The HCRHPORTSTS[1:ndp] register is used to control and report port events on a per-port basis. ndp represents the number of HCRHPORTSTS registers that are implemented in hardware. The lower word is used to reflect the port status, whereas the upper word reflects the status change bits. Some status bits are implemented with special write behavior (see below). If a transaction (token through handshake) is in progress when a write to change port status occurs, the resulting port status change must be postponed until the transaction completes. Reserved bits should always be written '0'.

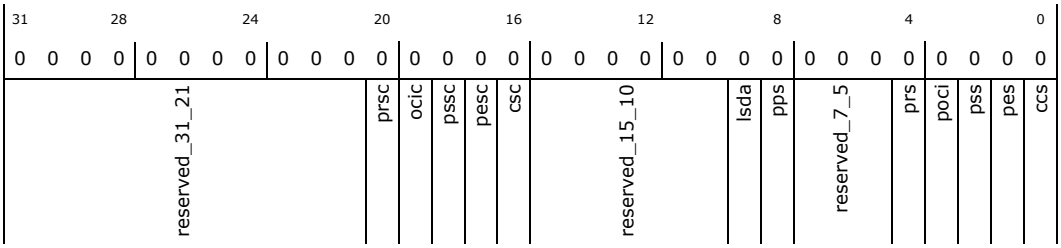
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

HCRHPORTSTS: [BAR0] + 54h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:4] + 10h

Default: 00000000h

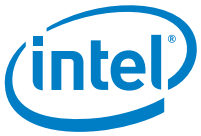


Bit Range	Default & Access	Description
31: 21	0h RO	**reserved_31_21 (reserved_31_21):** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Description
20	0b RW/1C	**Port Reset Status Change (prsc):** This bit is set at the end of the 10-ms port reset signal. The HCD writes a 1 to clear this bit. Writing a 0 has no effect. 0: port reset is not complete 1: port reset is complete
19	0b RW/1C	**Port Over Current Indicator Change (ocic):** This bit is valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the poci bit. The HCD writes a 1 to clear this bit. Writing a 0 has no effect. 0: no change in poci 1: poci has changed
18	0b RW/1C	**Port Suspend Status Change (pssc):** This bit is set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resychronization delay. The HCD writes a 1 to clear this bit. Writing a 0 has no effect. This bit is also cleared when prsc is set. 0: resume is not completed 1: resume completed
17	0b RW/1C	**Port Enable Status Change (pesc):** This bit is set when hardware events cause the pes bit to be cleared. Changes from HCD writes do not set this bit. The HCD writes a 1 to clear this bit. Writing a 0 has no effect. 0: no change in pes 1: change in pes
16	0b RW/1C	**Connect Status Change (csc):** This bit is set whenever a connect or disconnect event occurs. The HCD writes a 1 to clear this bit. Writing a 0 has no effect. If ccs is cleared when a SetPortReset, SetPortEnable, or SetPortSuspend write occurs, this bit is set to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected. 0: no change in ccs 1 = change in ccs Note: If the device_removable[ndp] bit is set, this bit is set only after a Root Hub reset to inform the system that the device is attached.
15: 10	0h RO	**reserved_15_10 (reserved_15_10):** Reserved bits. These bits are reserved and should be set to zero.
9	0b RW/1C	**Speed Device Attached / Clear Port Power (lsda):** (read) Low or Full Speed Device Attached. This bit indicates the speed of the device attached to this port. When set, a Low Speed device is attached to this port. When clear, a Full Speed device is attached to this port. This field is valid only when the ccs is set. 0: full speed device attached 1: low speed device attached (write) ClearPortPower. The HCD clears the pps bit by writing a 1 to this bit. Writing a 0 has no effect.
8	0b RW/1C	**Port Power Status / Set Port Power (pps):** (read) Port Power Status. This bit reflects the port power status, regardless of the type of power switching implemented. This bit is cleared if an overcurrent condition is detected. HCD sets this bit by writing SetPortPower or SetGlobalPower. HCD clears this bit by writing ClearPortPower or ClearGlobalPower. Which power control switches are enabled is determined by pwr_switch_mode and PortPortControlMask[ndp]. In global switching mode , only Set/ClearGlobalPower controls this bit. In per-port power switching (pwr_switch_mode=1), if the port_pwr_ctrlmask[ndp] bit for the port is set, only Set/ ClearPortPower commands are enabled. If the mask is not set, only Set/ ClearGlobalPower commands are enabled. When port power is disabled, ccs, pes, pss, and prs should be reset. 0: port power is off 1: port power is on (write) SetPortPower. The HCD writes a 1 to set the pps bit. Writing a 0 has no effect.
7: 5	0h RO	**reserved_7_5 (reserved_7_5):** Reserved bits. These bits are reserved and should be set to zero.

Bit Range	Default & Access	Description
4	0b RW/1C	**Port Reset Status / Set Port Reset (prs):** (read) Port Reset Status. When this bit is set by a write to SetPortReset, port reset signaling is asserted. When reset is completed, this bit is cleared when prsc is set. This bit cannot be set if ccs is cleared. 0: port reset signal is not active 1: port reset signal is active (write) SetPortReset. The HCD sets the port reset signaling by writing a 1 to this bit. Writing a 0 has no effect. If ccs is cleared, this write does not set prs, but instead sets csc. This informs the driver that it attempted to reset a disconnected port.
3	0b RW/1C	**Port Over Current Indicator / Clear Suspend Status (poci):** (read) Port Over Current Indicator. This bit is only valid when the Root Hub is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is not supported, this bit is set to 0. If cleared, all power operations are normal for this port. If set an overcurrent condition exists on this port. This bit always reflects the overcurrent input signal 0: no overcurrent condition. 1: overcurrent condition detected. (write) ClearSuspendStatus. The HCD writes a 1 to initiate a resume. Writing a 0 has no effect. A resume is initiated only if pss is set.
2	0b RW/1C	**Port Suspend Status / Set Port Suspend (pss):** Port Suspend Status. This bit indicates the port is suspended or in the resume sequence. It is set by a SetSuspendState write and cleared when pssc is set at the end of the resume interval. This bit cannot be set if ccs is cleared. This bit is also cleared when prsc is set at the end of the port reset or when the HC is placed in the USBRESUME state. If an upstream resume is in progress, it should propagate to the HC. 0: port is not suspended 1: port is suspended (write) SetPortSuspend. The HCD sets the pss bit by writing a 1 to this bit. Writing a 0 has no effect. If ccs is cleared, this write does not set pss; instead it sets csc. This informs the driver that it attempted to suspend a disconnected port.
1	0b RW/1C	**Port Enable Status / Set Port Enable (pes):** (read) Port Enable Status. This bit indicates whether the port is enabled or disabled. The Root Hub may clear this bit when an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected. This change also causes pesc to be set. HCD sets this bit by writing SetPortEnable and clears it by writing ClearPortEnable. This bit cannot be set when ccs is cleared. This bit is also set, if not already, at the completion of a port reset when prsc is set or port suspend when pssc is set. 0: port is disabled 1: port is enabled (write) SetPortEnable. The HCD sets pes by writing a 1. Writing a 0 has no effect. If ccs is cleared, this write does not set pes, but instead sets csc. This informs the driver that it attempted to enable a disconnected port.
0	0b RW/1C	**Current Connect Status / Clear Port Enable (ccs):** (read) Current Connect Status. This bit reflects the current state of the downstream port. 0: no device connected 1: device connected (write) ClearPortEnable The HCD writes a 1 to this bit to clear the pes bit. Writing a 0 has no effect. The ccs is not affected by any write. Note: This bit is always read 1 when the attached device is non-removable (device_removable[ndp]).

§ §



October 2013
Document Number: 329676-001US

17.0 SDIO/SD/eMMC

The Intel® Quark SoC X1000 provides an SDIO/SD/eMMC controller that supports a single port configurable as:

- One SDIO 3.0 interface
- One SD 3.0 interface
- One eMMC 4.41 interface

17.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 92. SDIO/SD/eMMC Interface Signals

Signal Name	Direction/ Type	Description
SD_CLK	O TBD	**SD Card Clock** Clock frequency up to 50 MHz.
SD_DATA[7:0]	I/O TBD	**SD Card Data** Bidirectional port used to transfer data to and from SD/eMMC card. By default, after power up or reset, only D[0] is used for data transfer. A wider data bus can be configured for data transfer, using D[0]-D[7].
SD_CD_B	I TBD	**SD Card Detect** Active low when a card is present. Floating (pulled high with internal PU) when a card is not present.
SD_CMD	I/O MG	**SD Card Command** This signal is used for card initialization and transfer of commands. It has two modes—open-drain for initialization, and push-pull for fast command transfer.
SD_WP	I MG	**SD Card Write Protect** Active high to protect from write.
SD_LED	O MG	**SD Card Access LED Control** Controls the LED to indicate that the card is being accessed
SD_PWR	O MG	**SD Card Power Supply Control**

17.2 Features

17.2.1 SDIO/SD/eMMC Features

Table 93 summarizes the SDIO/SD/eMMC supported/non-supported features.

Table 93. SDIO/SD/eMMC Features

Features	Supported
Meets SD Memory Card Specification version 3.0	Yes
Meets SD Host Controller Standard Specification Version 3.0	Yes
Meets SDIO card specification version 3.0	Yes
Meets eMMC Specification version 4.41	Yes
Supports both DMA and Non-DMA mode of operation	Yes
Supports SDMA	Yes
Supports ADMA1 and ADMA2	Yes
eMMC supports 1 bit, 4 bit and 8 bit bus modes	Yes
SD/SDIO supports 1 bit and 4 bit bus modes	Yes
SDXC Capacity up to 2TB	Yes
High Speed (SD Clock up to 50 MHz - 25 MByte/s for SDIO/SD and 50 MByte/s for eMMC)	Yes
Integrated ADMA Controller	Yes
SD SPI mode	Yes
Multi SD/SDIO card Slots	No
Boot OS from SD/eMMC device	No
SDIO Dual Voltage support 1.8V/3.3V (Supports 3.3V Only)	No
SDIO/SD UHS-I Support	No
Wakeup On Card Insertion	No

17.2.2 SD 3.0/ SDIO 3.0 / eMMC 4.41 Interfaces

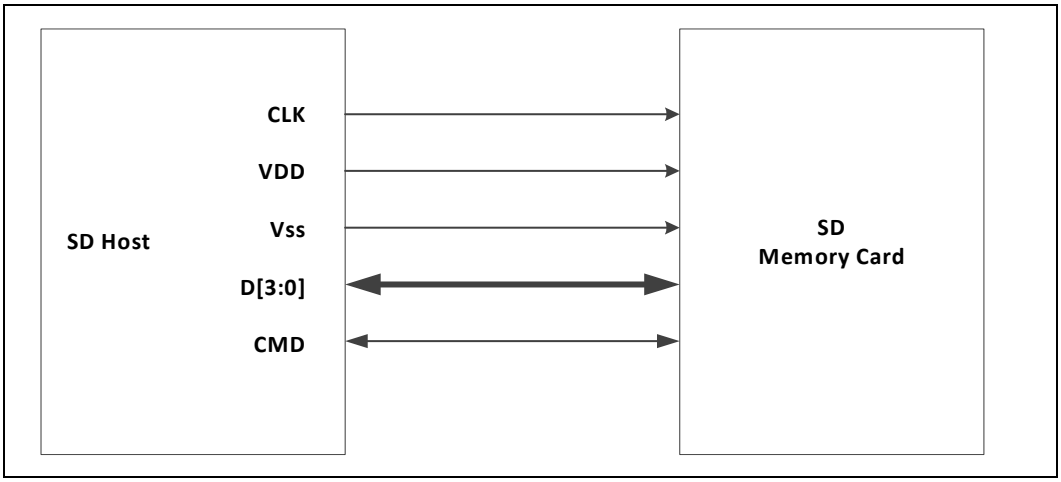
This section provides a very high level overview of the SD, SDIO, eMMC 4.41 specification. Refer to the SD and eMMC specification for complete details.

17.2.2.1 SD 3.0 Bus Topology

The SD Memory Card bus has a single master, single slave (card), synchronous topology (refer to Figure 29). During the initialization process, commands are sent to the card allowing the application to detect the card and assign logical addresses to the physical slot. All data communication in the Card Identification Mode uses the command line (CMD) only.

The SD bus allows dynamic configuration of the number of data lines. After power up, by default, the SD Memory Card uses only D[0] for data transfer. After initialization the host can change the bus width (number of active data lines). This feature allows easy trade off between hardware cost and system performance. Note that while D[1:3] are not in use, the related host's data lines should be in tri-state (input mode).

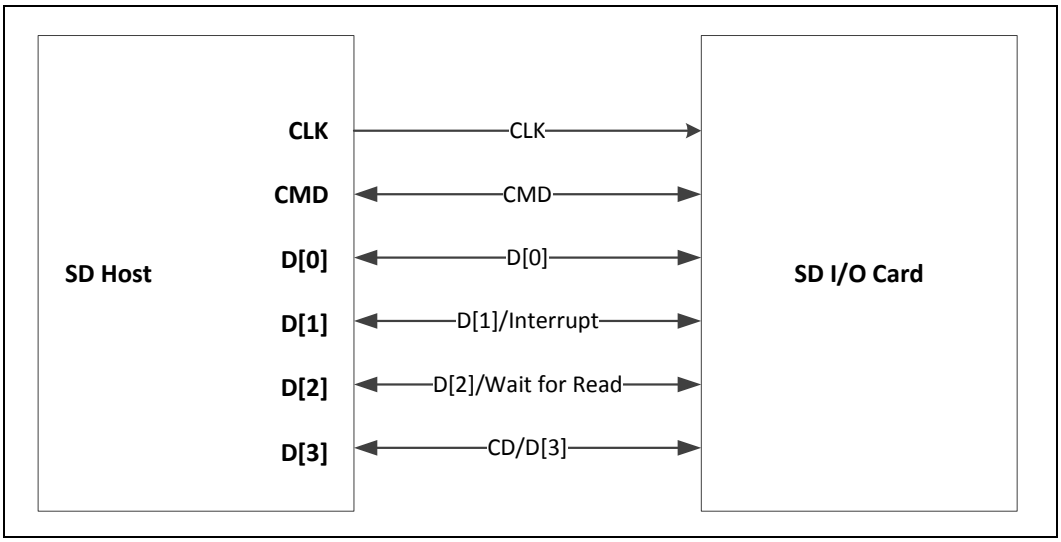
Figure 29. SD Memory Card Bus Topology



17.2.2.2 SDIO 3.0 Interface

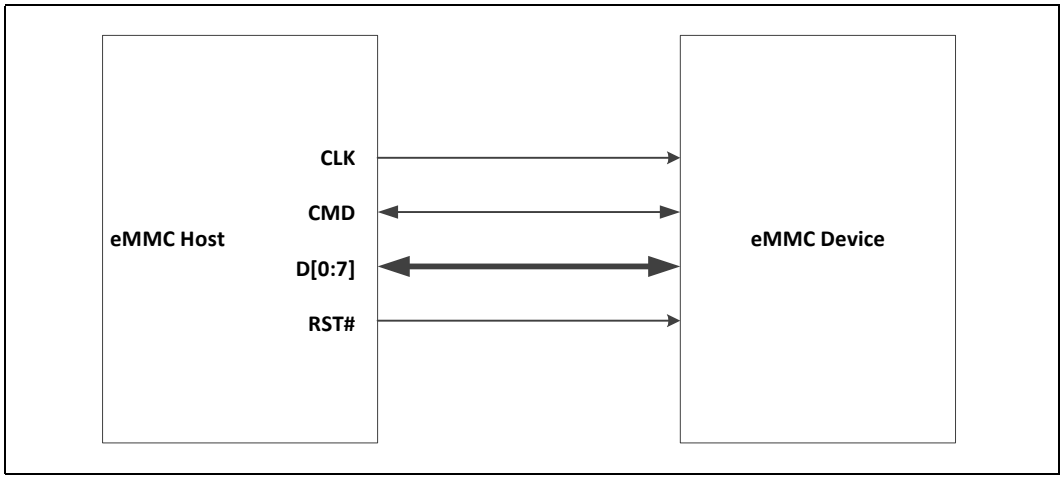
The SDIO card interface is very much like the SD Memory Card interface. The SoC supports one SDIO card slot.

Figure 30. SDIO Card Bus Topology



17.2.2.3 eMMC Interface

Figure 31. eMMC Interface

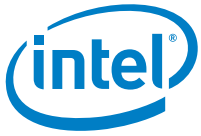


17.2.3 SDIO/SD/eMMC Host Controller

The Secure Digital Host Controller 3.0 specification defines a standardized host controller for interfacing to eMMC devices, as well as Secure Digital memory and I/O cards. The specification encompasses the following:

- A register map and register set
- Data buffer model
- Data movement model, including both Programmed I/O (PIO) and Direct Memory Access (DMA)
- Three different DMA modes:
  - Single DMA (SDMA)
  - Advanced DMA Mode 1; this is linked list DMA with the restriction that data buffers must reside at 4k boundaries.
  - Advanced DMA Mode 2; this is linked list DMA with the restriction of 4k aligned data buffers removed; data buffers may reside at any arbitrary alignment. 64-bit addressing is also supported.
- Interrupt model
- Suspend and resume mechanism
- Power state definition
- Method for automatic generation of STOP_TRANSMISSION commands (Auto CMD12)
- Test registers

In short, the SD Host Controller specification defines a standard software model for accessing SD/SDIO/eMMC devices, and makes it possible for standards-compliant host controllers to work with off-the-shelf device drivers.



17.2.3.1 SD DMA

A new DMA transfer algorithm, called ADMA (Advanced DMA), is defined in the SD Host Controller Standard Specification Version 2.00. The DMA algorithm defined in the SD Host Controller Standard Specification Version 1.00 is called SDMA (Single Operation DMA). SDMA had the disadvantage that a DMA Interrupt generated at every page boundary disturbs the CPU to reprogram the new system address. This SDMA algorithm forms a performance bottleneck by interruption at every page boundary. Only one SD command transaction can be executed per a SDMA operation.

ADMA adopts a scatter-gather DMA algorithm, so that higher data transfer speed is available. The Host Driver can program a list of data transfers between system memory and SD card to the Descriptor Table before executing ADMA. It enables ADMA to operate without interrupting the Host Driver.

There are two types of ADMA; ADMA1 and ADMA2. ADMA1 can support data transfer of only 4 KByte aligned data in system memory. ADMA2 improves the restriction so that data of any location and any size can be transferred in system memory. The format of Descriptor Table is different between them. The Host Controller Specification Ver2.00 defines ADMA2 as standard ADMA and recommends supporting ADMA2 rather than ADMA1.

The SDIO/SD/eMMC controller supports all three flavors of DMA described in the SD Host Controller 2.0 specification - SDMA, ADMA1, and ADMA2.

Note: Although the SD Host Controller Standard Specification Version 3.00 states that ADMA1 is not supported in Standard Host Controller versions 3.0 and latter, the SoC SDIO/SD/eMMC controller supports both ADMA1 and ADMA2.

17.3 References

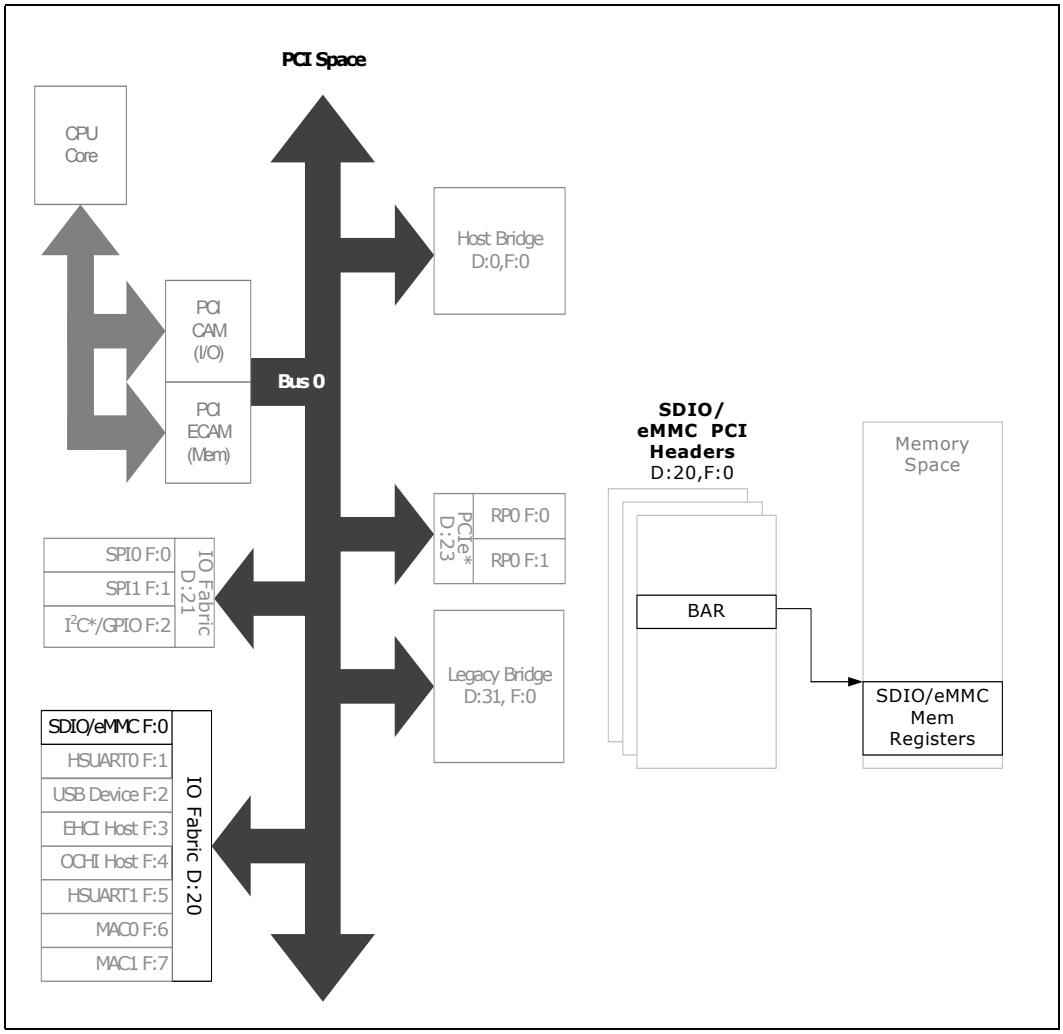
The SDIO/SD/eMMC controller is a Secure Digital I/O (SDIO), Secure Digital (SD), MultiMediaCard (eMMC) host controller that is configured to comply with:

- SD Specification Part 1 Physical Layer Specification version 3.00, April 16, 2009 https://www.sdcard.org
- SD Specification Part E1 SDIO Specification version 3.00, December 16, 2010 https://www.sdcard.org
- SD Specification Part A2 SD Host Controller Standard Specification version 3.00, February 18, 2010 https://www.sdcard.org
- Embedded MultiMediaCard (eMMC) Product Standard v4.41, JESD84-A441 http://www.jedec.org/.

17.4 Register Map

See Chapter 5.0, “Register Access Methods” for additional information.

Figure 32. SDIO/SD/eMMC Register Map



17.5 PCI Configuration Registers

Table 94. Summary of PCI Configuration Registers—0/20/0

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 583	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 584	08A7h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 584	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 585	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 585	08050110h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 586	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 586	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 587	80h

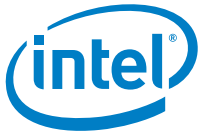


Table 94. Summary of PCI Configuration Registers—0/20/0 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
Fh	Fh	"BIST (BIST)—Offset Fh" on page 587	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 588	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 588	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 589	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 589	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 589	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 590	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 590	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 591	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 591	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 591	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 592	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 592	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 592	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 593	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 594	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 594	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 595	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 595	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 595	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 596	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 596	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 597	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 597	00000000h

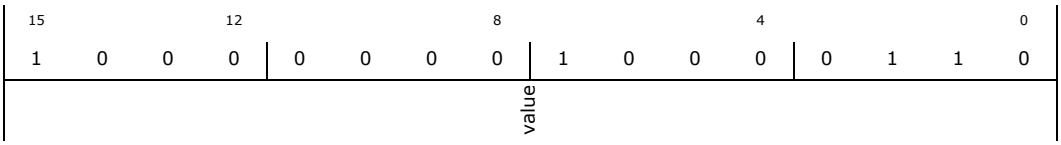
17.5.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:0] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

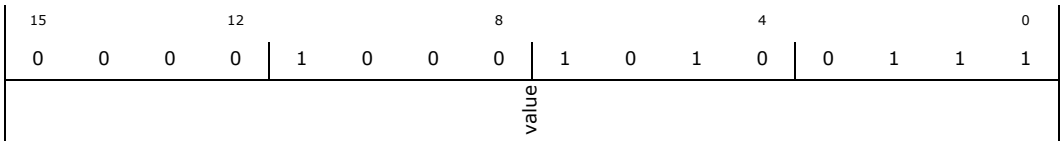
17.5.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:0] + 2h

Default: 08A7h



Bit Range	Default & Access	Description
15: 0	08A7h RO	**Device ID (value):** PCI Device ID

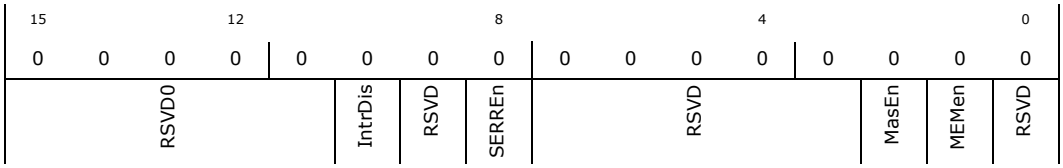
17.5.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:0] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

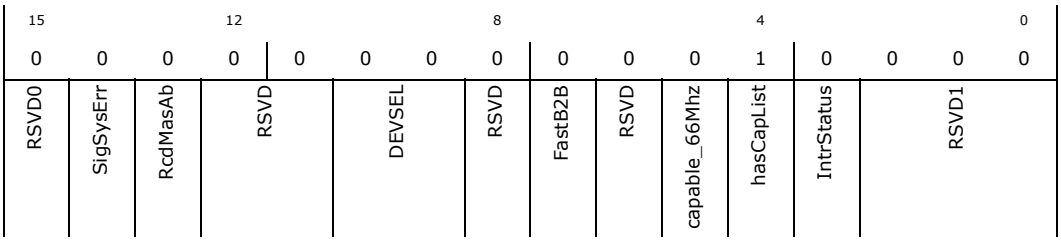
17.5.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:0] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

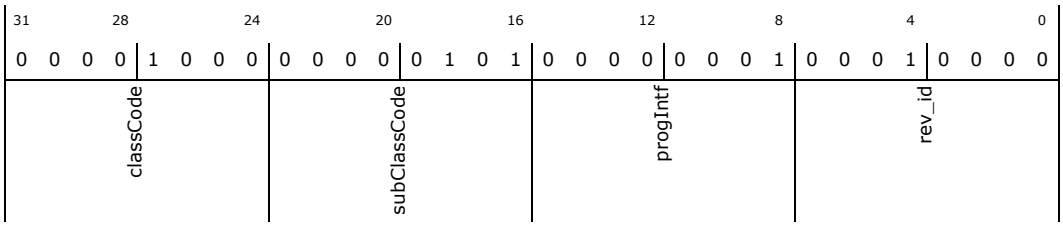
17.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:0] + 8h

Default: 08050110h



Bit Range	Default & Access	Description
31: 24	08h RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	05h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	01h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

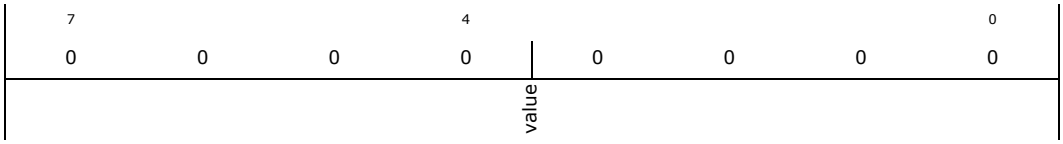
17.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:0] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

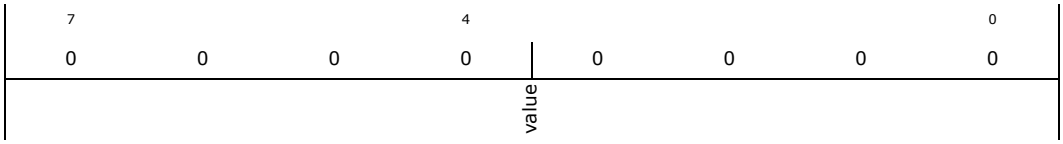
17.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:0] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

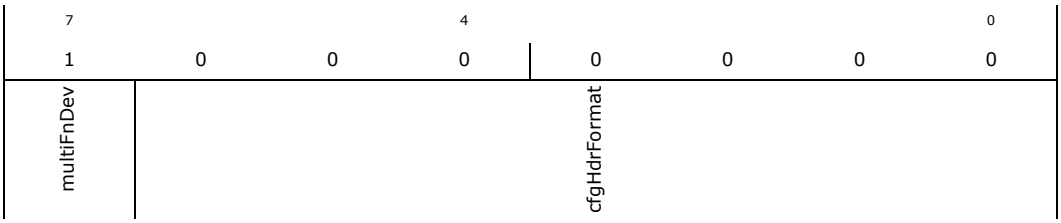
17.5.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:0] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

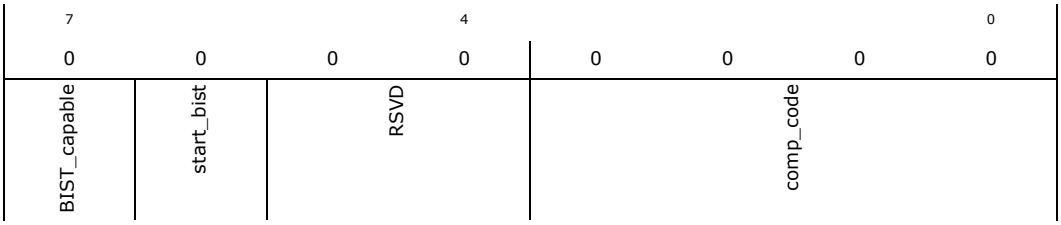
17.5.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:0] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

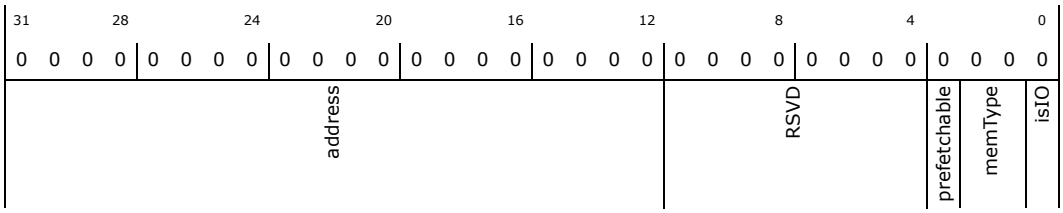
17.5.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

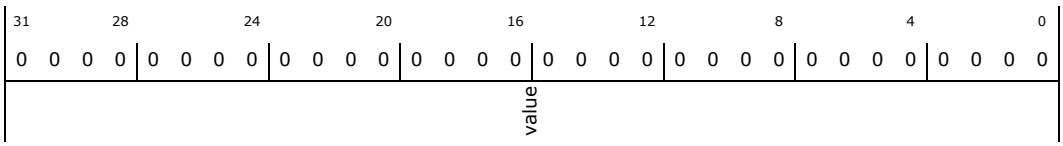
17.5.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:0] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

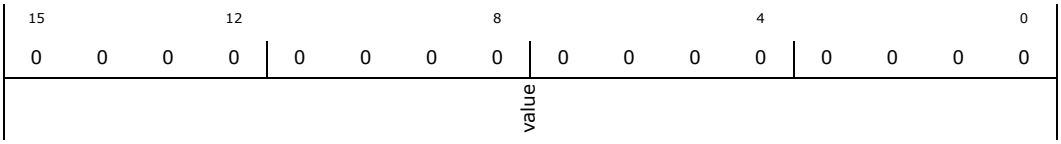
17.5.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:0] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

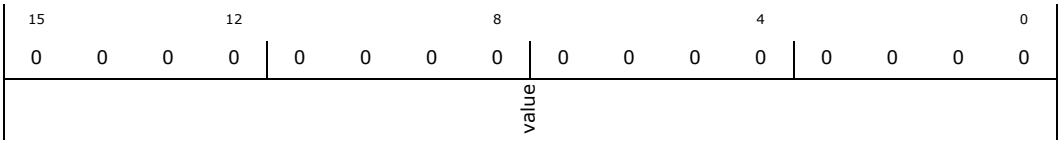
17.5.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:0] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

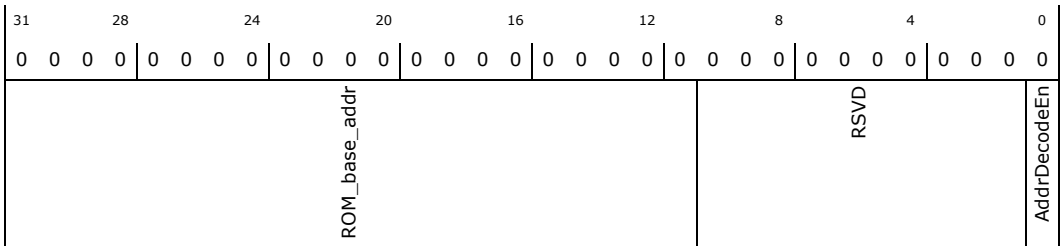
17.5.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:0] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

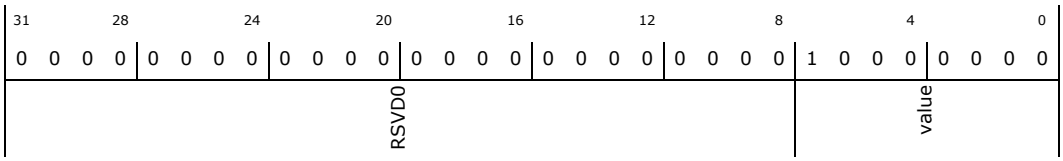
17.5.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:0] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

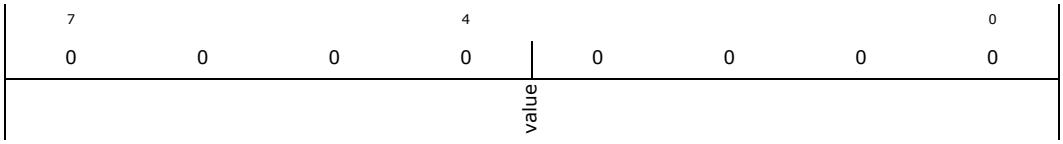
17.5.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:20, F:0] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

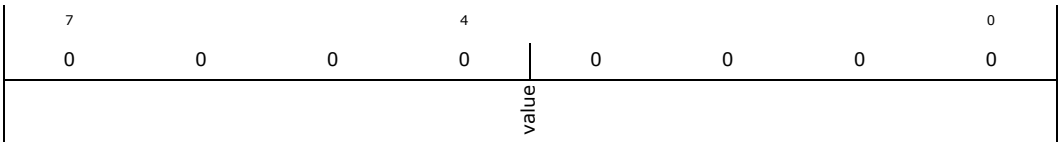
17.5.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:20, F:0] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

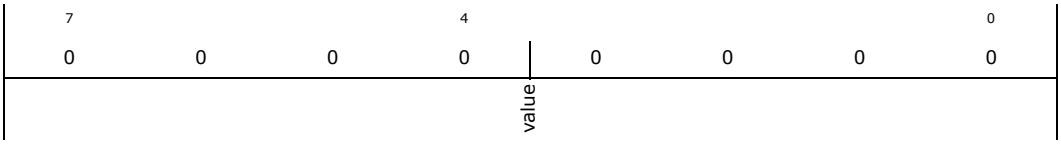
17.5.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:0] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

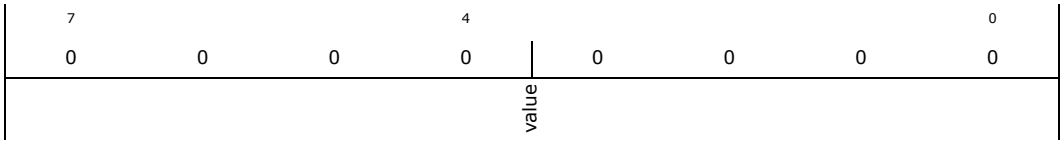
17.5.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:0] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

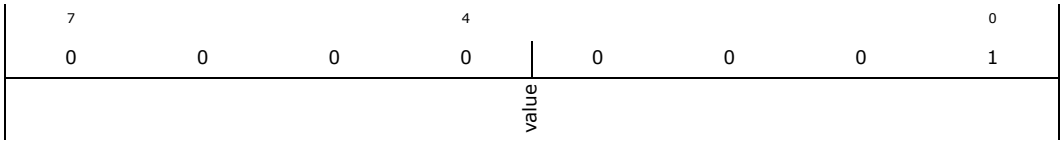
17.5.20 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:0] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

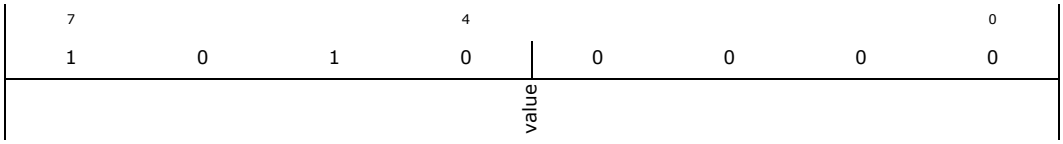
17.5.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:0] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

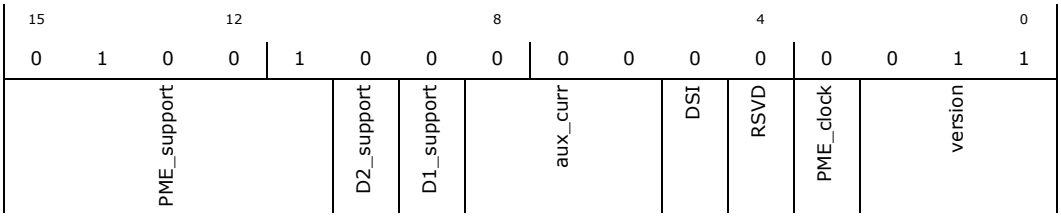
17.5.22 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:0] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

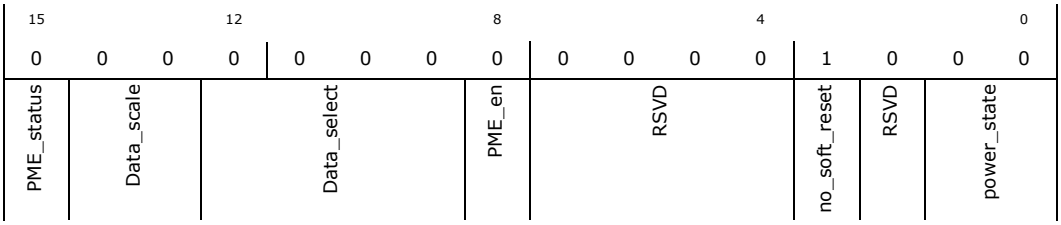
17.5.23 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:0] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported

Bit Range	Default & Access	Description
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

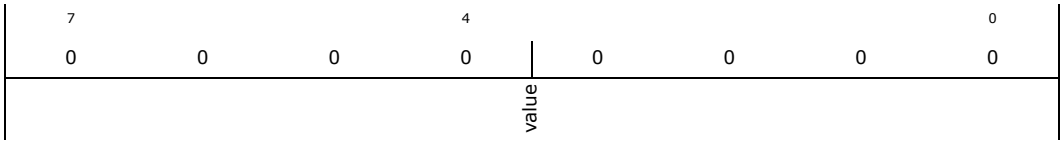
17.5.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:0] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

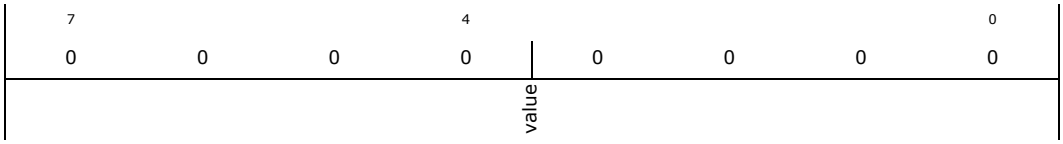
17.5.25 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:0] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

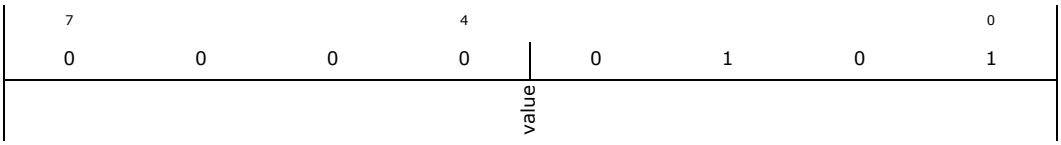
17.5.26 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:0] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

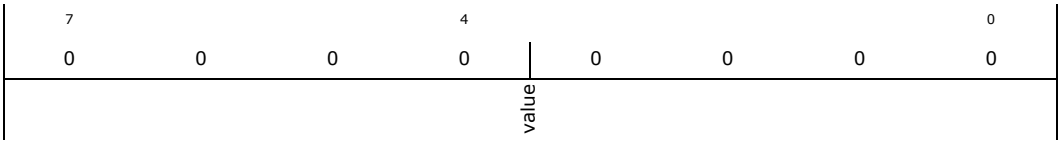
17.5.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:0] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

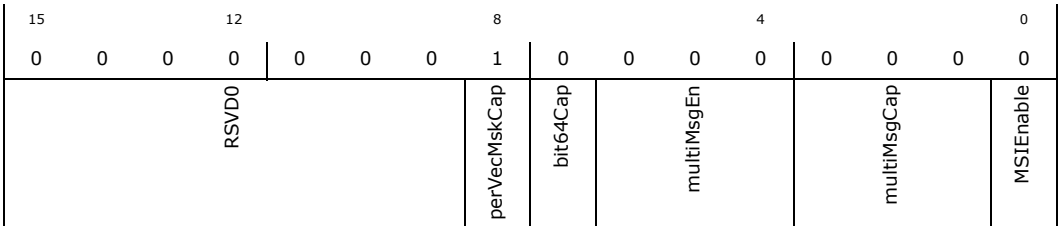
17.5.28 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:0] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

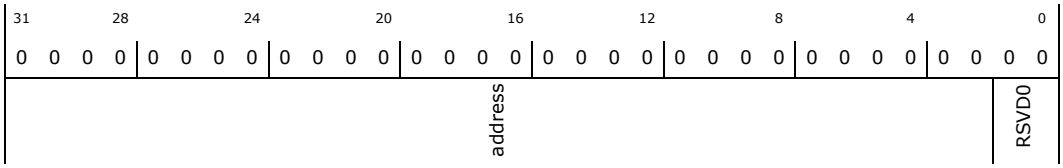
17.5.29 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:0] + A4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

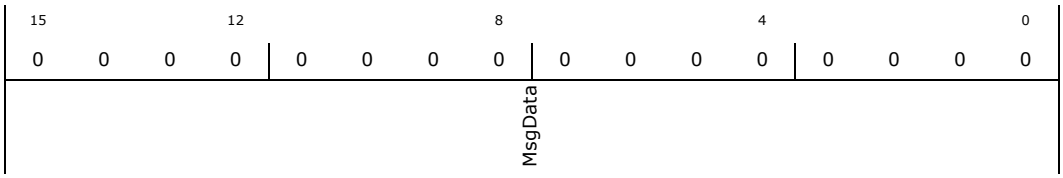
17.5.30 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:0] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

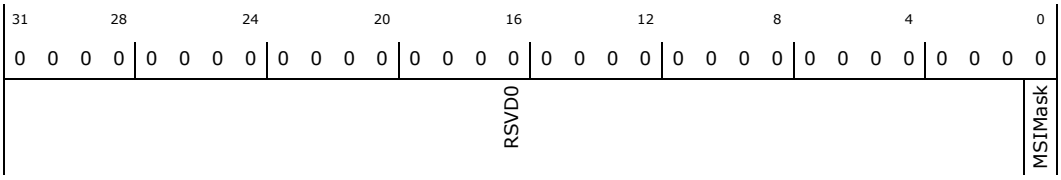
17.5.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:0] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

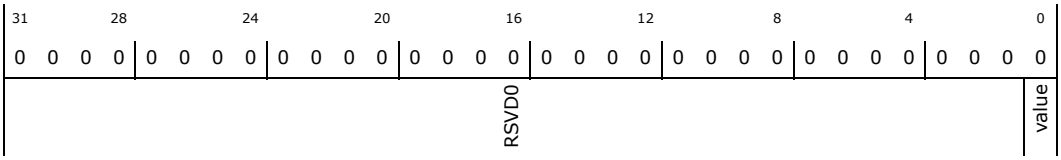
17.5.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:0] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

17.6 Memory Mapped Registers

Table 95. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"SDMA System Address Register (SYS_ADR)—Offset 0h" on page 599	00000000h
4h	5h	"Block Size Register (BLK_SIZE)—Offset 4h" on page 600	0000h
6h	7h	"Block Count Register (BLK_COUNT)—Offset 6h" on page 601	0000h
8h	Bh	"Argument Register (ARGUMENT)—Offset 8h" on page 602	00000000h
Ch	Dh	"Transfer Mode Register (TX_MODE)—Offset Ch" on page 602	0000h
Eh	Fh	"Command Register (CMD)—Offset Eh" on page 604	0000h
10h	13h	"Response Register 0 (RESPONSE0)—Offset 10h" on page 605	00000000h
14h	17h	"Response Register 2 (RESPONSE2)—Offset 14h" on page 606	00000000h
18h	1Bh	"Response Register 4 (RESPONSE4)—Offset 18h" on page 606	00000000h
1Ch	1Fh	"Response Register 6 (RESPONSE6)—Offset 1Ch" on page 607	00000000h
20h	23h	"Buffer Data Port Register (BUF_DATA_PORT)—Offset 20h" on page 607	00000000h
24h	27h	"Present State Register (PRE_STATE)—Offset 24h" on page 608	1FF00000h
28h	28h	"Host Control Register (HOST_CTL)—Offset 28h" on page 612	00h
29h	29h	"Power Control Register (PWR_CTL)—Offset 29h" on page 613	00h
2Ah	2Ah	"Block Gap Control Register (BLK_GAP_CTL)—Offset 2Ah" on page 613	00h
2Bh	2Bh	"Wakeup Control Register (WAKEUP_CTL)—Offset 2Bh" on page 615	00h
2Ch	2Dh	"Clock Control Register (CLK_CTL)—Offset 2Ch" on page 616	0000h
2Eh	2Eh	"Timeout Control Register (TIMEOUT_CTL)—Offset 2Eh" on page 618	00h
2Fh	2Fh	"Software Reset Register (SW_RST)—Offset 2Fh" on page 619	00h
30h	31h	"Normal Interrupt Status Register (NML_INT_STATUS)—Offset 30h" on page 620	0000h
32h	33h	"Error Interrupt Status Register (ERR_INT_STATUS)—Offset 32h" on page 622	0000h
34h	35h	"Normal Interrupt Status Enable (NRM_INT_STATUS_EN)—Offset 34h" on page 624	0000h
36h	37h	"Error Interrupt Status Enable Register (ERR_INT_STAT_EN)—Offset 36h" on page 625	0000h
38h	39h	"Normal Interrupt Signal Enable Register (NRM_INT_SIG_EN)—Offset 38h" on page 626	0000h
3Ah	3Bh	"Error Interrupt Signal Enable Register (ERR_INT_SIG_EN)—Offset 3Ah" on page 628	0000h
3Ch	3Dh	"Auto CMD12 Error Status Register (CMD12_ERR_STAT)—Offset 3Ch" on page 629	0000h
3Eh	3Fh	"Host Control 2 Register (HOST_CTRL_2)—Offset 3Eh" on page 630	0000h
40h	43h	"Capabilities Register (CAPABILITIES)—Offset 40h" on page 631	01EC32B2h
44h	47h	"Capabilities Register 2 (CAPABILITIES_2)—Offset 44h" on page 633	03000000h
48h	4Bh	"Maximum Current Capabilities Register (MAX_CUR_CAP)—Offset 48h" on page 634	00000001h
50h	51h	"Force Event Register for Auto CMD12 Error Status (FORCE_EVENT_CMD12_ERR_STAT)—Offset 50h" on page 635	0000h
52h	53h	"Force Event Register for Error Interrupt Status (FORCE_EVENT_ERR_INT_STAT)—Offset 52h" on page 636	0000h
54h	54h	"ADMA Error Status Register (ADMA_ERR_STAT)—Offset 54h" on page 637	00h

Table 95. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
58h	5Bh	"ADMA System Address Register (ADMA_SYS_ADDR)—Offset 58h" on page 638	00000000h
60h	61h	"initialization Preset Values Register (3.3v or 1.8v) (PRESET_VALUE_0)—Offset 60h" on page 639	0040h
62h	63h	"Default Speed Preset Values Register (PRESET_VALUE_1)—Offset 62h" on page 639	0001h
64h	65h	"High Speed Preset Values Register (PRESET_VALUE_2)—Offset 64h" on page 640	0000h
66h	67h	"SDR12 Preset Values Register (PRESET_VALUE_3)—Offset 66h" on page 641	0001h
68h	69h	"SDR25 Preset Values Register (PRESET_VALUE_4)—Offset 68h" on page 641	0000h
6Ah	6Bh	"SDR50 Preset Values Register (PRESET_VALUE_5)—Offset 6Ah" on page 642	0000h
6Ch	6Dh	"SDR104 Preset Values Register (PRESET_VALUE_6)—Offset 6Ch" on page 643	0000h
6Eh	6Fh	"DDR50 Preset Values Register (PRESET_VALUE_7)—Offset 6Eh" on page 643	0000h
70h	73h	"Boot Time-out control register (BOOT_TIMEOUT_CTRL)—Offset 70h" on page 644	00000000h
74h	74h	"Debug Selection Register (DEBUG_SEL)—Offset 74h" on page 644	00h
E0h	E3h	"Shared Bus Control Register (SHARED_BUS)—Offset E0h" on page 645	00000000h
F0h	F0h	"SPI Interrupt Support Register (SPI_INT_SUP)—Offset F0h" on page 647	00h
FCh	FDh	"Slot Interrupt Status Register (SLOT_INT_STAT)—Offset FCh" on page 647	0000h
FEh	FFh	"Host Controller Version Register (HOST_CTRL_VER)—Offset FEh" on page 648	A702h

17.6.1 SDMA System Address Register (SYS_ADR)—Offset 0h

This register contains the physical system memory address used for DMA transfers or the second argument for the Auto CMD23

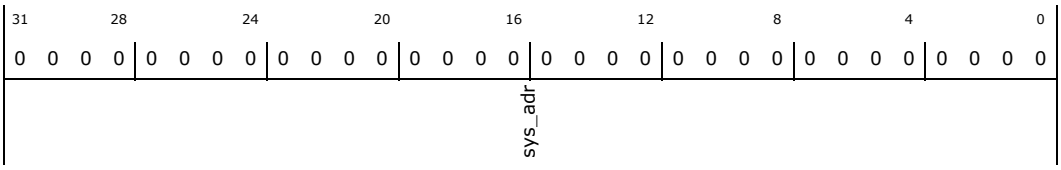
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SYS_ADR: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**SDMA System Address / Auto CMD23 Argument 2 (sys_adr):** This register contains the physical system memory address used for DMA transfers (1) or the second argument for the Auto CMD23 (2). (1) SDMA System Address This register contains the system memory address for a SDMA transfer. When the Host Controller stops a SDMA transfer, this register shall point to the system address of the next contiguous data position. It can be accessed only if no transaction is executing (i.e., after a transaction has stopped). Read operations during transfers may return an invalid value. The Host Driver shall initialize this register before starting a SDMA transaction. After SDMA has stopped, the next system address of the next contiguous data position can be read from this register. The SDMA transfer waits at the every boundary specified by the Host SDMA Buffer Boundary in the Block Size register. The Host Controller generates DMA Interrupt to request the Host Driver to update this register. The Host Driver sets the next system address of the next data position to this register. When the most upper byte of this register (003h) is written, the Host Controller restarts the SDMA transfer. When restarting SDMA by the Resume command or by setting Continue Request in the Block Gap Control register, the Host Controller shall start at the next contiguous address stored here in the SDMA System Address register. ADMA does not use this register (2) Argument 2 This register is used with the Auto CMD23 to set a 32-bit block count value to the argument of the CMD23 while executing Auto CMD23. If Auto CMD23 is used with ADMA, the full 32-bit block count value can be used. If Auto CMD23 is used without AMDA, the available block count value is limited by the Block Count register. 65535 blocks is the maximum value in this case.

17.6.2 Block Size Register (BLK_SIZE)—Offset 4h

This register is used to configure the number of bytes in a data block.

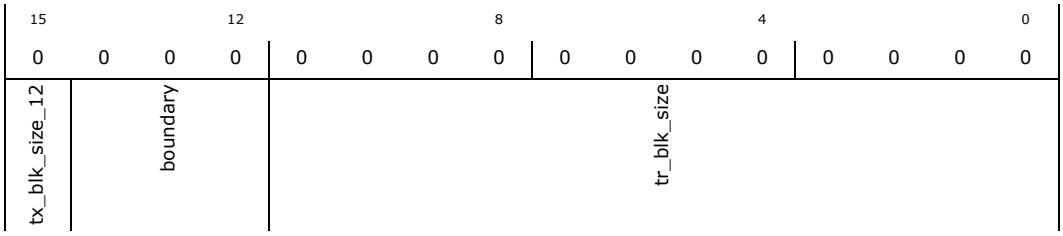
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

BLK_SIZE: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15	0h RW	**Transfer Block Size [12] (tx_blk_size_12):** Transfer Block Size 12th bit. This bit is added to support 4Kb Data block transfer.

Bit Range	Default & Access	Description
14: 12	000b RW	**Host SDMA Buffer Boundary (boundary):** The large contiguous memory space may not be available in the virtual memory system. To perform long SDMA transfer, SDMA System Address register shall be updated at every system memory boundary during SDMA transfer. These bits specify the size of contiguous buffer in the system memory. The SDMA transfer shall wait at the every boundary specified by these fields and the Host Controller generates the DMA Interrupt to request the Host Driver to update the SDMA System Address register. At the end of transfer, the Host Controller may issue or may not issue DMA Interrupt. In particular, DMA Interrupt shall not be issued after Transfer Complete Interrupt is issued. In case of this register is set to 0 (buffer size = 4K bytes), lower 12-bit of byte address points data in the contiguous buffer and the upper 20-bit points the location of the buffer in the system memory. The SDMA transfer stops when the Host Controller detects carry out of the address from bit 11 to 12. These bits shall be supported when the SDMA Support in the Capabilities register is set to 1 and this function is active when the DMA Enable in the Transfer Mode register is set to 1. ADMA does not use this register. 000b 4K bytes (Detects A11 carry out) 001b 8K bytes (Detects A12 carry out) 010b 16K Bytes (Detects A13 carry out) 011b 32K Bytes (Detects A14 carry out) 100b 64K bytes (Detects A15 carry out) 101b 128K Bytes (Detects A16 carry out) 110b 256K Bytes (Detects A17 carry out) 111b 512K Bytes (Detects A18 carry out)
11: 0	000h RW	**Transfer Block Size (tr_blk_size):** This register specifies the block size of data transfers for CMD17, CMD18, CMD24, CMD25, and CMD53. Values ranging from 1 up to the maximum buffer size can be set. In case of memory, it shall be set up to 512 bytes (Refer to SD Host Controller Simplified Specification Version 3.00 - Implementation Note in Section 1.7.2). It can be accessed only if no transaction is executing (i.e., after a transaction has stopped). Read operations during transfers may return an invalid value, and write operations shall be ignored. 0800h 2048 Bytes ... ... 0200h 512 Bytes 01FFh 511 Bytes ... ... 0004h 4 Bytes 0003h 3 Bytes 0002h 2 Bytes 0001h 1 Byte 0000h No data transfer

17.6.3 Block Count Register (BLK_COUNT)—Offset 6h

This register is enabled when Block Count Enable in the Transfer Mode register is set to 1 and is valid only for multiple block transfers.

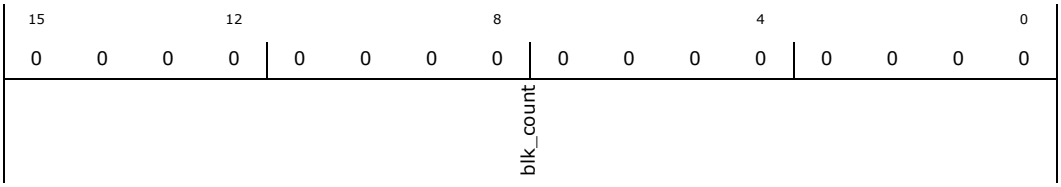
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

BLK_COUNT: [BAR0] + 6h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0000h RW	**Blocks Count For Current Transfer (blk_count):** The Host Driver shall set this register to a value between 1 and the maximum block count. The Host Controller decrements the block count after each block transfer and stops when the count reaches zero. Setting the block count to 0 results in no data blocks is transferred. This register should be accessed only when no transaction is executing (i.e., after transactions are stopped). During data transfer, read operations on this register may return an invalid value and write operations are ignored. When a suspend command is completed, the number of blocks yet to be transferred can be determined by reading this register. Before issuing a resume command, the Host Driver shall restore the previously saved block count. FFFFh 65535 blocks ... ... 0002h 2 blocks 0001h 1 block 0000h Stop Count

17.6.4 Argument Register (ARGUMENT)—Offset 8h

This register contains the SD Command Argument.

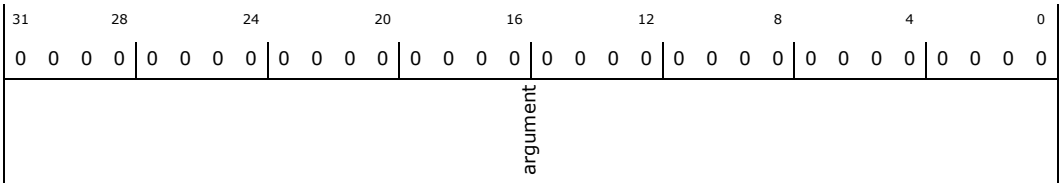
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

ARGUMENT: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**Command Argument 1 (argument):** The SD Command Argument is specified as bit39-8 of Command-Format in the SD Physical Layer Specification.

17.6.5 Transfer Mode Register (TX_MODE)—Offset Ch

This register is used to control the operation of data transfers. The Host Driver shall set this register before issuing a command which transfers data (Refer to Data Present Select in the Command register), or before issuing a Resume command. The Host Driver shall save the value of this register when the data transfer is suspended (as a result of a Suspend command) and restore it before issuing a Resume command. To prevent data loss, the Host Controller shall implement write protection for this register during data transactions. Writes to this register shall be ignored when the Command Inhibit (DAT) in the Present State register is 1.

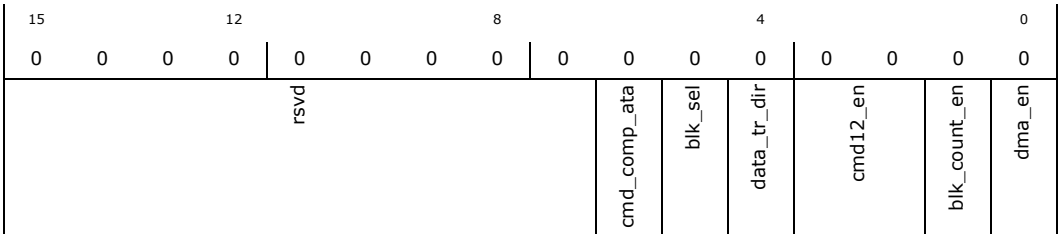
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

TX_MODE: [BAR0] + Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 7	000h RO	**RSVD (rsvd):** Reserved
6	0b RW	**Command Completion Signal Enable for CE-ATA Device (cmd_comp_ata):** 1 - Device will send command completion Signal 0 - Device will not send command completion Signal NOTE: This filed is not part of the SD Host Controller Specification v3.00.
5	0h RW	**Multi / Single Block Select (blk_sel):** This bit is set when issuing multiple-block transfer commands using DAT line. For any other commands, this bit shall be set to 0. If this bit is 0, it is not necessary to set the Block Count register. (Refer to Table 2-8 on SD Host Controller Simplified Specification Version 3.00) 1 Multiple Block 0 Single Block
4	0h RW	**Data Transfer Direction Select (data_tr_dir):** This bit defines the direction of DAT line data transfers. The bit is set to 1 by the Host Driver to transfer data from the SD card to the SD Host Controller and it is set to 0 for all other commands. 1 Read (Card to Host) 0 Write (Host to Card)
3: 2	00b RW	**Auto CMD Enable (cmd12_en):** This field determines use of auto command functions. 00b Auto Command Disabled 01b Auto CMD12 Enable 10b Auto CMD23 Enable 11b Reserved There are two methods to stop Multiple-block read and write operation. (1) Auto CMD12 Enable The Host Controller issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the Auto CMD Error Status register. The Host Driver shall not set this bit if the command does not require CMD12. In particular, secure commands defined in the Part 3 File Security specification do not require CMD12. (2) Auto CMD23 Enable The Host Controller issues a CMD23 automatically before issuing a command specified in the Command Register. The following conditions are required to use the Auto CMD23: - Auto CMD23 Supported - A memory card that supports CMD23 (SCR[33]=1) - If DMA is used, it shall be ADMA. - Only when CMD18 or CMD25 is issued (Note, the Host Controller does not check command index.) Auto CMD23 can be used with or without ADMA. By writing the Command register, the Host Controller issues a CMD23 first and then issues a command specified by the Command Index in Command register. If response errors of CMD23 are detected, the second command is not issued. A CMD23 error is indicated in the Auto CMD Error Status register. 32-bit block count value for CMD23 is set to SDMA System Address / Argument 2 register
1	0b RW	**Block Count Enable (blk_count_en):** This bit is used to enable the Block Count register, which is only relevant for multiple block transfers. When this bit is 0, the Block Count register is disabled, which is useful in executing an infinite transfer. (Refer to Table 2-8 on SD Host Controller Simplified Specification Version 3.00) If ADMA2 data transfer is more than 65535 blocks, this bit shall be set to 0. In this case, data transfer length is designated by Descriptor Table. 1 Enable 0 Disable

Bit Range	Default & Access	Description
0	0b RW	**DMA Enable (dma_en):** DMA can be enabled only if it is supported as indicated in the Capabilities register. One of the DMA modes can be selected by DMA Select in the Host Control 1 register. If DMA is not supported, this bit is meaningless and shall always read 0. If this bit is set to 1, a DMA operation shall begin when the Host Driver writes to the upper byte of Command register (00Fh). 1 DMA Data transfer 0 No data transfer or Non DMA data transfer

17.6.6 Command Register (CMD)—Offset Eh

The Host Driver shall check the Command Inhibit (DAT) bit and Command Inhibit (CMD) bit in the Present State register before writing to this register. Writing to the upper byte of this register triggers SD command generation. The Host Driver has the responsibility to write this register because the Host Controller does not protect for writing when Command Inhibit (CMD) is set.

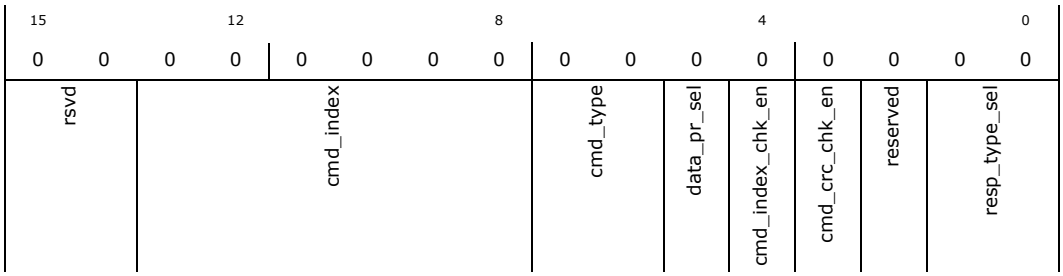
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

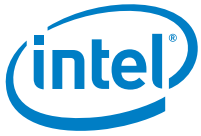
CMD: [BAR0] + Eh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	0h RO	**RSVD (rsvd):** Reserved
13: 8	0h RW	**Command Index (cmd_index):** These bits shall be set to the command number (CMD0-63, ACMD0-63) that is specified in bits 45-40 of the Command-Format in the Physical Layer Specification and SDIO Card Specification



Bit Range	Default & Access	Description
7: 6	00b RW	**Command Type (cmd_type):** There are three types of special commands: Suspend, Resume and Abort. These bits shall be set to 00b for all other commands. (1) Suspend Command If the Suspend command succeeds, the Host Controller shall assume the SD Bus has been released and that it is possible to issue the next command, which uses the DAT line. The Host Controller shall de-assert Read Wait for read transactions and stop checking busy for write transactions. The interrupt cycle shall start, in 4-bit mode. If the Suspend command fails, the Host Controller shall maintain its current state, and the Host Driver shall restart the transfer by setting Continue Request in the Block Gap Control register. (2) Resume Command The Host Driver re-starts the data transfer by restoring the registers in the range of 000-00Dh. The Host Controller shall check for busy before starting write transfers. (3) Abort Command If this command is set when executing a read transfer, the Host Controller shall stop reads to the buffer. If this command is set when executing a write transfer, the Host Controller shall stop driving the DAT line. After issuing the Abort command, the Host Driver should issue a software reset. 11b Abort CMD12, CMD52 for writing I/O Abort in CCCR 10b Resume CMD52 for writing Function Select in CCCR 01b Suspend CMD52 for writing Bus Suspend in CCCR 00b Normal Other commands
5	0b RW	**Data Present Select (data_pr_sel):** This bit is set to 1 to indicate that data is present and shall be transferred using the DAT line. It is set to 0 for the following: (1) Commands using only CMD line (ex. CMD52). (2) Commands with no data transfer but using busy signal on DAT[0] line (R1b or R5b ex. CMD38) (3) Resume command 1 Data Present 0 No Data Present
4	0b RW	**Command Index Check Enable (cmd_index_chk_en):** If this bit is set to 1, the Host Controller shall check the Index field in the response to see if it has the same value as the command index. If it is not, it is reported as a Command Index Error. If this bit is set to 0, the Index field is not checked. 1 Enable 0 Disable
3	0b RW	**Command CRC Check Enable (cmd_crc_chk_en):** If this bit is set to 1, the Host Controller shall check the CRC field in the response. If an error is detected, it is reported as a Command CRC Error. If this bit is set to 0, the CRC field is not checked. The position of CRC field is determined according to the length of the response. (Refer to definition in D01-00 and Table 2-10 in the SD Host Controller Simplified Specification Version 3.00). 1 Enable 0 Disable
2	0h RO	**Reserved (reserved):** Reserved
1: 0	0h RW	**Response Type Select (resp_type_sel):** 00 No Response 01 Response Length 136 10 Response Length 48 11 Response Length 48 check Busy after response

17.6.7 Response Register 0 (RESPONSE0)—Offset 10h

This register is used to store responses from SD cards

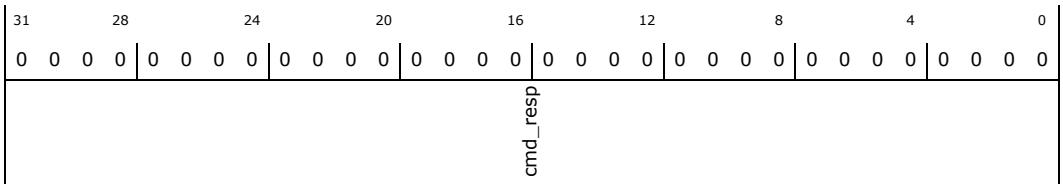
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RESPONSE0: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Command Response (cmd_resp):** Section 2.2.7 and Table 2-12 in the SD Host Controller Simplified Specification Version 3.00 describe the mapping of command responses from the SD Bus to this register for each response type. In the table, R[] refers to a bit range within the response data as transmitted on the SD Bus, REP[] refers to a bit range within the Response register

17.6.8 Response Register 2 (RESPONSE2)—Offset 14h

This register is used to store responses from SD cards

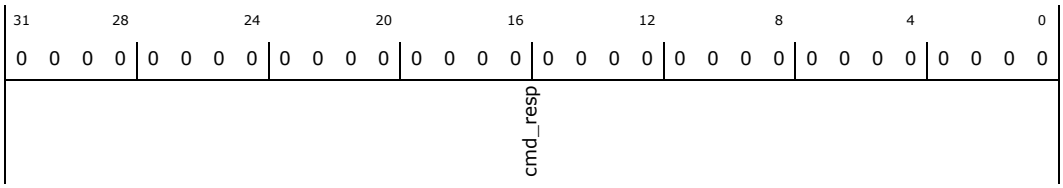
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RESPONSE2: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Command Response (cmd_resp):** Section 2.2.7 and Table 2-12 in the SD Host Controller Simplified Specification Version 3.00 describe the mapping of command responses from the SD Bus to this register for each response type. In the table, R[] refers to a bit range within the response data as transmitted on the SD Bus, REP[] refers to a bit range within the Response register

17.6.9 Response Register 4 (RESPONSE4)—Offset 18h

This register is used to store responses from SD cards

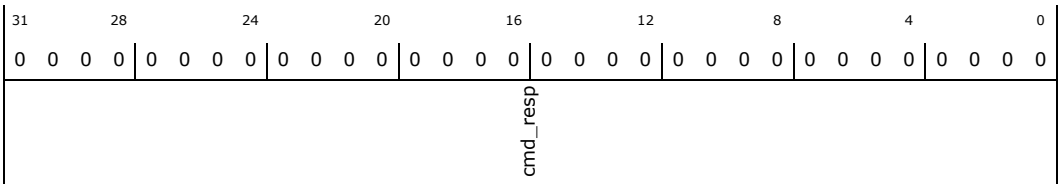
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RESPONSE4: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Command Response (cmd_resp):** Section 2.2.7 and Table 2-12 in the SD Host Controller Simplified Specification Version 3.00 describe the mapping of command responses from the SD Bus to this register for each response type. In the table, R[] refers to a bit range within the response data as transmitted on the SD Bus, REP[] refers to a bit range within the Response register

17.6.10 Response Register 6 (RESPONSE6)—Offset 1Ch

This register is used to store responses from SD cards

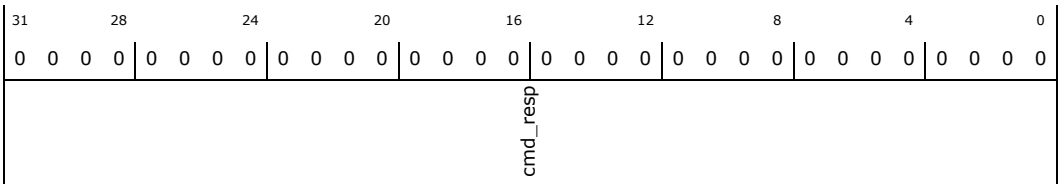
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RESPONSE6: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Command Response (cmd_resp):** Section 2.2.7 and Table 2-12 in the SD Host Controller Simplified Specification Version 3.00 describe the mapping of command responses from the SD Bus to this register for each response type. In the table, R[] refers to a bit range within the response data as transmitted on the SD Bus, REP[] refers to a bit range within the Response register

17.6.11 Buffer Data Port Register (BUF_DATA_PORT)—Offset 20h

32-bit data port register to access internal buffer

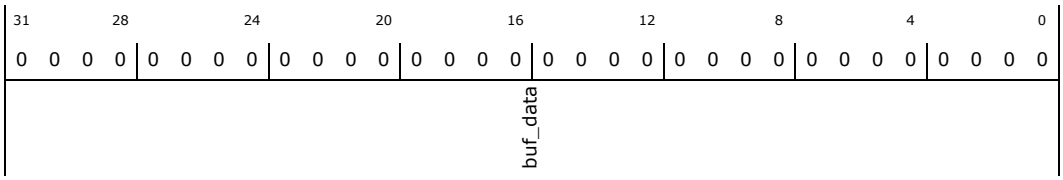
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

BUF_DATA_PORT: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**Buffer Data (buf_data):** The Host Controller buffer can be accessed through this 32-bit Data Port register. Refer to section 1.7 in the SD Host Controller Simplified Specification Version 3.00

17.6.12 Present State Register (PRE_STATE)—Offset 24h

This bit indicates whether one of the DAT line on SD bus is in use. The Host Driver can get status of the Host Controller from this 32-bit read only register

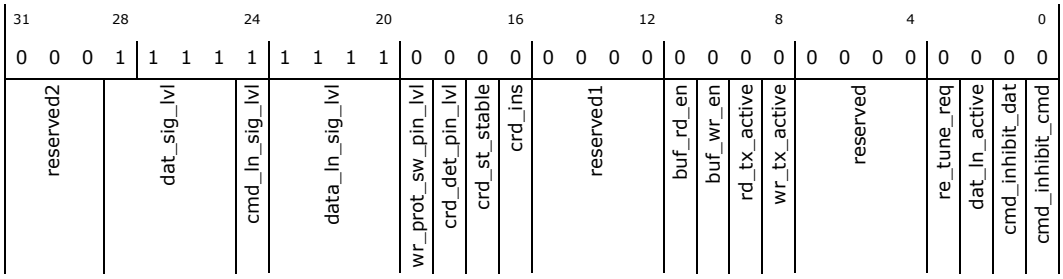
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

PRE_STATE: [BAR0] + 24h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 1FF00000h



Bit Range	Default & Access	Description
31: 29	0h RO	**Reserved2 (reserved2):** Reserved
28: 25	Fh RO	**DAT[7:4] Line Signal Level (dat_sig_lvl):** This status is used to check DAT line level to recover from errors, and for debugging. D28 - DAT[7] D27 - DAT[6] D26 - DAT[5] D25 - DAT[4] NOTE: This filed is not part of the SD Host Controller Specification v3.00.
24	1b RO	**CMD Line Signal Level (cmd_ln_sig_lvl):** This status is used to check the CMD line level to recover from errors, and for debugging.
23: 20	Fh RO	**DAT[3:0] Line Signal Level (data_ln_sig_lvl):** This status is used to check the DAT line level to recover from errors, and for debugging. This is especially useful in detecting the busy signal level from DAT[0]. D23 - DAT[3] D22 - DAT[2] D21 - DAT[1] D20 - DAT[0]

Bit Range	Default & Access	Description
19	0b RO	**Write Protect Switch Pin Level (wr_prot_sw_pin_lvl):** The Write Protect Switch is supported for memory and combo cards. This bit reflects the inverse value of the SD_WP pin. 1 Write enabled (SD_WP=0) 0 Write protected (SD_WP=1)
18	0b RO	**Card Detect Pin Level (crd_det_pin_lvl):** This bit reflects the inverse value of the SD_CD_B pin. Debouncing is not performed on this bit. This bit may be valid when Card State Stable is set to 1, but it is not guaranteed because of propagation delay. Use of this bit is limited to testing since it must be debounced by software. 1 Card present (SD_CD_B=0) 0 No card present (SD_CD_B=1)
17	0b RO	**Card State Stable (crd_st_stable):** This bit is used for testing. If it is 0, the Card Detect Pin Level is not stable. If this bit is set to 1, it means the Card Detect Pin Level is stable. No Card state can be detected by this bit is set to 1 and Card Inserted is set to 0. The Software Reset For All in the Software Reset register shall not affect this bit. 1 No Card or Inserted (stable) 0 Reset or Debouncing (unstable)
16	0b RO	**Card Inserted (crd_ins):** This bit indicates whether a card has been inserted. The Host Controller shall debounce this signal so that the Host Driver will not need to wait for it to stabilize. Changing from 0 to 1 generates a Card Insertion interrupt in the Normal Interrupt Status register and changing from 1 to 0 generates a Card Removal interrupt in the Normal Interrupt Status register. The Software Reset For All in the Software Reset register shall not affect this bit. If a card is removed while its power is on and its clock is oscillating, the Host Controller shall clear SD Bus Power in the Power Control register and SD Clock Enable in the Clock Control register. When this bit is changed from 1 to 0, the Host Controller shall immediately stop driving CMD and DAT[3:0] (tri-state). In addition, the Host Driver should clear the Host Controller by the Software Reset For All in Software Reset register. The card detect is active regardless of the SD Bus Power. 1 Card Inserted 0 Reset or Debouncing or No Card
15: 12	0h RO	**Reserved1 (reserved1):** Reserved
11	0b RO	**Buffer Read Enable (buf_rd_en):** This status is used for non-DMA read transfers. The Host Controller may implement multiple buffers to transfer data efficiently. This read only flag indicates that valid data exists in the host side buffer. If this bit is 1, readable data exists in the buffer. A change of this bit from 1 to 0 occurs when all the block data is read from the buffer. A change of this bit from 0 to 1 occurs when block data is ready in the buffer and generates the Buffer Read Ready interrupt. 1 Read enable 0 Read disable
10	0b RO	**Buffer Write Enable (buf_wr_en):** This status is used for non-DMA write transfers. The Host Controller can implement multiple buffers to transfer data efficiently. This read only flag indicates if space is available for write data. If this bit is 1, data can be written to the buffer. A change of this bit from 1 to 0 occurs when all the block data is written to the buffer. A change of this bit from 0 to 1 occurs when top of block data can be written to the buffer and generates the Buffer Write Ready interrupt. The Host Controller should neither set Buffer Write Enable nor generate Buffer Write Ready Interrupt after the last block data is written to the Buffer Data Port Register. 1 Write enable 0 Write disable
9	0b RO	**Read Transfer Active (rd_tx_active):** This status is used for detecting completion of a read transfer. (Refer to Section 3.12.3 in the SD Host Controller Simplified Specification Version 3.00) This bit is set to 1 for either of the following conditions: (1) After the end bit of the read command. (2) When read operation is restarted by writing a 1 to Continue Request in the Block Gap Control register. This bit is cleared to 0 for either of the following conditions: (1) When the last data block as specified by block length is transferred to the System. (2) In case of ADMA2, end of read operation is designated by Descriptor Table. (3) When all valid data blocks in the Host Controller have been transferred to the System and no current block transfers are being sent as a result of the Stop At Block Gap Request being set to 1. A Transfer Complete interrupt is generated when this bit changes to 0. 1 Transferring data 0 No valid data

Bit Range	Default & Access	Description
8	0b RO	**Write Transfer Active (wr_tx_active):** This status indicates a write transfer is active. If this bit is 0, it means no valid write data exists in the Host Controller. (Refer to Section 3.12.4 in the SD Host Controller Simplified Specification Version 3.00) This bit is set in either of the following cases: (1) After the end bit of the write command. (2) When write operation is restarted by writing a 1 to Continue Request in the Block Gap Control register. This bit is cleared in either of the following cases: (1) After getting the CRC status of the last data block as specified by the transfer count (Single and Multiple) In case of ADMA2, transfer count is designated by Descriptor Table. (2) After getting the CRC status of any block where data transmission is about to be stopped by a Stop At Block Gap Request. During a write transaction, a Block Gap Event interrupt is generated when this bit is changed to 0, as the result of the Stop At Block Gap Request being set. This status is useful for the Host Driver in determining non DAT line commands can be issued during write busy. 1 Transferring data 0 No valid data
7: 4	0h RO	**Reserved (reserved):** Reserved
3	0b RO	**Re-Tuning Request (re_tune_req):** Host Controller may request Host Driver to execute re-tuning sequence by setting this bit when the data window is shifted by temperature drift and a tuned sampling point does not have a good margin to receive correct data. This bit is cleared when a command is issued with setting Execute Tuning in the Host Control 2 register. Changing of this bit from 0 to 1 generates Re-Tuning Event. Refer to Normal Interrupt Status registers for more detail. This bit isn't set to 1 if Sampling Clock Select in the Host Control 2 register is set to 0 (using fixed sampling clock). Refer to Re-Tuning Modes in the Capabilities register for more detail.

Bit Range	Default & Access	Description
2	0b RO	**DAT Line Active (dat_ln_active):** This bit indicates whether one of the DAT line on SD Bus is in use. (a) In the case of read transactions This status indicates whether a read transfer is executing on the SD Bus. Changing this value from 1 to 0 generates a Block Gap Event interrupt in the Normal Interrupt Status register, as the result of the Stop At Block Gap Request being set. Refer to Section 3.12.3 for details on timing. This bit shall be set in either of the following cases: (1) After the end bit of the read command. (2) When writing a 1 to Continue Request in the Block Gap Control register to restart a read transfer. This bit shall be cleared in either of the following cases: (1) When the end bit of the last data block is sent from the SD Bus to the Host Controller. In case of ADMA2, the last block is designated by the last transfer of Descriptor Table. (2) When a read transfer is stopped at the block gap initiated by a Stop At Block Gap Request. The Host Controller shall stop read operation at the start of the interrupt cycle of the next block gap by driving Read Wait or stopping SD clock. If the Read Wait signal is already driven (due to data buffer cannot receive data), the Host Controller can continue to stop read operation by driving the Read Wait signal. It is necessary to support Read Wait in order to use suspend / resume function. (b) In the case of write transactions This status indicates that a write transfer is executing on the SD Bus. Changing this value from 1 to 0 generate a Transfer Complete interrupt in the Normal Interrupt Status register. Refer to Section 3.12.4 for sequence details. This bit shall be set in either of the following cases: (1) After the end bit of the write command. (2) When writing to 1 to Continue Request in the Block Gap Control register to continue a write transfer. This bit shall be cleared in either of the following cases: (1) When the SD card releases write busy of the last data block. If SD card does not drive busy signal for 8 SD Clocks, the Host Controller shall consider the card drive -Not Busy-. In case of ADMA2, the last block is designated by the last transfer of Descriptor Table. (2) When the SD card releases write busy prior to waiting for write transfer as a result of a Stop At Block Gap Request. (c) Command with busy This status indicates whether a command indicates busy (ex. erase command for memory) is executing on the SD Bus. This bit is set after the end bit of the command with busy and cleared when busy is de-asserted. Changing this bit from 1 to 0 generate a Transfer Complete interrupt in the Normal Interrupt Status register. Refer Figure 2-11 to Figure 2-13 on SD Host Controller Simplified Specification Version 3.00. 1 DAT Line Active 0 DAT Line Inactive
1	0b RO	**Command Inhibit (DAT) (cmd_inhibit_dat):** This status bit is generated if either the DAT Line Active or the Read Transfer Active is set to 1. If this bit is 0, it indicates the Host Controller can issue the next SD Command. Commands with busy signal belong to Command Inhibit (DAT) (ex. R1b, R5b type). Changing from 1 to 0 generates a Transfer Complete interrupt in the Normal Interrupt Status register. Note: The SD Host Driver can save registers in the range of 000-00Dh for a suspend transaction after this bit has changed from 1 to 0. 1 Cannot issue command which uses the DAT line 0 Can issue command which uses the DAT line
0	0b RO	**Command Inhibit (CMD) (cmd_inhibit_cmd):** If this bit is 0, it indicates the CMD line is not in use and the Host Controller can issue a SD Command using the CMD line. This bit is set immediately after the Command register (00Fh) is written. This bit is cleared when the command response is received. Auto CMD12 and Auto CMD23 consist of two responses. In this case, this bit is not cleared by the response of CMD12 or CMD23 but cleared by the response of a read/write command. Status issuing Auto CMD12 is not read from this bit. So if a command is issued during Auto CMD12 operation, Host Controller shall manage to issue two commands: CMD12 and a command set by Command register. Even if the Command Inhibit (DAT) is set to 1, commands using only the CMD line can be issued if this bit is 0. Changing from 1 to 0 generates a Command Complete Interrupt in the Normal Interrupt Status register. If the Host Controller cannot issue the command because of a command conflict error or because of Command Not Issued By Auto CMD12 Error (Refer to Section (Refer to Section 2.2.18 and 2.2.23 SD Host Controller Simplified Specification Version 3.00), this bit shall remain 1 and the Command Complete is not set. 1 Cannot issue command 0 Can issue command using only CMD line

17.6.13 Host Control Register (HOST_CTL)—Offset 28h

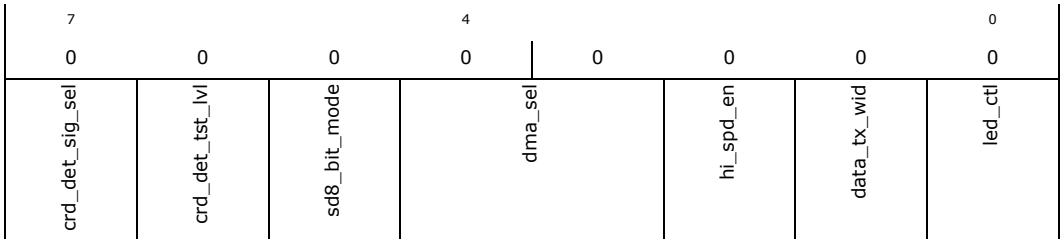
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

HOST_CTL: [BAR0] + 28h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7	0b RW	**Card Detect Signal Selection (crd_det_sig_sel):** This bit selects source for the card detection. When the source for the card detection is switched, the interrupt should be disabled during the switching period by clearing the Interrupt Status/Signal Enable register in order to mask unexpected interrupt being caused by the glitch. The Interrupt Status/Signal Enable should be disabled during over the period of debouncing. 1 The Card Detect Test Level is selected (for test purpose) 0 SDCD# is selected (for normal use)
6	0h RW	**Card Detect Test Level (crd_det_tst_lvl):** This bit is enabled while the Card Detect Signal Selection is set to 1 and it indicates card inserted or not. Generates (card ins or card removal) interrupt when the normal int sts enable bit is set. 1 - Card Inserted 0 - No Card
5	0h RW	**Extended Data Transfer Width (sd8_bit_mode):** This bit controls 8-bit bus width mode for embedded device. Support of this function is indicated in 8-bit Support for Embedded Device in the Capabilities register. If a device supports 8-bit bus mode, this bit may be set to 1. If this bit is 0, bus width is controlled by Data Transfer Width in the Host Control 1 register. This bit is not effective when multiple devices are installed on a bus slot (Slot Type is set to 10b in the Capabilities register). In this case, each device bus width is controlled by Bus Width Preset field in the Shared Bus Control register. 1 8-bit Bus Width 0 Bus Width is Selected by Data Transfer Width
4: 3	00b RW	**DMA Select (dma_sel):** One of supported DMA modes can be selected. The host driver shall check support of DMA modes by referring the Capabilities register. 00 - SDMA is selected 01 - 32-bit Address ADMA1 is selected 10 - 32-bit Address ADMA2 is selected 11 - 64-bit Address ADMA2 is selected NOTE: Codes 01 and 11 are not part of the SD Host Controller Simplified Specification Version 3.00
2	0b RW	**High Speed Enable (hi_spd_en):** This bit is optional. Before setting this bit, the HD shall check the High Speed Support in the capabilities register. If this bit is set to 0 (default), the HC outputs CMD line and DAT lines at the falling edge of the SD clock (up to 25 MHz/ 20MHz for MMC). If this bit is set to 1, the HC outputs CMD line and DAT lines at the rising edge of the SD clock (up to 50 MHz for SD/52MHz for MMC)/ 208Mhz (for SD3.0) If Preset Value Enable in the Host Control 2 register is set to 1, Host Driver needs to reset SD Clock Enable before changing this field to avoid generating clock glitches. After setting this field, the Host Driver sets SD Clock Enable again 1 - High Speed Mode 0 - Normal Speed Mode

Bit Range	Default & Access	Description
1	0h RW	**Data Transfer Width (SD1 or SD4) (data_tx_wid):** This bit selects the data width of the Host Controller. The Host Driver shall set it to match the data width of the SD card. 1 4-bit mode 0 1-bit mode
0	0h RW	**LED Control (led_ctl):** This bit is used to caution the user not to remove the card while the SD card is being accessed. If the software is going to issue multiple SD commands, this bit can be set during all these transactions. It is not necessary to change for each transaction. 1 LED on 0 LED off

17.6.14 Power Control Register (PWR_CTL)—Offset 29h

Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

PWR_CTL: [BAR0] + 29h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
rsvd			hw_rst	sd_bus_volt_sel			sd_bus_pwr

Bit Range	Default & Access	Description
7: 5	0h RO	**RSVD (rsvd):** Reserved
4	0b RW	**HW reset (hw_rst):** Hardware reset signal is generated for eMMC4.4 card when this bit is set. NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
3: 1	0h RW	**SD Bus Voltage Select (sd_bus_volt_sel):** By setting these bits, the Host Driver selects the voltage level for the SD card. Before setting this register, the Host Driver shall check the Voltage Support bits in the Capabilities register. If an unsupported voltage is selected, the Host System shall not supply SD Bus voltage. 111 3.3V (Typ.) 110 3.0V (Typ.) 101 1.8V (Typ.)
0	0b RW	**SD Bus Power (sd_bus_pwr):** Before setting this bit, the SD Host Driver shall set SD Bus Voltage Select. If the Host Controller detects the No Card state, this bit shall be cleared. If this bit is cleared, the Host Controller shall immediately stop driving CMD and DAT[3:0] (tri-state) and drive SDCLK to low level (Refer to Section 2.2.14 of SD Host Controller Simplified Specification Version 3.00). 1 Power on 0 Power off

17.6.15 Block Gap Control Register (BLK_GAP_CTL)—Offset 2Ah

Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

BLK_GAP_CTL: [BAR0] + 2Ah

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
rsvd	alt_boot_en	boot_en	spi_mode	int_blk_gap	rd_wait_ctl	cont_req	stp_blk_gap_req

Bit Range	Default & Access	Description
7	0b RO	**RSVD (rsvd):** Reserved
6	0b RW	**Alternate Boot Mode Enable (alt_boot_en):** To start boot code access in alternative mode. 1 - To start alternate boot mode access 0 - To stop alternate boot mode access NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
5	0b RW	**Boot Enable (boot_en):** To start boot code access 1 - To start boot code access 0 - To stop boot code access NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
4	0b RW	**SPI mode enable (spi_mode):** SPI mode enable bit. 1 - SPI mode 0 - SD mode NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
3	0b RW	**Interrupt At Block Gap (int_blk_gap):** This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. Setting to 0 disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the Host Driver detects an SD card insertion, it shall set this bit according to the CCCR of the SDIO card. 1 Enabled 0 Disabled
2	0b RW	**Read Wait Control (rd_wait_ctl):** The read wait function is optional for SDIO cards. If the card supports read wait, set this bit to enable use of the read wait protocol to stop read data using the DAT[2] line. Otherwise, the Host Controller has to stop the SD Clock to hold read data, which restricts commands generation. When the Host Driver detects an SD card insertion, it shall set this bit according to the CCCR of the SDIO card. If the card does not support read wait, this bit shall never be set to 1 otherwise DAT line conflict may occur. If this bit is set to 0, Suspend/Resume cannot be supported. 1 Enable Read Wait Control 0 Disable Read Wait Control
1	0b RW	**Continue Request (cont_req):** This bit is used to restart a transaction, which was stopped using the Stop At Block Gap Request. To cancel stop at the block gap, set Stop At Block Gap Request to 0 and set this bit 1 to restart the transfer. The Host Controller automatically clears this bit in either of the following cases: (1) In the case of a read transaction, the DAT Line Active changes from 0 to 1 as a read transaction restarts. (2) In the case of a write transaction, the Write Transfer Active changes from 0 to 1 as the write transaction restarts. Therefore, it is not necessary for Host Driver to set this bit to 0. If Stop At Block Gap Request is set to 1, any write to this bit is ignored. 1 Restart 0 Not affect

Bit Range	Default & Access	Description
0	0b RW	**Stop At Block Gap Request (stp_blk_gap_req):** This bit is used to stop executing read and write transaction at the next block gap for non-DMA, SDMA and ADMA transfers. The Host Driver shall leave this bit set to 1 until the Transfer Complete is set to 1. Clearing both Stop At Block Gap Request and Continue Request shall not cause the transaction to restart. When Host Controller version is 1.00, the Host Driver can set this bit if the card supports Read Wait Control. When Host Controller version is 2.00 or later, the Host Driver can set this bit regardless of the card supports Read Wait Control. The Host Controller shall stop read transfer by using Read Wait or stopping SD clock. In case of write transfers in which the Host Driver writes data to the Buffer Data Port register, the Host Driver shall set this bit after all block data is written. If this bit is set to 1, the Host Driver shall not write data to Buffer Data Port register. This bit affects Read Transfer Active, Write Transfer Active, DAT Line Active and Command Inhibit (DAT) in the Present State register. 1 Stop 0 Transfer

17.6.16 Wakeup Control Register (WAKEUP_CTL)—Offset 2Bh

This register is used for wakeup event control.

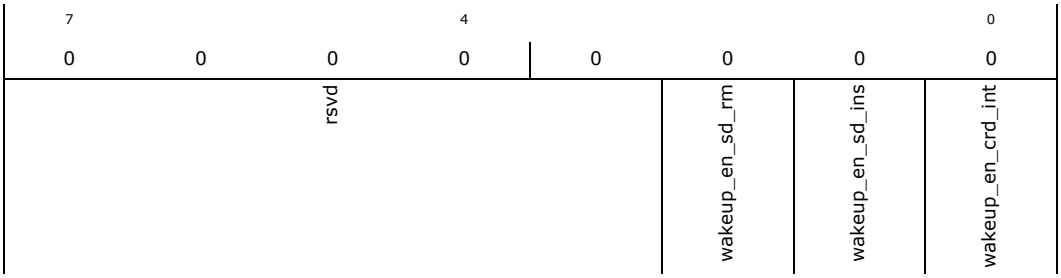
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

WAKEUP_CTL: [BAR0] + 2Bh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7: 3	0h RO	**RSVD (rsvd):** Reserved
2	0h RW	**Wakeup Event Enable On SD Card Removal (wakeup_en_sd_rm):** This bit enables wakeup event via Card Removal assertion in the Normal Interrupt Status register. FN_WUS (Wake Up Support) in CIS does not affect this bit. 1 Enable 0 Disable
1	0h RW	**Wakeup Event Enable On SD Card Insertion (wakeup_en_sd_ins):** This bit enables wakeup event via Card Insertion assertion in the Normal Interrupt Status register. FN_WUS (Wake Up Support) in CIS does not affect this bit. 1 Enable 0 Disable
0	0h RW	**Wakeup Event Enable On Card Interrupt (wakeup_en_crd_int):** This bit enables wakeup event via Card Interrupt assertion in the Normal Interrupt Status register. This bit can be set to 1 if FN_WUS (Wake Up Support) in CIS is set to 1. 1 Enable 0 Disable

17.6.17 Clock Control Register (CLK_CTL)—Offset 2Ch

This register is used configure the frequency of the SDIO controller, and enable the clock.

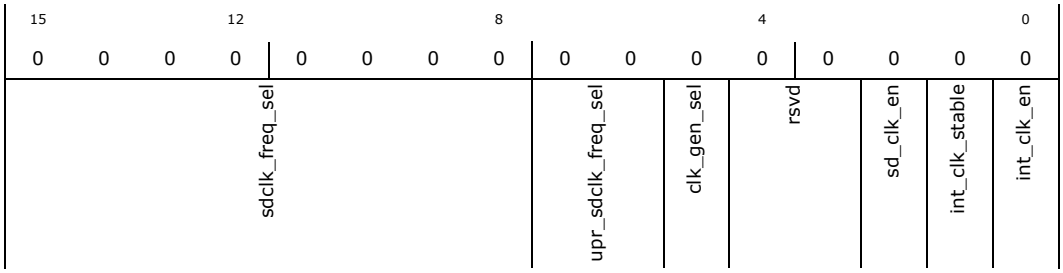
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

CLK_CTL: [BAR0] + 2Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 8	00h RW	**SDCLK Frequency Select (sdclk_freq_sel):** This register is used to select the frequency of the SDCLK pin. The frequency is not programmed directly; rather this register holds the divisor of the Base Clock Frequency For SD clock in the capabilities register. Only the following settings are allowed. (1) 8-bit Divided Clock Mode 80h - base clock divided by 256 40h - base clock divided by 128 20h - base clock divided by 64 10h - base clock divided by 32 08h - base clock divided by 16 04h - base clock divided by 8 02h - base clock divided by 4 01h - base clock divided by 2 00h - base clock(10MHz-63MHz) Setting 00h specifies the highest frequency of the SD Clock. When setting multiple bits, the most significant bit is used as the divisor. But multiple bits should not be set. The two default divider values can be calculated by the frequency that is defined by the Base Clock Frequency For SD Clock in the Capabilities register. - 400KHz divider value - 25MHz divider value - 50MHz divider value The frequency of the SDCLK is set by the following formula: Clock Frequency = (Base clock) / divisor. Thus choose the smallest possible divisor which results in a clock frequency that is less than or equal to the target frequency. Maximum Frequency for SD = 50Mhz (base clock) Maximum Frequency for MMC = 52Mhz (base clock) Minimum Frequency = 195.3125Khz (50Mhz / 256), same clock for MMC also. For example, if the Base Clock Frequency For SD Clock in the Capabilities register has the value 33MHz, and the target frequency is 25MHz, then choosing the divisor value of 01h will yield 16.5MHz, which is the nearest frequency less than or equal to the target. Similarly, to approach a clock value of 400KHz, the divisor value of 40h yields the optimal clock value of 258KHz. (2) 10-bit Divided Clock Mode Host Controller Version 3.00 supports this mandatory mode instead of the 8-bit Divided Clock Mode. The length of divider is extended to10 bits and all divider values shall be supported. 3FFh 1/2046 Divided Clock ... N 1/2N Divided Clock (Duty 50%) ... 002h 1/4 Divided Clock 001h 1/2 Divided Clock 000h Base Clock (10MHz-255MHz) (3) Programmable Clock Mode Host Controller Version 3.00 supports this mode as optional. A non-zero value set to Clock Multiplier in the Capabilities register indicates support of this clock mode. The multiplier enables the Host System to select a finer grain SD clock frequency. It is not necessary to support all frequency generation specified by this field because programmable clock generator is vendor specific and dependent on the implementation. Therefore, this mode is used with Preset Value registers. The Host Controller vendor provides possible settings and the Host System vendor sets appropriate values to the Preset Value registers. 3FFh Base Clock * M / 1024 ... N - 1 Base Clock * M / N ... 002h Base Clock * M / 3 001h Base Clock * M / 2 000h Base Clock * M This field depends on setting of Preset Value Enable in the Host Control 2 register. If Preset Value Enable = 0, this field is set by Host Driver. If the Preset Value Enable = 1, this field is automatically set to a value specified in one of Preset Value registers.
7: 6	00b RW	**Upper Bits of SDCLK Frequency Select (upr_sdclk_freq_sel):** Host Controller Version 1.00 and 2.00 do not support these bits and they are treated as 00b fixed value (ROC). Host Controller Version 3.00 shall support these bits to expand SDCLK Frequency Select to 10-bit. Bit 07-06 is assigned to bit 09-08 of clock divider in SDCLK Frequency Select.

Bit Range	Default & Access	Description
5	0b RW	**Clock Generator Select (clk_gen_sel):** This bit is used to select the clock generator mode in SDCLK Frequency Select. If the Programmable Clock Mode is supported (non-zero value is set to Clock Multiplier in the Capabilities register), this bit attribute is RW, and if not supported, this bit attribute is RO and zero is read. This bit depends on the setting of Preset Value Enable in the Host Control 2 register. If the Preset Value Enable = 0, this bit is set by Host Driver. If the Preset Value Enable = 1, this bit is automatically set to a value specified in one of Preset Value registers. 1 Programmable Clock Mode 0 Divided Clock Mode
4: 3	00b RO	**RSVD (rsvd):** Reserved
2	0b RW	**SD Clock Enable (sd_clk_en):** The Host Controller shall stop SDCLK when writing this bit to 0. SDCLK Frequency Select can be changed when this bit is 0. Then, the Host Controller shall maintain the same clock frequency until SDCLK is stopped (Stop at SDCLK=0). If the Card Inserted in the Present State register is cleared, this bit shall be cleared. 1 Enable 0 Disable
1	0b RO	**Internal Clock Stable (int_clk_stable):** This bit is set to 1 when SD Clock is stable after writing to Internal Clock Enable in this register to 1. The SD Host Driver shall wait to set SD Clock Enable until this bit is set to 1. Note: This is useful when using PLL for a clock oscillator that requires setup time. 1 Ready 0 Not Ready
0	0b RW	**Internal Clock Enable (int_clk_en):** This bit is set to 0 when the Host Driver is not using the Host Controller or the Host Controller awaits a wakeup interrupt. The Host Controller should stop its internal clock to go very low power state. Still, registers shall be able to be read and written. Clock starts to oscillate when this bit is set to 1. When clock oscillation is stable, the Host Controller shall set Internal Clock Stable in this register to 1. This bit shall not affect card detection. 1 Oscillate 0 Stop

17.6.18 Timeout Control Register (TIMEOUT_CTL)—Offset 2Eh

At the initialization of the Host Controller, the Host Driver shall set the Data Timeout Counter Value according to the Capabilities register.

Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

TIMEOUT_CTL: [BAR0] + 2Eh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
reserved				data_timeout_cnt_val			

Bit Range	Default & Access	Description
7: 4	0h RO	**Reserved (reserved):** Reserved
3: 0	0h RW	**Data Timeout Counter Value (data_timeout_cnt_val):** This value determines the interval by which DAT line time-outs are detected. Refer to the Data Time-out Error in the Error Interrupt Status register for information on factors that dictate time-out generation. Time-out clock frequency will be generated by dividing the sdclockTMCLK by this value. When setting this register, prevent inadvertent time-out events by clearing the Data Time-out Error Status Enable (in the Error Interrupt Status Enable register). 1111 - Reserved 1110 - TMCLK * 2^27 --- 0001 - TMCLK * 2^14 0000 - TMCLK * 2^13

17.6.19 Software Reset Register (SW_RST)—Offset 2Fh

A reset pulse is generated when writing 1 to each bit of this register. After completing the reset, the Host Controller shall clear each bit. Because it takes some time to complete software reset, the SD Host Driver shall confirm that these bits are 0.

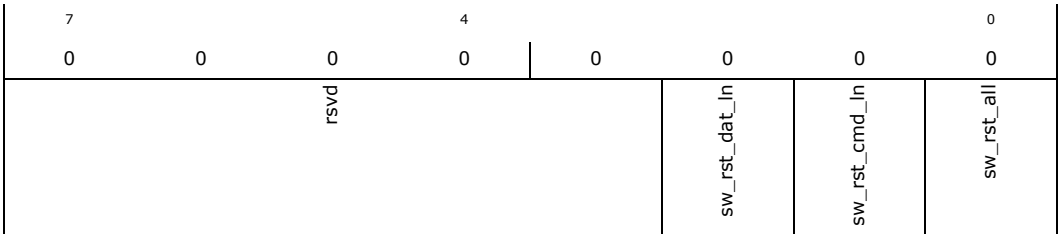
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

SW_RST: [BAR0] + 2Fh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7: 3	0h RO	**RSVD (rsvd):** Reserved

Bit Range	Default & Access	Description
2	0b RW	**Software Reset For DAT Line (sw_rst_dat_ln):** Only part of data circuit is reset. The following registers and bits are cleared by this bit: - Buffer Data Port Register: Buffer is cleared and Initialized. - Present State register Buffer read Enable Buffer write Enable Read Transfer Active Write Transfer Active DAT Line Active Command Inhibit (DAT) - Block Gap Control register Continue Request Stop At Block Gap Request - Normal Interrupt Status register Buffer Read Ready Buffer Write Ready Block Gap Event Transfer Complete 1 - Reset 0 - Work
1	0h RW	**Software Reset For CMD Line (sw_rst_cmd_ln):** Only part of command circuit is reset. The following registers and bits are cleared by this bit: - Present State register Command Inhibit (CMD) - Normal Interrupt Status register Command Complete 1 Reset 0 Work
0	0b RW	**Software Reset For All (sw_rst_all):** This reset affects the entire Host Controller except for the card detection circuit. Register bits of type ROC, RW, RW1C, RWAC are cleared to 0. During its initialization, the Host Driver shall set this bit to 1 to reset the Host Controller. The Host Controller shall reset this bit to 0 when Capabilities registers are valid and the Host Driver can read them. Additional use of Software Reset For All may not affect the value of the Capabilities registers. If this bit is set to 1, the host driver should issue reset command and reinitialize the SD card. 1 Reset 0 Work

17.6.20 Normal Interrupt Status Register (NML_INT_STATUS)—Offset 30h

The Normal Interrupt Status Enable affects reads of this register, but Normal Interrupt Signal Enable does not affect these reads. An interrupt is generated when the Normal Interrupt Signal Enable is enabled and at least one of the status bits is set to 1. Writing 1 to a bit of RW1C attribute clears it; writing 0 keeps the bit unchanged. Writing 1 to a bit of ROC attribute keeps the bit unchanged. More than one status can be cleared with a single register write. The Card Interrupt is cleared when the card stops asserting the interrupt; that is, when the Card Driver services the interrupt condition.

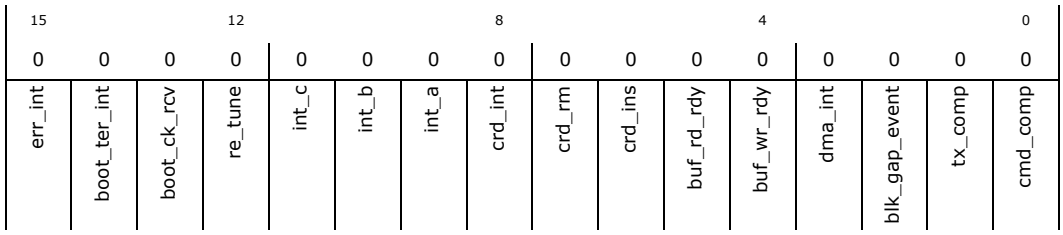
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

NML_INT_STATUS: [BAR0] + 30h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15	0b RO	**Error Interrupt (err_int):** If any of the bits in the Error Interrupt Status Register are set, then this bit is set. Therefore the HD can test for an error by checking this bit first. 0 - No Error. 1 - Error.
14	0h RW/1C	**Boot Terminate Interrupt (boot_ter_int):** This status is set if the boot operation get terminated. 0 - Boot operation is not terminated. 1 - Boot operation is terminated NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
13	0h RW/1C	**Boot Acknowledge Received (boot_ck_rcv):** This status is set if the boot acknowledge is received from device. 0 - Boot ack is not received. 1 - Boot ack is received. NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
12	0h RO	**Re-Tuning Event (re_tune):** This status is set if Re-Tuning Request in the Present State register changes from 0 to 1. Host Controller requests Host Driver to perform re-tuning for next data transfer. Current data transfer (not large block count) can be completed without re-tuning. 1 - Re-Tuning should be performed, 0 - Re-Tuning is not required
11	0h RO	**INT_C (int_c):** This status is set if INT_C is enabled and INT_C# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_C interrupt factor
10	0h RO	**INT_B (int_b):** This status is set if INT_B is enabled and INT_B# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_B interrupt factor
9	0h RO	**INT_A (int_a):** This status is set if INT_A is enabled and INT_A# pin is in low level. Writing this bit to 1 does not clear this bit. It is cleared by resetting the INT_A interrupt factor
8	0b RO	**Card Interrupt (crd_int):** Writing this bit to 1 does not clear this bit. It is cleared by resetting the SD card interrupt factor. In 1-bit mode, the HC shall detect the Card Interrupt without SD Clock to support wakeup. In 4-bit mode, the card interrupt signal is sampled during the interrupt cycle, so there are some sample delays between the interrupt signal from the card and the interrupt to the Host system. when this status has been set and the HD needs to start this interrupt service, Card Interrupt Status Enable in the Normal Interrupt Status register shall be set to 0 in order to clear the card interrupt statuses latched in the HC and stop driving the Host System. After completion of the card interrupt service (the reset factor in the SD card and the interrupt signal may not be asserted), set Card Interrupt Status Enable to 1 and start sampling the interrupt signal again. Interrupt detected by DAT[1] is supported when there is a card per slot. In case of shared bus, interrupt pins are used to detect interrupts. If 000b is set to Interrupt Pin Select in the Shared Bus Control register, this status is effective. Non-zero value is set to Interrupt Pin Select, INT_A, INT_B or INT_C is then used to device interrupts. 0 - No Card Interrupt, 1 - Generate Card Interrupt
7	0b RW/1C	**Card Removal (crd_rm):** This status is set if the Card Inserted in the Present State register changes from 1 to 0. When the HD writes this bit to 1 to clear this status the status of the Card Inserted in the Present State register should be confirmed. Because the card detect may possibly be changed when the HD clear this bit an Interrupt event may not be generated. 0 - Card State Stable or Debouncing, 1 - Card Removed
6	0b RW/1C	**Card Insertion (crd_ins):** This status is set if the Card Inserted in the Present State register changes from 0 to 1. When the HD writes this bit to 1 to clear this status the status of the Card Inserted in the Present State register should be confirmed. Because the card detect may possibly be changed when the HD clear this bit an Interrupt event may not be generated. 0 - Card State Stable or Debouncing, 1 - Card Inserted
5	0b RW/1C	**Buffer Read Ready (buf_rd_rdy):** This status is set if the Buffer Read Enable changes from 0 to 1. Buffer Read Ready is set to 1 for every CMD19 execution in tuning procedure. 0 - Not Ready to read Buffer. 1 - Ready to read Buffer.

Bit Range	Default & Access	Description
4	0b RW/1C	**Buffer Write Ready (buf_wr_rdy):** This status is set if the Buffer Write Enable changes from 0 to 1. 0 - Not Ready to Write Buffer. 1 - Ready to Write Buffer.
3	0b RW/1C	**DMA Interrupt (dma_int):** This status is set if the HC detects the Host DMA Buffer Boundary in the Block Size register. 0 - No DMA Interrupt, 1 - DMA Interrupt is Generated
2	0b RW/1C	**Block Gap Event (blk_gap_event):** If the Stop At Block Gap Request in the Block Gap Control Register is set, this bit is set. Read Transaction: This bit is set at the falling edge of the DAT Line Active Status (When the transaction is stopped at SD Bus timing. The Read Wait must be supported in order to use this function). Write Transaction: This bit is set at the falling edge of Write Transfer Active Status (After getting CRC status at SD Bus timing). 0 - No Block Gap Event, 1 - Transaction stopped at Block Gap
1	0b RW/1C	**Transfer Complete (tx_comp):** This bit is set when a read / write transfer and a command with busy is completed. (1) In the case of a Read Transaction This bit is set at the falling edge of Read Transfer Active Status. This interrupt is generated in two cases. The first is when a data transfer is completed as specified by data length (After the last data has been read to the Host System). The second is when data has stopped at the block gap and completed the data transfer by setting the Stop At Block Gap Request in the Block Gap Control register (After valid data has been read to the Host System). Refer to Section 3.12.3 of SD Host Controller Simplified Specification Version 3.00 for more details on the sequence of events. (2) In the case of a Write Transaction This bit is set at the falling edge of the DAT Line Active Status. This interrupt is generated in two cases. The first is when the last data is written to the SD card as specified by data length and the busy signal released. The second is when data transfers are stopped at the block gap by setting Stop At Block Gap Request in the Block Gap Control register and data transfers completed. (After valid data is written to the SD card and the busy signal released). Refer to Section 3.12.4 for more details on the sequence of events. (3) In the case of a command with busy This bit is set when busy is de-asserted. Refer to DAT Line Active and Command Inhibit (DAT) in the Present State register. Table on page 66 of SD Host Controller Simplified Specification Version 3.00, Relation between Transfer Complete and Data Timeout Error, shows that Transfer Complete has higher priority than Data Timeout Error. If both bits are set to 1, execution of a command can be considered to be completed. 1 Command execution is completed 0 Not complete While performing tuning procedure (Execute Tuning is set to 1), Transfer Complete is not set to 1.
0	0b RW/1C	**Command Complete (cmd_comp):** This bit is set when get the end bit of the command response. Auto CMD12 and Auto CMD23 consist of two responses. Command Complete is not generated by the response of CMD12 or CMD23 but generated by the response of a read/write command. Refer to Command Inhibit (CMD) in the Present State register for how to control this bit. Table on page 67 of SD Host Controller Simplified Specification Version 3.00, Relation between command complete and command time-out error, shows that Command Timeout Error has higher priority than Command Complete. If both bits are set to 1, it can be considered that the response was not received correctly. 1 Command complete 0 No command complete

17.6.21 Error Interrupt Status Register (ERR_INT_STATUS)—Offset 32h

Signals defined in this register can be enabled by the Error Interrupt Status Enable register, but not by the Error Interrupt Signal Enable register. The interrupt is generated when the Error Interrupt Signal Enable is enabled and at least one of the statuses is set to 1. Writing to 1 clears the bit and writing to 0 keeps the bit unchanged. More than one status can be cleared at the one register write.

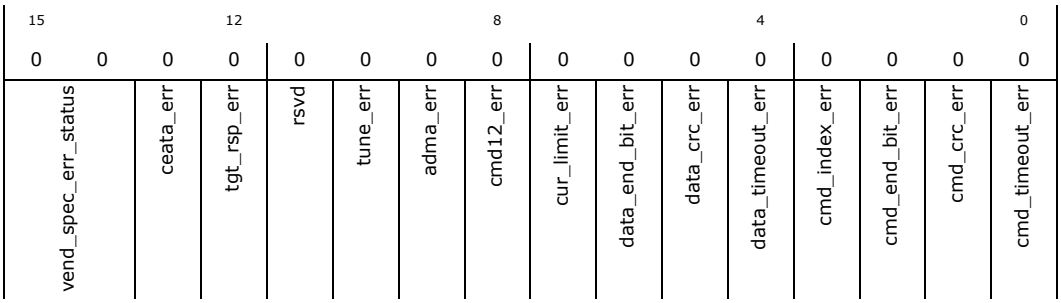
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

ERR_INT_STATUS: [BAR0] + 32h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RW	**Vendor Specific Error Status (vend_spec_err_status):** Reserved
13	0b RW	**CEATA Error Status (ceata_err):** Occurs when ATA command termination has occurred due to an error condition the device has encountered. 0 - no error, 1 - error NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
12	0b RW	**Target Response Error (tgt_rsp_err):** Occurs when detecting ERROR in m_hresp (dma transaction) 0 - no error, 1 - error NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
11	0b RO	**RSVD (rsvd):** Reserved
10	0b RW	**Tuning Error (tune_err):** This bit is set when an unrecoverable error is detected in a tuning circuit except during tuning procedure (Occurrence of an error during tuning procedure is indicated by Sampling Select). By detecting Tuning Error, Host Driver needs to abort a command executing and perform tuning. To reset tuning circuit, Sampling Clock shall be set to 0 before executing tuning procedure. The Tuning Error is higher priority than the other error interrupts generated during data transfer. By detecting Turning Error, the Host Driver should discard data transferred by a current read/write command and retry data transfer after the Host Controller retrieved from tuning circuit error. 1 - Error, 0 - No Error
9	0b RW	**ADMA Error (adma_err):** This bit is set when the Host Controller detects errors during ADMA based data transfer. The state of the ADMA at an error occurrence is saved in the ADMA Error Status Register. 1 - Error, 0 - No Error
8	0b RW	**Auto CMD Error (cmd12_err):** Auto CMD12 and Auto CMD23 use this error status. This bit is set when detecting that one of the bits D00-D04 in Auto CMD Error Status register has changed from 0 to 1. In case of Auto CMD12, this bit is set to 1, not only when the errors in Auto CMD12 occur but also when Auto CMD12 is not executed due to the previous command error. 0 - No Error, 1 - Error
7	0b RW	**Current Limit Error (cur_limit_err):** By setting the SD Bus Power bit in the Power Control Register, the HC is requested to supply power for the SD Bus. If the HC supports the Current Limit Function, it can be protected from an Illegal card by stopping power supply to the card in which case this bit indicates a failure status. Reading 1 means the HC is not supplying power to SD card due to some failure. Reading 0 means that the HC is supplying power and no error has occurred. This bit shall always set to be 0, if the HC does not support this function. 0 - No Error, 1 - Power Fail
6	0b RW	**Data End Bit Error (data_end_bit_err):** Occurs when detecting 0 at the end bit position of read data which uses the DAT line or the end bit position of the CRC status. 0 - No Error, 1 - Error
5	0b RW	**Data CRC Error (data_crc_err):** Occurs when detecting CRC error when transferring read data which uses the DAT line or when detecting the Write CRC Status having a value of other than 010. 0 - No Error, 1 - Error

Bit Range	Default & Access	Description
4	0b RW	**Data Timeout Error (data_timeout_err):** Occurs when detecting one of following timeout conditions. 1. Busy Time-out for R1b, R5b type. 2. Busy Time-out after Write CRC status 3. Write CRC status Time-out 4. Read Data Time-out 0 - No Error, 1 - Timeout
3	0b RW	**Command Index Error (cmd_index_err):** Occurs if a Command Index error occurs in the Command Response. 0 - No Error, 1 - Error
2	0b RW	**Command End Bit Error (cmd_end_bit_err):** Occurs when detecting that the end bit of a command response is 0. 0 - No Error, 1 - End Bit Error Generated
1	0b RW	**Command CRC Error (cmd_crc_err):** Command CRC Error is generated in two cases. 1. If a response is returned and the Command Time-out Error is set to 0, this bit is set to 1 when detecting a CRT error in the command response 2. The HC detects a CMD line conflict by monitoring the CMD line when a command is issued. If the HC drives the CMD line to 1 level, but detects 0 level on the CMD line at the next SDCLK edge, then the HC shall abort the command (Stop driving CMD line) and set this bit to 1. The Command Time-out Error shall also be set to 1 to distinguish CMD line conflict. 0 - No Error, 1 - CRC Error Generated
0	0b RW	**Command Timeout Error (cmd_timeout_err):** Occurs only if the no response is returned within 64 SDCLK cycles from the end bit of the command. If the HC detects a CMD line conflict, in which case Command CRC Error shall also be set. This bit shall be set without waiting for 64 SDCLK cycles because the command will be aborted by the HC. 0 - No Error, 1 - Timeout

17.6.22 Normal Interrupt Status Enable (NRM_INT_STATUS_EN)—Offset 34h

Setting to 1 enables Interrupt Status.

Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

NRM_INT_STATUS_EN: [BAR0] + 34h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
fixed_0	boot_ter_int_stat_en	boot_ack_rcv_stat_en	re_tune_stat_en	int_c_stat_en	int_b_stat_en	int_a_stat_en	crd_int_stat_en	crd_rm_stat_en	crd_ins_stat_en	buf_rd_rdy_stat_en	buf_wr_rdy_stat_en	dma_int_stat_en	blk_gap_event_stat_en	tx_comp_stat_en	cmd_comp_stat_en

Bit Range	Default & Access	Description
15	0h RO	**Fixed to 0 (fixed_0):** The HC shall control error Interrupts using the Error Interrupt Status Enable register.
14	0h RW	**Boot Terminate Interrupt Status Enable (boot_ter_int_stat_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
13	0h RW	**Boot ACK Receive Enable (boot_ack_rcv_stat_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00

Bit Range	Default & Access	Description
12	0h RW	**Re-Tuning Event Status Enable (re_tune_stat_en):** 0 - Masked, 1 - Enabled
11	0h RW	**INT_C Status Enable (int_c_stat_en):** If this bit is set to 0, the Host Controller shall clear the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_C and may set this bit again after all interrupt requests to INT_C pin are cleared to prevent inadvertent interrupts.
10	0h RW	**INT_B Status Enable (int_b_stat_en):** If this bit is set to 0, the Host Controller shall clear the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_B and may set this bit again after all interrupt requests to INT_B pin are cleared to prevent inadvertent interrupts.
9	0h RW	**INT_A Status Enable (int_a_stat_en):** If this bit is set to 0, the Host Controller shall clear the interrupt request to the System. The Host Driver may clear this bit before servicing the INT_A and may set this bit again after all interrupt requests to INT_A pin are cleared to prevent inadvertent interrupts.
8	0b RW	**Card Interrupt Status Enable (crd_int_stat_en):** If this bit is set to 0, the HC shall clear Interrupt request to the System. The Card Interrupt detection is stopped when this bit is cleared and restarted when this bit is set to 1. The HD may clear the Card Interrupt Status Enable before servicing the Card Interrupt and may set this bit again after all Interrupt requests from the card are cleared to prevent inadvertent Interrupts. 0 - Masked, 1 - Enabled
7	0h RW	**Card Removal Status Enable (crd_rm_stat_en):** 0 - Masked, 1 - Enabled
6	0h RW	**Card Insertion Status Enable (crd_ins_stat_en):** 0 - Masked, 1 - Enabled
5	0h RW	**Buffer Read Ready Status Enable (buf_rd_rdy_stat_en):** 0 - Masked, 1 - Enabled
4	0h RW	**Buffer Write Ready Status Enable (buf_wr_rdy_stat_en):** 0 - Masked, 1 - Enabled
3	0h RW	**DMA Interrupt Status Enable (dma_int_stat_en):** 0 - Masked, 1 - Enabled
2	0h RW	**Block Gap Event Status Enable (blk_gap_event_stat_en):** 0 - Masked, 1 - Enabled
1	0h RW	**Transfer Complete Status Enable (tx_comp_stat_en):** 0 - Masked, 1 - Enabled
0	0h RW	**Command Complete Status Enable (cmd_comp_stat_en):** 0 - Masked, 1 - Enabled

17.6.23 Error Interrupt Status Enable Register (ERR_INT_STAT_EN)—Offset 36h

Setting to 1 enables Interrupt Status.

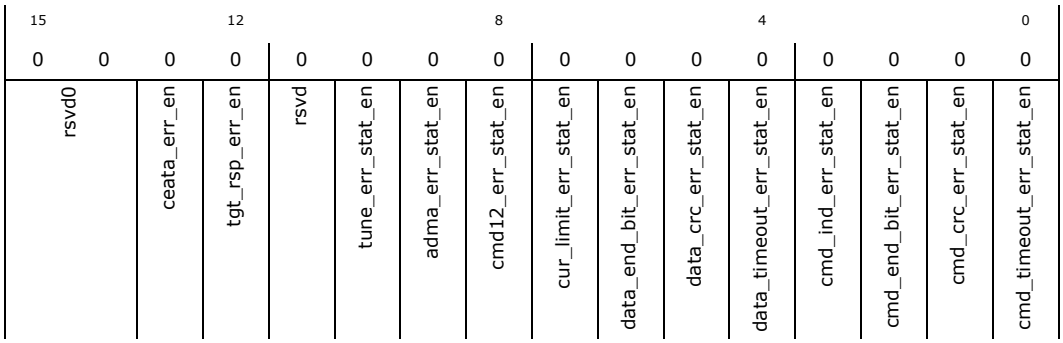
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

ERR_INT_STAT_EN: [BAR0] + 36h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	0b RO	**Vendor Specific Error Status Enable (rsvd0):** Reserved
13	0b RW	**CEATA Error Status Enable (ceata_err_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
12	0b RW	**Target Response Error Status Enable (tgt_rsp_err_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
11	0h RO	**RSVD (rsvd):** Reserved
10	0h RW	**Tuning Error Status Enable (tune_err_stat_en):** 0 - Masked, 1 - Enabled
9	0h RW	**ADMA Error Status Enable (adma_err_stat_en):** 0 - Masked, 1 - Enabled
8	0h RW	**Auto CMD12 Error Status Enable (cmd12_err_stat_en):** 0 - Masked, 1 - Enabled
7	0h RW	**Current Limit Error Status Enable (cur_limit_err_stat_en):** 0 - Masked, 1 - Enabled
6	0h RW	**Data End Bit Error Status Enable (data_end_bit_err_stat_en):** 0 - Masked, 1 - Enabled
5	0h RW	**Data CRC Error Status Enable (data_crc_err_stat_en):** 0 - Masked, 1 - Enabled
4	0h RW	**Data Timeout Error Status Enable (data_timeout_err_stat_en):** 0 - Masked, 1 - Enabled
3	0h RW	**Command Index Error Status Enable (cmd_ind_err_stat_en):** 0 - Masked, 1 - Enabled
2	0h RW	**Command End Bit Error Status Enable (cmd_end_bit_err_stat_en):** 0 - Masked, 1 - Enabled
1	0h RW	**Command CRC Error Status Enable (cmd_crc_err_stat_en):** 0 - Masked, 1 - Enabled
0	0h RW	**Command Timeout Error Status Enable (cmd_timeout_err_stat_en):** 0 - Masked, 1 - Enabled

17.6.24 Normal Interrupt Signal Enable Register (NRM_INT_SIG_EN)—Offset 38h

This register is used to select which interrupt status is indicated to the Host System as the interrupt. These status bits all share the same1 bit interrupt line. Setting any of these bits to 1 enables interrupt generation.

Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

NRM_INT_SIG_EN: [BAR0] + 38h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
fixed_0	boot_ter_int_sig_en	boot_ack_rcv_sig_en	re_tune_sig_en	int_c_sig_en	int_b_sig_en	int_a_sig_en	crd_int_sig_en	crd_rm_sig_en	crd_ins_sig_en	buf_rd_rdy_sig_en	buf_wr_rdy_sig_en	dma_int_sig_en	blk_gap_event_sig_en	tx_comp_sig_en	cmd_comp_sig_en

Bit Range	Default & Access	Description
15	0h RO	**Fixed to 0 (fixed_0):** The HD shall control error Interrupts using the Error Interrupt Signal Enable register.
14	0h RW	**Boot Terminate Interrupt Signal Enable (boot_ter_int_sig_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
13	0h RW	**Boot ACK Received Signal Enable (boot_ack_rcv_sig_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
12	0h RW	**Re-Tuning Event Signal Enable (re_tune_sig_en):** 0 - Masked, 1 - Enabled
11	0h RW	**INT_C Signal Enable (int_c_sig_en):** 0 - Masked, 1 - Enabled
10	0h RW	**INT_B Signal Enable (int_b_sig_en):** 0 - Masked, 1 - Enabled
9	0h RW	**INT_A Signal Enable (int_a_sig_en):** Reserved.
8	0h RW	**Card Interrupt Signal Enable (crd_int_sig_en):** 0 - Masked, 1 - Enabled
7	0h RW	**Card Removal Signal Enable (crd_rm_sig_en):** 0 - Masked, 1 - Enabled
6	0h RW	**Card Insertion Signal Enable (crd_ins_sig_en):** 0 - Masked, 1 - Enabled
5	0h RW	**Buffer Read Ready Signal Enable (buf_rd_rdy_sig_en):** 0 - Masked, 1 - Enabled
4	0h RW	**Buffer Write Ready Signal Enable (buf_wr_rdy_sig_en):** 0 - Masked, 1 - Enabled
3	0h RW	**DMA Interrupt Signal Enable (dma_int_sig_en):** 0 - Masked, 1 - Enabled
2	0h RW	**Block Gap Event Signal Enable (blk_gap_event_sig_en):** 0 - Masked, 1 - Enabled
1	0h RW	**Transfer Complete Signal Enable (tx_comp_sig_en):** 0 - Masked, 1 - Enabled

Bit Range	Default & Access	Description
0	0h RW	**Command Complete Signal Enable (cmd_comp_sig_en):** 0 - Masked, 1 - Enabled

17.6.25 Error Interrupt Signal Enable Register (ERR_INT_SIG_EN)—Offset 3Ah

This register is used to select which interrupt status is notified to the Host System as the interrupt. These status bits all share the same 1 bit interrupt line. Setting any of these bits to 1 enables interrupt generation.

Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

ERR_INT_SIG_EN: [BAR0] + 3Ah

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rsvd0		ceata_err_sig_en	tgt_err_rsp_sig_en	rsvd	tune_err_sig	adma_err_sig_en	cmd12_err_sig_en	cur_limit_err_sig_en	data_end_bit_err_sig_en	data_crc_err_sig_en	data_timeout_err_stat_en	cmd_ind_err_stat_en	cmd_end_bit_err_stat_en	cmd_crc_err_stat_en	cmd_timeout_err_stat_en

Bit Range	Default & Access	Description
15: 14	0b RO	**Vendor Specific Error Signal Enable (rsvd0):** Reserved
13	0b RW	**CEATA Error Signal Enable (ceata_err_sig_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
12	0b RW	**Target Response Error Signal Enable (tgt_err_rsp_sig_en):** 0 - Masked, 1 - Enabled NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
11	0h RO	**RSVD (rsvd):** Reserved
10	0h RW	**Tuning Error Signal Enable (tune_err_sig):** 0 - Masked, 1 - Enabled
9	0h RW	**ADMA Error Signal Enable (adma_err_sig_en):** 0 - Masked, 1 - Enabled
8	0h RW	**Auto CMD12 Error Signal Enable (cmd12_err_sig_en):** 0 - Masked, 1 - Enabled
7	0h RW	**Current Limit Error Signal Enable (cur_limit_err_sig_en):** 0 - Masked, 1 - Enabled
6	0h RW	**Data End Bit Error Signal Enable (data_end_bit_err_sig_en):** 0 - Masked, 1 - Enabled

Bit Range	Default & Access	Description
5	0h RW	**Data CRC Error Signal Enable (data_crc_err_sig_en):** 0 - Masked, 1 - Enabled
4	0h RW	**Data Timeout Error Signal Enable (data_timeout_err_stat_en):** 0 - Masked, 1 - Enabled
3	0h RW	**Command Index Error Signal Enable (cmd_ind_err_stat_en):** 0 - Masked, 1 - Enabled
2	0h RW	**Command End Bit Error Signal Enable (cmd_end_bit_err_stat_en):** 0 - Masked, 1 - Enabled
1	0h RW	**Command CRC Error Signal Enable (cmd_crc_err_stat_en):** 0 - Masked, 1 - Enabled
0	0h RW	**Command Timeout Error Signal Enable (cmd_timeout_err_stat_en):** 0 - Masked, 1 - Enabled

17.6.26 Auto CMD12 Error Status Register (CMD12_ERR_STAT)—Offset 3Ch

This register is used to indicate CMD12 response error of Auto CMD12 and CMD23 response error of Auto CMD23. The Host driver can determine what kind of Auto CMD12 / CMD23 errors occur by this register. Auto CMD23 errors are indicated in bit 04-01.This register is valid only when the Auto CMD Error is set.

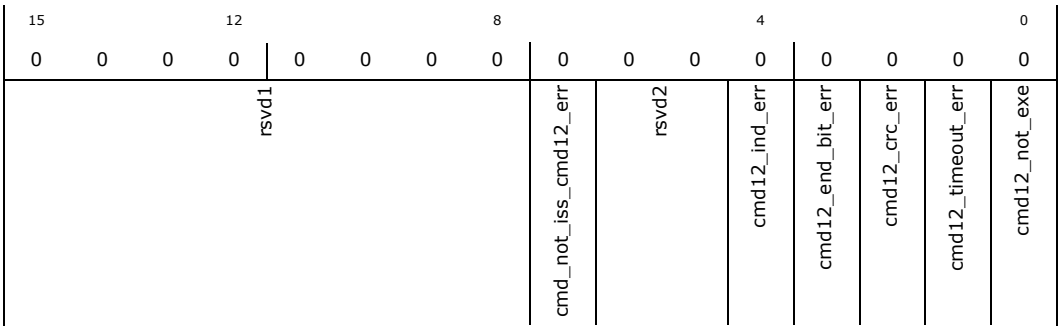
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

CMD12_ERR_STAT: [BAR0] + 3Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 8	0h RO	**RSVD1 (rsvd1):** Reserved
7	0b RO	**Command Not Issued By Auto CMD12 Error (cmd_not_iss_cmd12_err):** CMD_wo_DAT is not executed due to an Auto CMD12 error (D04 - D01) in this register. This bit is set to 0 when Auto CMD Error is generated by Auto CMD23. 0 - No Error, 1 - Not Issued
6: 5	0h RO	**RSVD2 (rsvd2):** Reserved

Bit Range	Default & Access	Description
4	0b RO	**Auto CMD Index Error (cmd12_ind_err):** Occurs if the Command Index error occurs in response to a command. 0 - No Error, 1 - Error
3	0b RO	**Auto CMD End Bit Error (cmd12_end_bit_err):** Occurs when detecting that the end bit of command response is 0. 0 - No Error, 1 - End Bit Error Generated
2	0b RO	**Auto CMD CRC Error (cmd12_crc_err):** Occurs when detecting a CRC error in the command response. 0 - No Error, 1 - CRC Error Generated
1	0b RO	**Auto CMD Timeout Error (cmd12_timeout_err):** Occurs if the no response is returned within 64 SDCLK cycles from the end bit of the command. If this bit is set to 1, the other error status bits (D04 - D02) are meaningless. 0 - No Error, 1 - Timeout
0	0b RO	**Auto CMD12 Not Executed (cmd12_not_exe):** If memory multiple block data transfer is not started due to command error, this bit is not set because it is not necessary to issue Auto CMD12. Setting this bit to 1 means the HC cannot issue Auto CMD12 to stop memory multiple block transfer due to some error. If this bit is set to 1, other error status bits (D04 - D01) are meaningless. This bit is set to 0 when Auto CMD Error is generated by Auto CMD23. 0 - Executed, 1 - Not Executed

17.6.27 Host Control 2 Register (HOST_CTRL_2)—Offset 3Eh

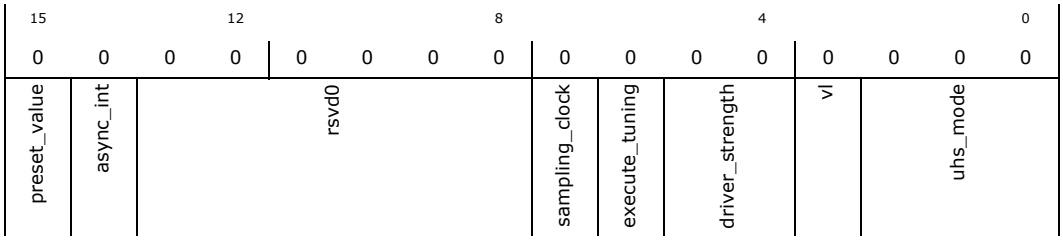
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

HOST_CTRL_2: [BAR0] + 3Eh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15	0b RW	**Preset Value Enable (preset_value):** Host Controller Version 3.00 supports this bit. As the operating SDCLK frequency and I/O driver strength depend on the Host System implementation, it is difficult to determine these parameters in the Standard Host Driver. When Preset Value Enable is set to automatic This bit enables the functions defined in the Preset Value registers. 1 Automatic Selection by Preset Value are Enabled 0 SDCLK and Driver Strength are controlled by Host Driver If this bit is set to 0, SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select in Host Control 2 register are set by Host Driver. If this bit is set to 1, SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select in Host Control 2 register are set by Host Controller as specified in the Preset Value registers.
14	0b RW	**Asynchronous Interrupt Enable (async_int):** This bit can be set to 1 if a card support asynchronous interrupt and Asynchronous Interrupt Support is set to 1 in the Capabilities register. Asynchronous interrupt is effective when DAT[1] interrupt is used in 4-bit SD mode (and zero is set to Interrupt Pin Select in the Shared Bus Control register). If this bit is set to 1, the Host Driver can stop the SDCLK during asynchronous interrupt period to save power. During this period, the Host Controller continues to deliver Card Interrupt to the host when it is asserted by the Card. 1 Enabled, 0 Disabled
13: 8	00h RO	**RSVD0 (rsvd0):** Reserved

Bit Range	Default & Access	Description
7	0b RW	**Sampling Clock Select (sampling_clock):** This bit is set by tuning procedure when Execute Tuning is cleared. Writing 1 to this bit is meaningless and ignored. Setting 1 means that tuning is completed successfully and setting 0 means that tuning is failed. Host Controller uses this bit to select sampling clock to receive CMD and DAT. This bit is cleared by writing 0. Change of this bit is not allowed while the Host Controller is receiving response or a read data block. 1 Tuned clock is used to sample data, 0 Fixed clock is used to sample data
6	0b RW/AC	**Execute Tuning (execute_tuning):** This bit is set to 1 to start tuning procedure and automatically cleared when tuning procedure is completed. The result of tuning is indicated to Sampling Clock Select. Tuning procedure is aborted by writing 0 for more detail about tuning procedure. 1 Execute Tuning, 0 Not Tuned or Tuning Completed
5: 4	0b RW	**Driver Strength Select (driver_strength):** Host Controller output driver in 1.8V signaling is selected by this bit. In 3.3V signaling, this field is not effective. This field can be set depends on Driver Type A, C and D support bits in the Capabilities register. This bit depends on setting of Preset Value Enable. If Preset Value Enable = 0, this field is set by Host Driver. If Preset Value Enable = 1, this field is automatically set by a value specified in the one of Preset Value registers. 00b Driver Type B is Selected (Default), 01b Driver Type A is Selected, 10b Driver Type C is Selected, 11b Driver Type D is Selected
3	0b RW	**1.8V Signaling Enable (vl):** This bit controls voltage regulator for I/O cell. 3.3V is supplied to the card regardless of signaling voltage. Setting this bit from 0 to 1 starts changing signal voltage from 3.3V to 1.8V. 1.8V regulator output shall be stable within 5ms. Host Controller clears this bit if switching to 1.8V signaling fails. Clearing this bit from 1 to 0 starts changing signal voltage from 1.8V to 3.3V. 3.3V regulator output shall be stable within 5ms. Host Driver can set this bit to 1 when Host Controller supports 1.8V signaling (One of support bits is set to 1: SDR50, SDR104 or DDR50 in the Capabilities register) and the card or device supports UHS-I. 1 - 1.8V Signaling, 0 - 3.3V Signaling
2: 0	0b RW	**UHS Mode Select (uhs_mode):** This field is used to select one of UHS-I modes and effective when 1.8V Signaling Enable is set to 1. If Preset Value Enable in the Host Control 2 register is set to 1, Host Controller sets SDCLK Frequency Select, Clock Generator Select in the Clock Control register and Driver Strength Select according to Preset Value registers. In this case, one of preset value registers is selected by this field. Host Driver needs to reset SD Clock Enable before changing this field to avoid generating clock glitch. After setting this field, Host Driver sets SD Clock Enable again. 000b - SDR12, 001b - SDR25, 010b - SDR50, 011b - SDR104, 100b - DDR50, 101b - 111 Reserved. When SDR50, SDR104 or DDR50 is selected for SDIO card, interrupt detection at the block gap shall not be used. Read Wait timing is changed for these modes. Refer to the SDIO Specification Version 3.00 for more detail.

17.6.28 Capabilities Register (CAPABILITIES)—Offset 40h

This register provides the Host Driver with information specific to the Host Controller implementation. The Host Controller may implement these values as fixed or loaded from flash memory during power on initialization. Refer to Software Reset For All in the Software Reset register for loading from flash memory and completion timing control.

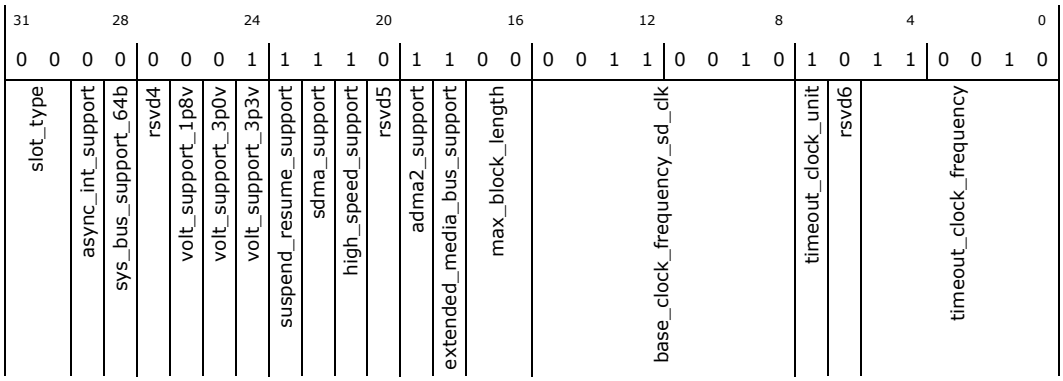
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

CAPABILITIES: [BAR0] + 40h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 01EC32B2h



Bit Range	Default & Access	Description
31: 30	0b RO	**Slot Type (slot_type):** This field indicates usage of a slot by a specific Host System. (A host controller register set is defined per slot.) Embedded slot for one device (01b) means that only one non-removable device is connected to a SD bus slot. Shared Bus Slot (10b) can be set if Host Controller supports Shared Bus Control register. The Standard Host Driver controls only a removable card or one embedded device is connected to a SD bus slot. If a slot is configured for shared bus (10b), the Standard Host Driver does not control embedded devices connected to a shared bus. Shared bus slot is controlled by a specific host driver developed by a Host System. 00b - Removable Card Slot, 01b - Embedded Slot for One Device, 10b - Shared Bus Slot, 11b - Reserved
29	0b RO	**Asynchronous Interrupt Support (async_int_support):** Refer to SDIO Specification Version 3.00 about asynchronous interrupt. 1 - Asynchronous Interrupt Supported, 0 - Asynchronous Interrupt Not Supported
28	0b RO	**64-bit System Bus Support (sys_bus_support_64b):** 1 - supports 64 bit system address, 0 - Does not support 64 bit system address
27	0b RO	**RSVD4 (rsvd4):** Reserved
26	0b RO	**Voltage Support 1.8V (volt_support_1p8v):** 0 - 1.8V Not Supported, 1 - 1.8V Supported
25	0b RO	**Voltage Support 3.0V (volt_support_3p0v):** 0 - 3.0V Not Supported, 1 - 3.0V Supported
24	1b RO	**Voltage Support 3.3V (volt_support_3p3v):** 0 - 3.3V Not Supported, 1 - 3.3V Supported
23	1b RO	**Suspend/Resume Support (suspend_resume_support):** This bit indicates whether the HC supports Suspend / Resume functionality. If this bit is 0, the Suspend and Resume mechanism are not supported and the HD shall not issue either Suspend / Resume commands. 0 - Not Supported, 1 - Supported
22	1b RO	**SDMA Support (sdma_support):** This bit indicates whether the HC is capable of using DMA to transfer data between system memory and the HC directly. 0 - SDMA Not Supported, 1 - SDMA Supported.
21	1b RO	**High Speed Support (high_speed_support):** This bit indicates whether the HC and the Host System support High Speed mode and they can supply SD Clock frequency from 25MHz to 50 MHz (for SD)/ 20MHz to 52MHz (for MMC). 0 - High Speed Not Supported, 1 - High Speed Supported
20	0b RO	**RSVD5 (rsvd5):** Reserved
19	1b RO	**ADMA2 Support (adma2_support):** 1 - ADMA2 support. 0 - ADMA2 not support
18	1b RO	**Extended Media Bus Support (extended_media_bus_support):** This bit indicates whether the Host Controller is capable of using 8-bit bus width mode. This bit is not effective when Slot Type is set to 10b. In this case, refer to Bus Width Preset in the Shared Bus resister. 1 - Extended Media Bus Supported, 0 - Extended Media Bus not Supported

Bit Range	Default & Access	Description
17: 16	00b RO	**Max Block Length (max_block_length):** This value indicates the maximum block size that the HD can read and write to the buffer in the HC. The buffer shall transfer this block size without wait cycles. Three sizes can be defined as indicated below. 00 - 512 byte, 01 - 1024 byte, 10 - 2048 byte, 11 - 4096 byte
15: 8	32h RO	**Base Clock Frequency for SD Clock (base_clock_frequency_sd_clk):** (1) 6-bit Base Clock Frequency This mode is supported by the Host Controller Version 1.00 and 2.00. Upper 2-bit is not effective and always 0. Unit values are 1MHz. The supported clock range is 10MHz to 63MHz. 11xx xxxxb - Not supported, 0011 1111b - 63MHz, 0000 0010b - 2MHz, 0000 0001b - 1MHz, 0000 0000b - Get information via another method. (2) 8-bit Base Clock Frequency This mode is supported by the Host Controller Version 3.00.Unit values are 1MHz. The supported clock range is 10MHz to 255MHz. FFh - 255MHz, 02h - 2MHz, 01h - 1MHz, 00h - Get information via another method. If the real frequency is 16.5MHz, the lager value shall be set 0001 0001b (17MHz) because the Host Driver use this value to calculate the clock divider value (Refer to the SDCLK Frequency Select in the Clock Control register.) and it shall not exceed upper limit of the SD Clock frequency. If these bits are all 0, the Host System has to get information via another method.
7	1b RO	**Timeout Clock Unit (timeout_clock_unit):** This bit shows the unit of base clock frequency used to detect Data Timeout Error. 0 - KHz, 1 - MHz
6	0b RO	**RSVD6 (rsvd6):** Reserved
5: 0	32h RO	**Timeout Clock Frequency (timeout_clock_frequency):** This bit shows the base clock frequency used to detect Data Timeout Error. Not 0 - 1KHz to 63KHz or 1MHz to 63MHz, 000000b - Get Information via another method.

17.6.29 Capabilities Register 2 (CAPABILITIES_2)—Offset 44h

This register provides the host driver with information specific to the host controller implementation.

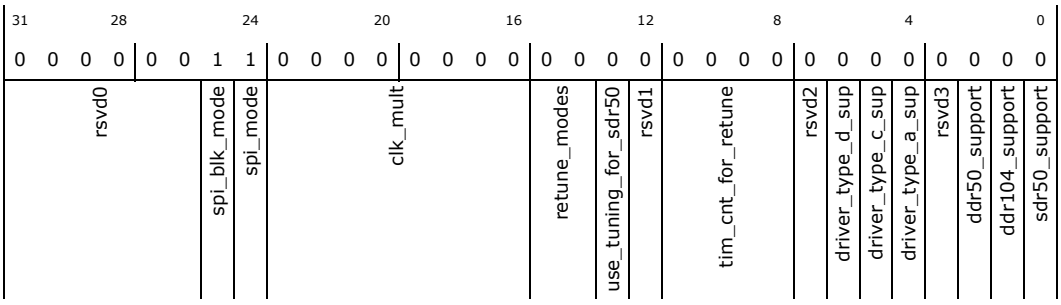
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

CAPABILITIES_2: [BAR0] + 44h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 03000000h



Bit Range	Default & Access	Description
31: 26	0b RO	**RSVD0 (rsvd0):** Reserved
25	1b RO	**SPI Block Mode (spi_blk_mode):** SPI block mode. 0 - Not Supported, 1 - Supported NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00

Bit Range	Default & Access	Description
24	1b RO	**SPI Mode (spi_mode):** SPI mode. 0 - Not Supported, 1 - Supported NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00
23: 16	0b RO	**Clock Multiplier (clk_mult):** This field indicates clock multiplier value of programmable clock generator. Refer to Clock Control register. Setting 00h means that Host Controller does not support programmable clock generator. FFh Clock Multiplier M = 256, ..., 02h Clock Multiplier M = 3, 01h Clock Multiplier M = 2, 00h Clock Multiplier is Not Supported
15: 14	0b RO	**Re-Tuning Modes (retune_modes):** This field defines the re-tuning capability of a Host Controller and how to manage the data transfer length and a Re-Tuning Timer by the Host Driver 00 - Mode1, 01 - Mode2, 10 - Mode3, 11 - Reserved. There are two re-tuning timings: Re-Tuning Request and expiration of a Re-Tuning Timer. By receiving either timing, the Host Driver executes the re-tuning procedure just before a next command issue
13	0b RO	**Use Tuning for SDR50 (use_tuning_for_sdr50):** If this bit is set to 1, this Host Controller requires tuning to operate SDR50. (Tuning is always required to operate SDR104.) 1 SDR50 requires tuning 0, SDR50 does not require tuning
12	0b RO	**RSVD1 (rsvd1):** Reserved
11: 8	0h RO	**Timer count for Re-Tuning (tim_cnt_for_retune):** This field indicates an initial value of the Re-Tuning Timer for Re-Tuning Mode 1 to 3. 0h - Get information via other source, 1h = 1 seconds, 2h = 2 seconds, 3h = 4 seconds, 4h = 8 seconds, --, n = 2(n-1) seconds, --, Bh = 1024 seconds, Fh - Ch = Reserved
7	0b RO	**RSVD2 (rsvd2):** Reserved
6	0b RO	**Driver Type D Support (driver_type_d_sup):** This bit indicates support of Driver Type D for 1.8 Signaling. 1 - Driver Type D is Supported, 0 - Driver Type D is Not Supported.
5	0b RO	**Driver Type C Support (driver_type_c_sup):** This bit indicates support of Driver Type C for 1.8 Signaling. 1 - Driver Type C is Supported, 0 - Driver Type C is Not Supported.
4	0b RO	**Driver Type A Support (driver_type_a_sup):** This bit indicates support of Driver Type A for 1.8 Signaling. 1 - Driver Type A is Supported, 0 - Driver Type A is Not Supported.
3	0b RO	**RSVD3 (rsvd3):** Reserved
2	0b RO	**DDR50 Support (ddr50_support):** 1 - DDR50 is Supported, 0 - DDR50 is Not Supported
1	0b RO	**DDR104 Support (ddr104_support):** 1 - DDR104 is Supported, 0 - DDR104 is Not Supported
0	0b RO	**SDR50 Support (sdr50_support):** If SDR104 is supported, this bit shall be set to 1. Bit 40 indicates whether SDR50 requires tuning or not. 1 - SDR50 is Supported, 0 - SDR50 is Not Supported

17.6.30 Maximum Current Capabilities Register (MAX_CUR_CAP)—Offset 48h

These registers indicate maximum current capability for each voltage. The value is meaningful if Voltage Support is set in the Capabilities register. If this information is supplied by the Host System via another method, all Maximum Current Capabilities register shall be 0.

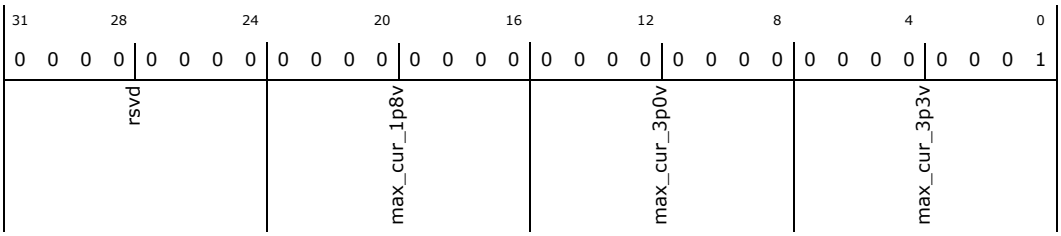
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

MAX_CUR_CAP: [BAR0] + 48h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000001h



Bit Range	Default & Access	Description
31: 24	0h RO	**RSVD (rsvd):** Reserved
23: 16	00h RO	**Maximum Current for 1.8V (max_cur_1p8v):** Maximum current capability for 1.8V
15: 8	00h RO	**Maximum Current for 3.0V (max_cur_3p0v):** Maximum current capability for 3.0V
7: 0	01h RO	**Maximum Current for 3.3V (max_cur_3p3v):** Maximum current capability for 3.3V

17.6.31 Force Event Register for Auto CMD12 Error Status (FORCE_EVENT_CMD12_ERR_STAT)—Offset 50h

The Force Event Register is not a physically implemented register. Rather, it is an address at which the Auto CMD Error Status Register can be written. Writing 1 : set each bit of the Auto CMD Error Status Register Writing 0 : no effect

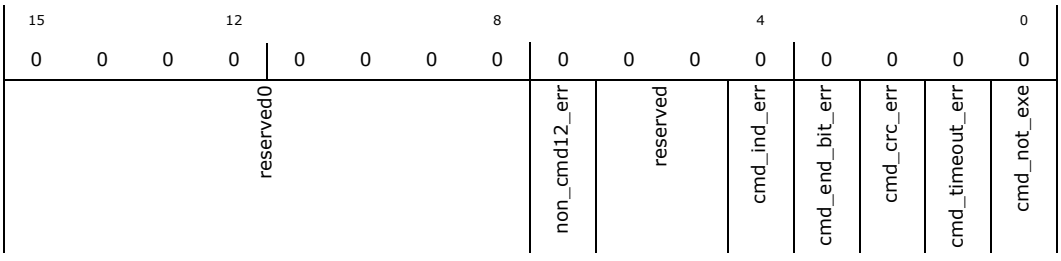
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

FORCE_EVENT_CMD12_ERR_STAT: [BAR0] + 50h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 8	00h RO	**Reserved0 (reserved0):** Reserved

Bit Range	Default & Access	Description
7	0b RW	**Force Event for Command Not Issued By Auto CMD12 Error (non_cmd12_err):** 1 - Interrupt is generated, 0 - No interrupt
6: 5	00b RO	**Reserved (reserved):** Reserved
4	0b RW	**Force Event for Auto CMD Index Error (cmd_ind_err):** 1 - Interrupt is generated, 0 - No interrupt
3	0b RW	**Force Event for Auto CMD End Bit Error (cmd_end_bit_err):** 1 - Interrupt is generated, 0 - No interrupt
2	0b RW	**Force Event for Auto CMD CRC Error (cmd_crc_err):** 1 - Interrupt is generated, 0 - No interrupt
1	0b RW	**Force Event for Auto CMD Timeout Error (cmd_timeout_err):** 1 - Interrupt is generated, 0 - No interrupt
0	0b RW	**Force Event for Auto CMD Not Executed (cmd_not_exe):** 1 - Interrupt is generated, 0 - No interrupt

17.6.32 Force Event Register for Error Interrupt Status (FORCE_EVENT_ERR_INT_STAT)—Offset 52h

The Force Event Register is not a physically implemented register. Rather, it is an address at which the Error Interrupt Status register can be written. The effect of a write to this address will be reflected in the Error Interrupt Status Register if the corresponding bit of the Error Interrupt Status Enable Register is set. Writing 1 : set each bit of the Error Interrupt Status Register Writing 0 : no effect Note: By setting this register, the Error Interrupt can be set in the Error Interrupt Status register. In order to generate interrupt signal, both the Error Interrupt Status Enable and Error Interrupt Signal Enable shall be set.

Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

FORCE_EVENT_ERR_INT_STAT: [BAR0] + 52h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rsvd0		ceata_err	tgt_rsp_err	rsvd		adma_err	cmd12_err	cur_limit_err	data_end_bit_err	data_crc_err	data_timeout_err	cmd_ind_err	cmd_end_bit_err	cmd_crc_err	cmd_timeout_err

Bit Range	Default & Access	Description
15: 14	00b RO	**Force Event for Vendor Specific Error Status (rsvd0):** Reserved
13	0b RW	**Force Event for CEATA error (ceata_err):** 1 - Interrupt is generated, 0 - No interrupt

Bit Range	Default & Access	Description
12	0b RW	**Force Event for Target Response Error (tgt_rsp_err):** 1 - Interrupt is generated, 0 - No interrupt
11: 10	0h RO	**RSVD (rsvd):** Reserved
9	0h RW	**Force Event for ADMA Error (adma_err):** 1 - Interrupt is generated, 0 - No interrupt
8	0h RW	**Force Event for Auto CMD Error (cmd12_err):** 1 - Interrupt is generated, 0 - No interrupt
7	0h RW	**Force Event for Current Limit Error (cur_limit_err):** 1 - Interrupt is generated, 0 - No interrupt
6	0h RW	**Force Event for Data End Bit Error (data_end_bit_err):** 1 - Interrupt is generated, 0 - No interrupt
5	0h RW	**Force Event for Data CRC Error (data_crc_err):** 1 - Interrupt is generated, 0 - No interrupt
4	0h RW	**Force Event for Data Timeout Error (data_timeout_err):** 1 - Interrupt is generated, 0 - No interrupt
3	0h RW	**Force Event for Command Index Error (cmd_ind_err):** 1 - Interrupt is generated, 0 - No interrupt
2	0h RW	**Force Event for Command End Bit Error (cmd_end_bit_err):** 1 - Interrupt is generated, 0 - No interrupt
1	0h RW	**Force Event for Command CRC Error (cmd_crc_err):** 1 - Interrupt is generated, 0 - No interrupt
0	0h RW	**Force Event for Command Timeout Error (cmd_timeout_err):** 1 - Interrupt is generated, 0 - No interrupt

17.6.33 ADMA Error Status Register (ADMA_ERR_STAT)—Offset 54h

When ADMA Error Interrupt is occurred, the ADMA Error States field in this register holds the ADMA state and the ADMA System Address Register holds the address around the error descriptor. For recovering the error, the Host Driver requires the ADMA state to identify the error descriptor address as follows: ST_STOP: Previous location set in the ADMA System Address register is the error descriptor address ST_FDS: Current location set in the ADMA System Address register is the error descriptor address ST_CADR: This sate is never set because do not generate ADMA error in this state. ST_TFR: Previous location set in the ADMA System Address register is the error descriptor address In case of write operation, the Host Driver should use ACMD22 to get the number of written block rather than using this information, since unwritten data may exist in the Host Controller. The Host Controller generates the ADMA Error Interrupt when it detects invalid descriptor data (Valid=0) at the ST_FDS state. In this case, ADMA Error State indicates that an error occurs at ST_FDS state. The Host Driver may find that the Valid bit is not set in the error descriptor.

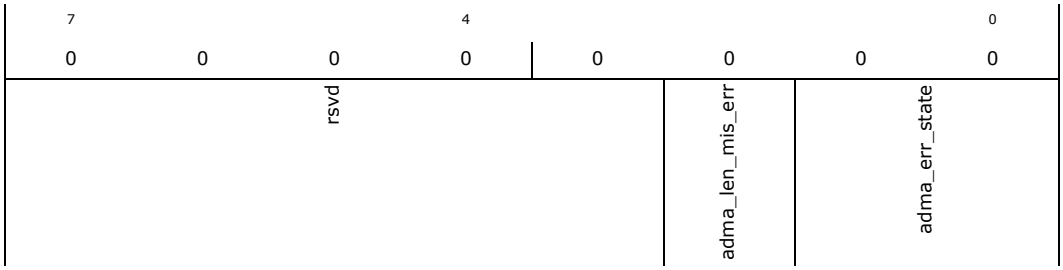
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

ADMA_ERR_STAT: [BAR0] + 54h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7: 3	0h RO	**RSVD (rsvd):** Reserved
2	0b RO	**ADMA Length Mismatch Error (adma_len_mis_err):** This error occurs in the following 2 cases. 1. While Block Count Enable being set, the total data length specified by the Descriptor table is different from that specified by the Block Count and Block Length. 2. Total data length can not be divided by the block length. 1 - Error, 0 - No error
1: 0	00b RO	**ADMA Error State (adma_err_state):** This field indicates the state of ADMA when error is occurred during ADMA data transfer. This field never indicates 10 because ADMA never stops in this state. D01 D00: ADMA Error State when error is occurred Contents of SYS_SDR register, 00 - ST_STOP (Stop DMA) Points next of the error descriptor, 01 - ST_FDS (Fetch Descriptor) Points the error descriptor, 10 - Never set this state (Not used), 11 - ST_TFR (Transfer Data) Points the next of the error descriptor

17.6.34 ADMA System Address Register (ADMA_SYS_ADDR)—Offset 58h

This register contains the physical Descriptor address used for ADMA data transfer.

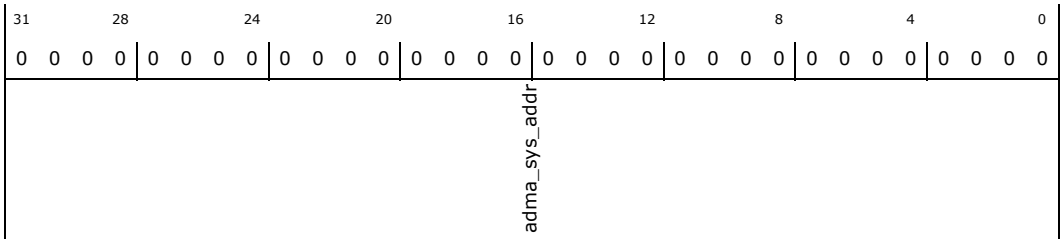
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

ADMA_SYS_ADDR: [BAR0] + 58h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**ADMA System Address (adma_sys_addr):** This register holds byte address of executing command of the Descriptor table. 32-bit Address Descriptor uses lower 32- bit of this register. At the start of ADMA, the Host Driver shall set start address of the Descriptor table. The ADMA increments this register address, which points to next line, when every fetching a Descriptor line. When the ADMA Error Interrupt is generated, this register shall hold valid Descriptor address depending on the ADMA state. The Host Driver shall program Descriptor Table on 32-bit boundary and set 32-bit boundary address to this register. ADMA2 ignores lower 2-bit of this register and assumes it to be 00b.

17.6.35 initialization Preset Values Register (3.3v or 1.8v) (PRESET_VALUE_0)—Offset 60h

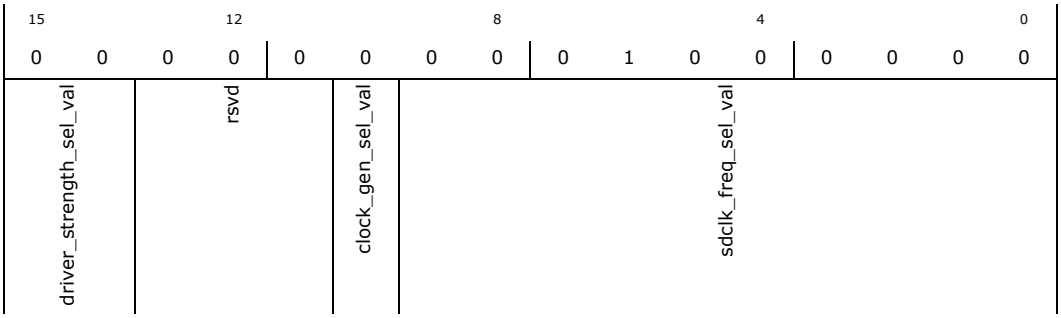
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_0: [BAR0] + 60h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0040h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	040h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.36 Default Speed Preset Values Register (PRESET_VALUE_1)—Offset 62h

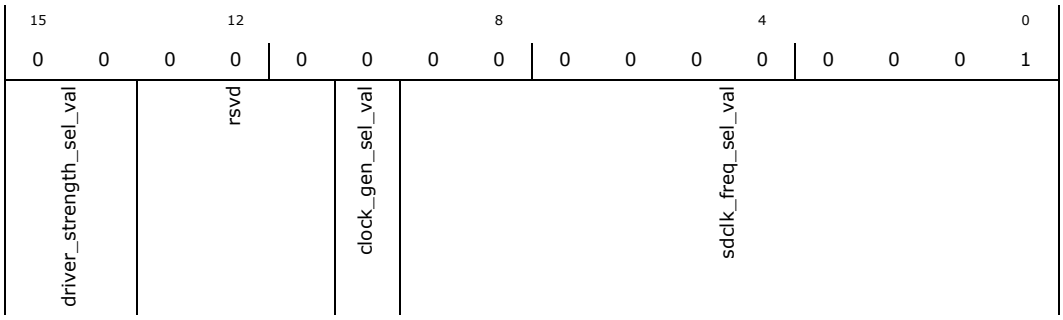
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_1: [BAR0] + 62h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0001h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	1h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.37 High Speed Preset Values Register (PRESET_VALUE_2)—Offset 64h

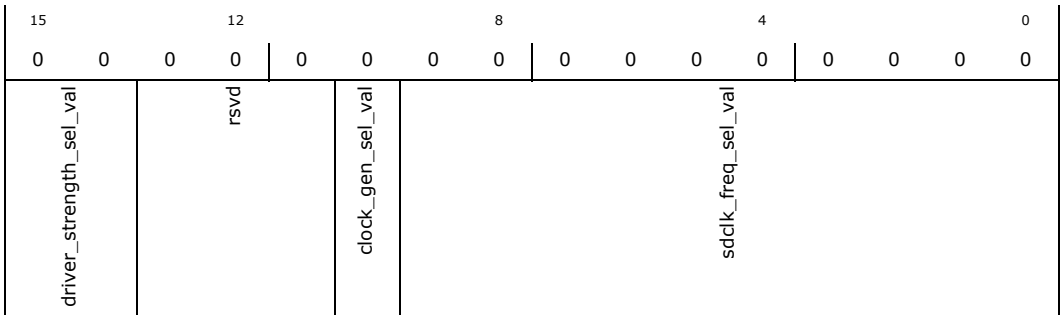
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_2: [BAR0] + 64h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected

Bit Range	Default & Access	Description
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	0h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.38 SDR12 Preset Values Register (PRESET_VALUE_3)—Offset 66h

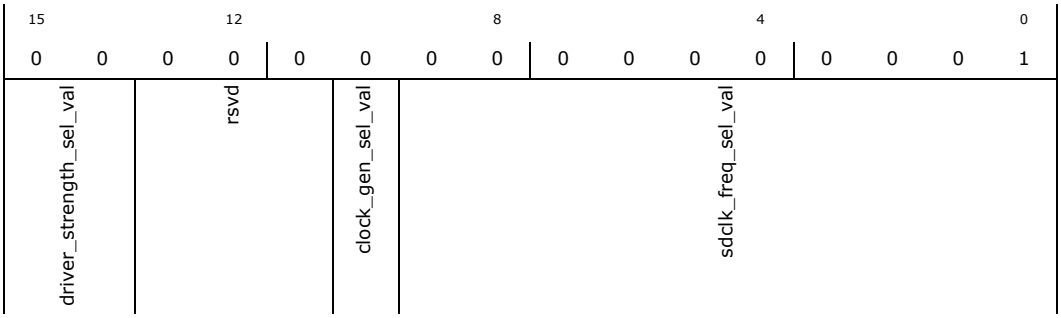
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_3: [BAR0] + 66h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0001h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	1h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.39 SDR25 Preset Values Register (PRESET_VALUE_4)—Offset 68h

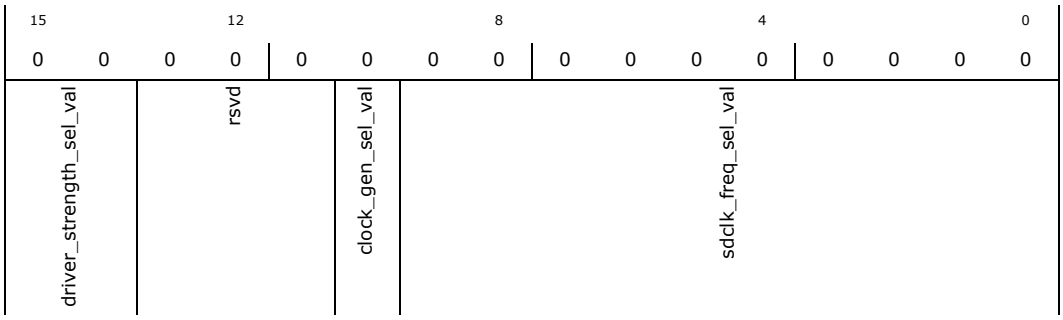
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_4: [BAR0] + 68h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	0h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.40 SDR50 Preset Values Register (PRESET_VALUE_5)—Offset 6Ah

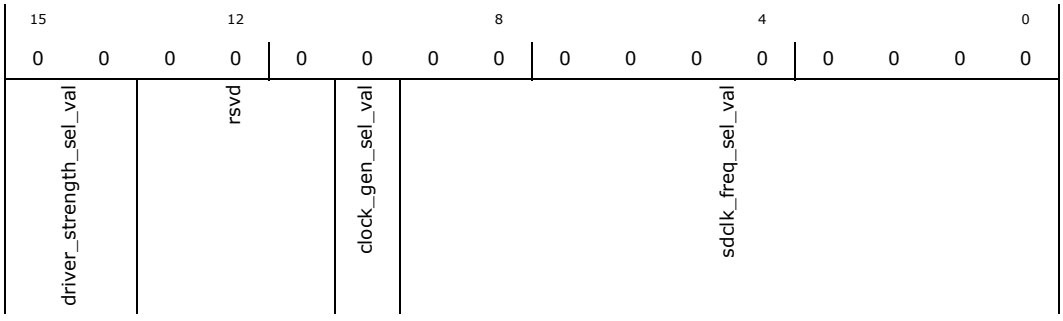
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_5: [BAR0] + 6Ah

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved

Bit Range	Default & Access	Description
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	0h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.41 SDR104 Preset Values Register (PRESET_VALUE_6)—Offset 6Ch

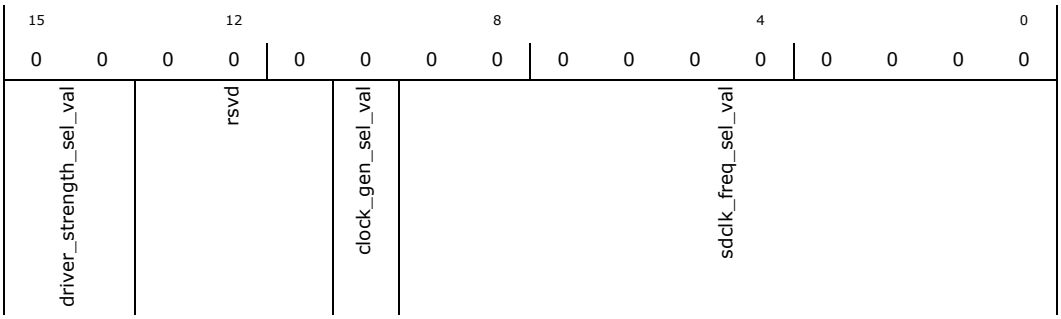
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_6: [BAR0] + 6Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	0h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.42 DDR50 Preset Values Register (PRESET_VALUE_7)—Offset 6Eh

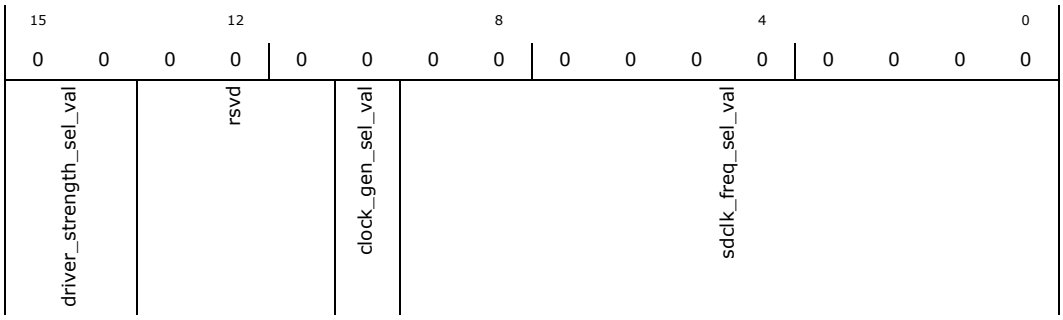
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PRESET_VALUE_7: [BAR0] + 6Eh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 14	00b RO	**Driver Strength Select Value (driver_strength_sel_val):** Driver Strength is supported by 1.8V signaling bus speed modes. This field is meaningless for 3.3V signaling. 11b - Driver Type D is Selected, 10b - Driver Type C is Selected, 01b - Driver Type A is Selected, 00b - Driver Type B is Selected
13: 11	000b RO	**RSVD (rsvd):** Reserved
10	0b RO	**Clock Generator Select Value (clock_gen_sel_val):** This bit is effective when Host Controller supports programmable clock generator. 1 - Programmable Clock Generator, 0 - Host Controller Ver2.00 Compatible Clock Generator
9: 0	0h RO	**SDCLK Frequency Select Value (sdclk_freq_sel_val):** 10-bit preset value to set SDCLK Frequency Select in the Clock Control Register is described by a host system.

17.6.43 Boot Time-out control register (BOOT_TIMEOUT_CTRL)—Offset 70h

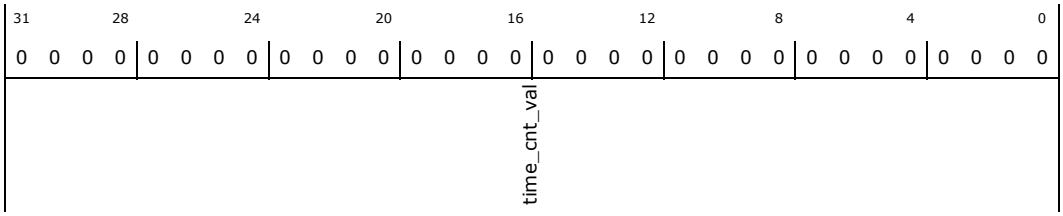
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

BOOT_TIMEOUT_CTRL: [BAR0] + 70h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RW	**Boot Data Timeout Counter Value (time_cnt_val):** This value determines the interval by which DAT line time-outs are detected during boot operation for eMMC4.4 card. The value is in number of sd clock.

17.6.44 Debug Selection Register (DEBUG_SEL)—Offset 74h

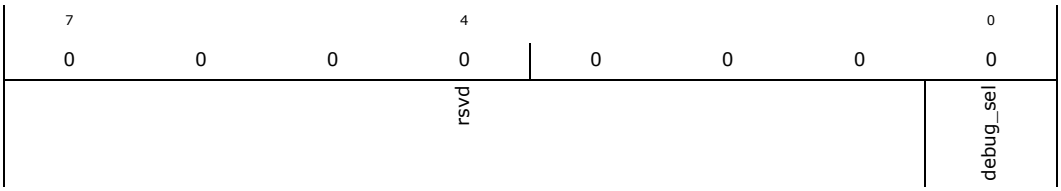
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

DEBUG_SEL: [BAR0] + 74h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7: 1	00h RO	**RSVD (rsvd):** Reserved
0	0b WO	**Debug Select (debug_sel):** 1- cmd register, Interrupt status, transmitter module, ahb_iface module and clk sdcard signals are probed out. 0 - receiver module and fifo_ctrl module signals are probed out

17.6.45 Shared Bus Control Register (SHARED_BUS)—Offset E0h

This register is optional. The devices on shared bus are not intended to be controlled by the Standard Host Driver. This is because shared bus configuration depends on a host system; the devices on shared bus may be controlled by a specific driver of a host system.

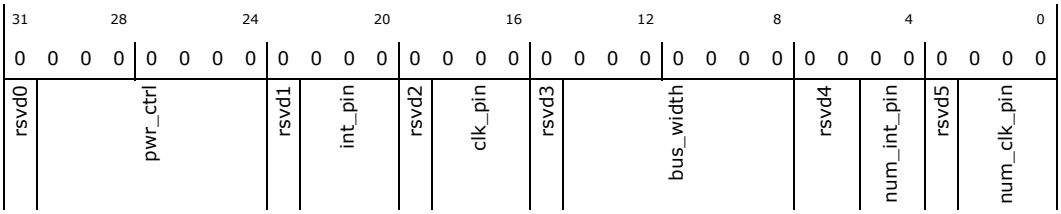
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SHARED_BUS: [BAR0] + E0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RO	**RSVD0 (rsvd0):** Reserved

Bit Range	Default & Access	Description
30: 24	0h RW	**Back-End Power Control (pwr_ctrl):** Each bit of this field controls back-end power supply for an embedded device. Host interface voltage (VDDH) is not controlled by this field. The number of devices supported is specified by Number of Clock Pins and a maximum of 7 devices can be controlled. D16 Back-end Power Control for Device 1 D17 Back-end Power Control for Device 2 D18 Back-end Power Control for Device 3 D19 Back-end Power Control for Device 4 D20 Back-end Power Control for Device 5 D21 Back-end Power Control for Device 6 D22 Back-end Power Control for Device 7 The function of each bit is defined as follows: 0 Back-end Power is Off 1 Back-end Power is Supplied Back-End power control is effective for embedded memory devices in the Sleep State that support the Sleep command (CMD14) to reduce power consumption and embedded SDIO devices when IOEx is set to 0.
23	0b RO	**RSVD1 (rsvd1):** Reserved
22: 20	0h RW	**Interrupt Pin Select (int_pin):** Interrupt pin inputs are enabled by this field. Enable of unsupported interrupt pin is meaningless. 000b - Interrupt is detected by Interrupt Cycle, xx1b - INT_A is Enabled, x1xb - INT_B is Enabled, 1xxb - INT_C is Enabled
19	0b RO	**RSVD2 (rsvd2):** Reserved
18: 16	0h RW	**Clock Pin Select (clk_pin):** One of clock pin outputs is selected by this field. Select of unsupported clock pin is meaningless. Refer to Figure 2-38 (An Example Timing of Selecting Clock Pin) on SD Host Controller Simplified Specification Version 3.00 for the timing of clock outputs. 000b - Clock Pins are Disabled, 001b - CLK[1] is Selected, 010b - CLK[2] is Selected --- 111b - CLK[7] is Selected
15	0b RO	**RSVD3 (rsvd3):** Reserved
14: 8	0h RO	**Bus Width Preset (bus_width):** Shared bus supports mixing of 4-bit and 8- bit bus width devices. Each bit of this field specifies the bus width for each embedded device. The number of devices supported is specified by Number of Clock Pins and a maximum of 7 devices are supported.This field is effective when multiple devices are connected to a shared bus (Slot Type is set to 10b in the Capabilities register). In the other case, Extended Data Transfer Width in the Host Control 1 register is used to select 8-bit bus width. As use of 1-bit mode is not intended for shared bus, Data Transfer Width in the Host Control 1 register should be set to 1. D08 - Bus width preset for Device 1, D09 - Bus width preset for Device 2, D10 - Bus width preset for Device 3, D11 - Bus width preset for Device 4, D12 - Bus width preset for Device 5, D13 - Bus width preset for Device 6, D14 - Bus width preset for Device 7 The function of each bit is defined as follows: 0 - 4 bit bus width mode, 1 - 8 bit bus width mode
7: 6	00b RO	**RSVD4 (rsvd4):** Reserved
5: 4	0h RO	**Number of Interrupt Input Pins (num_int_pin):** This field indicates support of interrupt input pins for shared bus system.Three asynchronous interrupt pins are defined, INT_A#, INT_B# and INT_C#. Which interrupt pin is used is determined by the system. Each one is driven by open drain and then wired or connection is possible. 00b - Interrupt Input Pin is Not Supported, 01b - INTA is Supported, 10b - INTA and INTB are Supported, 11b - INTA, INTB and INTC are Supported
3	0b RO	**RSVD5 (rsvd5):** Reserved

Bit Range	Default & Access	Description
2: 0	0h RO	**Number of Clock Pins (num_clk_pin):** This field indicates support of clock pins to select one of devices for shared bus system. Up to 7 clock pins can be supported.Shared bus is supported by specific system. Then Standard Host Driver does not support control of these clock pins. 000b - Shared bus is not supported, 001b - 1 SDCLK pin is supported, 010b - 2 SDCLK pins are supported, ... , 111b - 7 SDCLK pins are supported

17.6.46 SPI Interrupt Support Register (SPI_INT_SUP)—Offset F0h

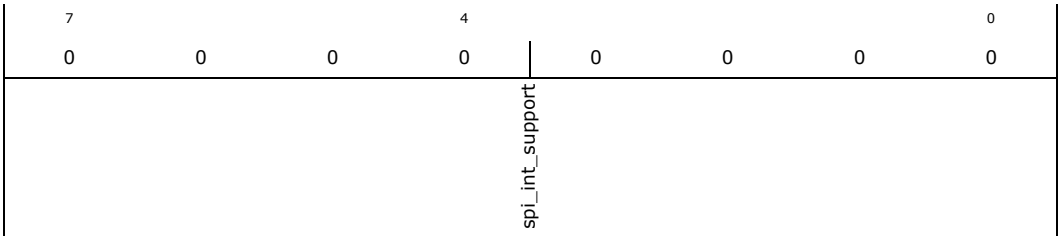
Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

SPI_INT_SUP: [BAR0] + F0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RW	**SPI Interrupt Support (spi_int_support):** This bit is set to indicate the assertion of interrupts in the SPI mode at any time, irrespective of the status of the card select (CS) line. If this bit is zero, then SDIO card can only assert the interrupt line in the SPI mode when the CS line is asserted. NOTE: Not part of the SD Host Controller Simplified Specification Version 3.00

17.6.47 Slot Interrupt Status Register (SLOT_INT_STAT)—Offset FCh

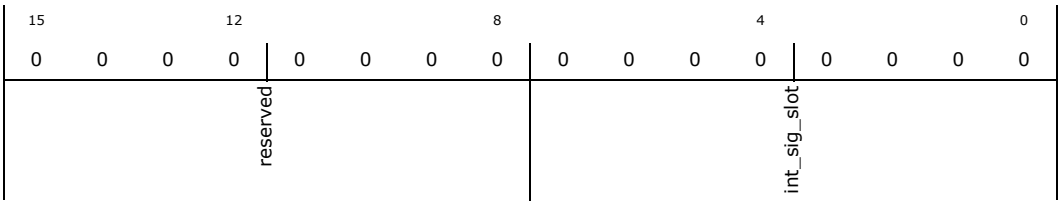
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

SLOT_INT_STAT: [BAR0] + FCh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: 0000h



Bit Range	Default & Access	Description
15: 8	00h RO	**RSVD (reserved):** Reserved
7: 0	00h RO	**Interrupt Signal For Each Slot (int_sig_slot):** These status bit indicate the logical OR of Interrupt signal and wake up signal for each slot. A maximum of 8 slots can be defined. If one interrupt signal is associated with multiple slots. the HD can know which interrupt is generated by reading these status bits. By a power on reset or by Software Reset For All, the Interrupt signal shall be de asserted and this status shall read 00h. Bit 00 - Slot 1, Bit 01 - Slot 2, Bit 02 - Slot 3, ----- -----, Bit 07 - Slot 8

17.6.48 Host Controller Version Register (HOST_CTRL_VER)—Offset FEh

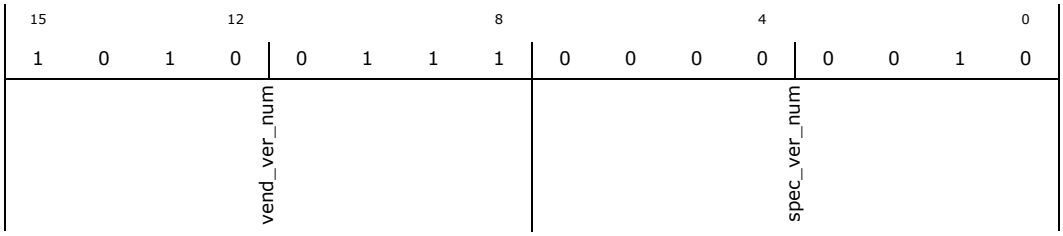
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

HOST_CTRL_VER: [BAR0] + FEh

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:0] + 10h

Default: A702h



Bit Range	Default & Access	Description
15: 8	a7h RO	**Vendor Version Number (vend_ver_num):** This status is reserved for the vendor version number. The HD should not use this status.
7: 0	02h RO	**Specification Version Number (spec_ver_num):** This status indicates the Host Controller Spec. Version. The upper and lower 4- bits indicate the version. 00 - SD Host Specification version 1.0, 01 - SD Host Specification version 2.00 including only the feature of the Test Register, 02 - SD Host Specification Version 3.00, others - Reserved

§ §

18.0 High Speed UART

The Intel® Quark SoC X1000 implements two instances of a 16550 compliant UART controller that supports baud rates between 300 and 2764800. Hardware flow control is also supported.

Note: Only one UART controller (UART0) provides MODEM pins.

18.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 96. UART 0 Interface Signals

Signal Name	Direction/ Type	Description
SIU0_RXD	I	UART 0 - Serial Input
SIU0_TXD	O	UART 0 - Serial Output
SIU0_RTS_B	O	UART 0 - MODEM Request to Send
SIU0_CTS_B	I	UART 0 - MODEM Clear to Send
SIU0_DCD_B	I	UART 0 - MODEM Data Carrier Detect
SIU0_DSR_B	I	UART 0 - MODEM Data Set Ready
SIU0_DTR_B	O	UART 0 - MODEM Data Terminal Ready
SIU0_RI_B	I	UART 0 - MODEM Ring Indicator

Table 97. UART 1 Interface Signals

Signal Name	Direction/ Type	Description
SIU1_RXD	I	UART 1 - Serial Input
SIU1_TXD	O	UART 1 -Serial Output
SIU1_RTS_B	O	UART 1 - MODEM Request to Send
SIU1_CTS_B	I	UART 1 - MODEM Clear to Send

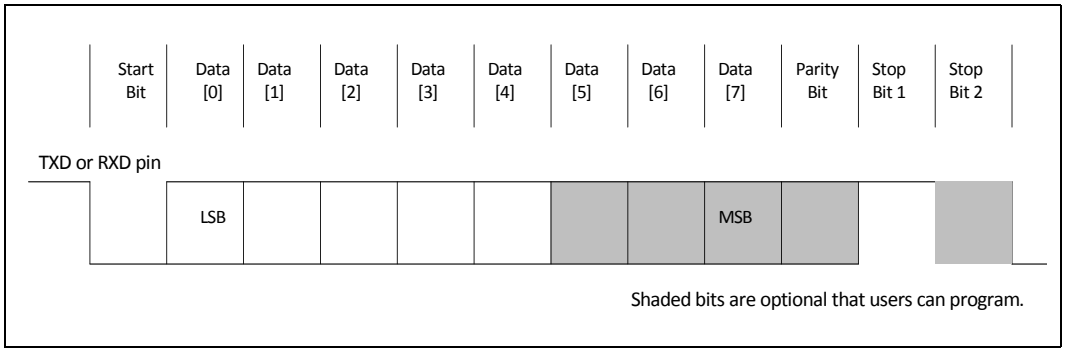
18.2 Features

18.2.1 UART Function

The UART transmits and receives data in bit frames as shown in Figure 33.

- Each data frame is between 7 and 12 bits long, depending on the size of data programmed and if parity and stop bits are enabled.
- The frame begins with a start bit that is represented by a high-to-low transition.
- Next, 5 to 8 bits of data are transmitted, beginning with the Least Significant Bit (LSB). An optional parity bit follows, which is set if even parity is enabled and an odd number of ones exist within the data byte; or, if odd parity is enabled and the data byte contains an even number of ones.
- The data frame ends with one, one and a half, or two stop bits (as programmed by users), which is represented by one or two successive bit periods of a logic one.

Figure 33. UART Data Transfer Flow



18.2.2 Baud Rate Generator

The baud rates for the UARTs are generated from the base frequency (Fbase) indicated in Table 98 by programming the DLH and DLL registers as divisor. The hexadecimal value of the divisor is (IER_DLH[7:0]<<8) | RBR_THR_DLL[7:0].

The output baud rate is equal to the base frequency divided by sixteen times the value of the divisor, as follows: baud rate = (Fbase) / (16 * divisor).

Table 98. Baud Rates Achievable with Different DLAB Settings

DLH,DLL Divisor	DLH,DLL Divisor Hexadecimal	Baud Rate
Fbase: 44236800 Hz		
1	0001	2764800
3	0003	921600
6	0006	460800
9	0009	307200
12	000C	230400
15	000F	184320
18	0012	153600
24	0018	115200
48	0030	57600
72	0048	38400

Table 98. Baud Rates Achievable with Different DLAB Settings

DLH,DLL Divisor	DLH,DLL Divisor Hexadecimal	Baud Rate
144	0090	19200
288	0120	9600
384	0180	7200
576	0240	4800
768	0300	3600
1152	0480	2400
1536	0600	1800
2304	0900	1200
4608	1200	600
9216	2400	300

18.3 Use

Each UART has a Transmit FIFO and a Receive FIFO and each FIFO holds 16 bytes of data. There are three separate methods for moving data into and out of the FIFOs: DMA, Interrupts, and Polling.

18.3.1 DMA Mode Operation

Each UART has an associated DMA Controller (DMAC) that is enabled by setting DMA_CFG_REG.DMA_EN to 1. Two DMA channels are used — one for transmit data and one for receive data.

A hardware interface between the UART and the DMA is used to signal when data can be read from the Receive FIFO and to signal when the Transmit FIFO is either empty or has reached a programmed threshold level. This interface gives the DMA all responsibility for the transfer of data and it must be programmed accordingly. Using the UART to set the DMA mode via IIR_FCR[3] has no effect. An interrupt is generated upon the completion of a DMA transfer.

18.3.1.1 Receiver DMA

The DMA Controller uses Channel 0 to transfer data from the UART to Host Memory. The UART requests a DMA transfer to memory under the following condition:

- When the Receiver FIFO is at or above programmed trigger level in FIFO mode

To transfer data from the UART, the source transfer width (CTL0_L.SRC_TR_WIDTH) is set to 8-bits to match the size of the FIFO entries and the destination transfer width (CTL0_L.DST_TR_WIDTH) is set to 32-bits.

The receive watermark level (IIR_FCR[7:6]) should be set such that DMA requests to transfer data to memory are made often enough for the Receiver FIFO to accept serial transfers continuously. This will prevent the Receiver FIFO from overflowing.

To prevent Receiver FIFO underflow, the source burst length must be set such that the FIFO can be emptied, but not underflowed, at the completion of the burst transaction. For optimal operation, CTL0_L.SRC_MSIZE should be set at the receive watermark level; that is:

- CTL0_L.SRC_MSIZE = decoded level of IIR_FCR[7:6]

18.3.1.2 Transmitter DMA

The DMA Controller uses Channel 1 to transfer data from Host Memory to the UART. The UART requests a DMA transfer from memory under the following conditions:

- When the transmitter FIFO is empty in FIFO mode with Programmable THRE interrupt mode disabled
- When the transmitter FIFO is at, or below the programmed threshold with FIFO and Programmable THRE interrupt mode enabled

To transfer data to the UART, the destination transfer width (CTL1_L.DST_TR_WIDTH) is set to 8-bits to match the size of the FIFO entries and the source transfer width (CTL1_L.SRC_TR_WIDTH) is set to 32-bits.

The transmit watermark level (IIR_FCR[5:4) should be set such that DMA requests to transfer data from memory are made often enough for the Transmitter FIFO to be able to perform serial transfers continuously. This will prevent the Transmitter FIFO from underflowing.

The Transmitter FIFO can overflow if CTL1_L.DEST_MSIZE is programmed to a value greater than the transmit watermark level as there may not be not enough space in the Transmitter FIFO to service the destination burst request. To avoid overflow, CTL1_L.DEST_MSIZE should be programmed as follows:

- CTL1_L.DEST_MSIZE <= UART FIFO DEPTH − decoded level of IIR_FCR[5:4]

For optimal operation, CTL1_L.DEST_MSIZE should be set at the FIFO level that triggers a transmit DMA request; that is:

- CTL1_L.DEST_MSIZE = UART FIFO DEPTH − decoded level of IIR_FCR[5:4]

18.3.2 FIFO Interrupt-Mode Operation

18.3.2.1 Receiver Interrupt

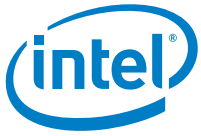
When the Receive FIFO and receiver interrupts are enabled (IIR_FCR[0]=1 and IER_DLH[0]=1), receiver interrupts occur as follows:

- The Receive Data Available Interrupt is asserted when the FIFO has reached its programmed trigger level. The interrupt is cleared when the FIFO drops below the programmed trigger level.
- The IIR Receive Data Available indication also occurs when the FIFO trigger level is reached, and like the interrupt, the bits are cleared when the FIFO drops below the trigger level.
- The Data Ready bit (LSR.DR) is set to 1 as soon as a character is transferred from the shift register to the Receive FIFO. This bit is reset to 0 when the FIFO is empty.

18.3.2.2 Transmitter Interrupt

When the transmitter FIFO and transmitter interrupt are enabled (IIR_FCR[0]=1, IER_DLH[1]=1), transmit interrupts occur as follows:

- When Programmable THRE Interrupt Mode is disabled (IER_DLH[7] set to 0), the transmitter interrupt occurs when the transmitter FIFO is empty. It is cleared when the transmitter FIFO is no longer empty or the interrupt identification register (IIR) is read.
- When Programmable THRE Interrupt Mode is enabled (IER_DLH[7] set to 1), the transmitter interrupt occurs when the number of entries in the transmitter FIFO is at or below a programmed transmitter FIFO empty threshold level (IIR_FCR[5:4]).



It is cleared when the number of entries in the transmitter FIFO is greater than the programmed transmitter FIFO empty threshold.

Users could cause the UART Transmit FIFO to overflow when too many characters are written. FIFO underflow does not cause an error as the UART waits for the Transmit FIFO to be serviced.

18.3.3 FIFO Polled-Mode Operation

With the FIFOs enabled (IIR_FCR[0] set to 1), clearing IER_DLH[7] and IER_DLH[3:0] puts the serial port in the FIFO Polled Operation mode. Because the receiver and the transmitter are controlled separately, either one or both can be in Polled Operation mode. In this mode, software checks Receiver and Transmitter status using the Line Status Register (LSR). The processor polls the following bits for Receive and Transmit Data Service.

18.3.3.1 Receive Data Service

The processor checks the Data Ready bit (LSR.DR) which is set when 1 or more bytes remains in the Receive FIFO or Receive Buffer Register (RBR_THR_DLL).

18.3.3.2 Transmit Data Service

The processor checks transmit data request LSR.THRE bit, which is set when the transmitter needs data.

The processor can also check transmitter empty LSR.TEMT, which is set when the Transmit FIFO or Holding register is empty.

18.3.4 Autoflow Control

Autoflow Control uses Clear-to-Send (nCTS) and Request-to-Send (nRTS) signals to automatically control the flow of data between the UART and external modem. When autoflow is enabled, the remote device is not allowed to send data unless the UART asserts nRTS low. If the UART de-asserts nRTS while the remote device is sending data, the remote device is allowed to send one additional byte after nRTS is de-asserted. An overflow could occur if the remote device violates this rule. Likewise, the UART is not allowed to transmit data unless the remote device asserts nCTS low. This feature increases system efficiency and eliminates the possibility of a Receive FIFO Overflow error due to long interrupt latency.

Autoflow mode can be used in two ways: Full autoflow, automating both nCTS and nRTS, and half autoflow, automating only nCTS. Full Autoflow is enabled by writing a 1 to bits 1 and 5 of the Modem Control Register (MCR). Auto-nCTS-Only mode is enabled by writing a 1 to bit 5 and a 0 to bit 1 of the MCR register.

18.3.4.1 RTS (UART Output)

When in Full-autoflow mode, nRTS is asserted when the UART FIFO is ready to receive data from the remote transmitter. This occurs when the amount of data in the Receive FIFO is below the programmable threshold value. When the amount of data in the Receive FIFO reaches the programmable threshold, nRTS is de-asserted. It will be asserted once again when enough bytes are removed from the FIFO to lower the data level below the threshold.

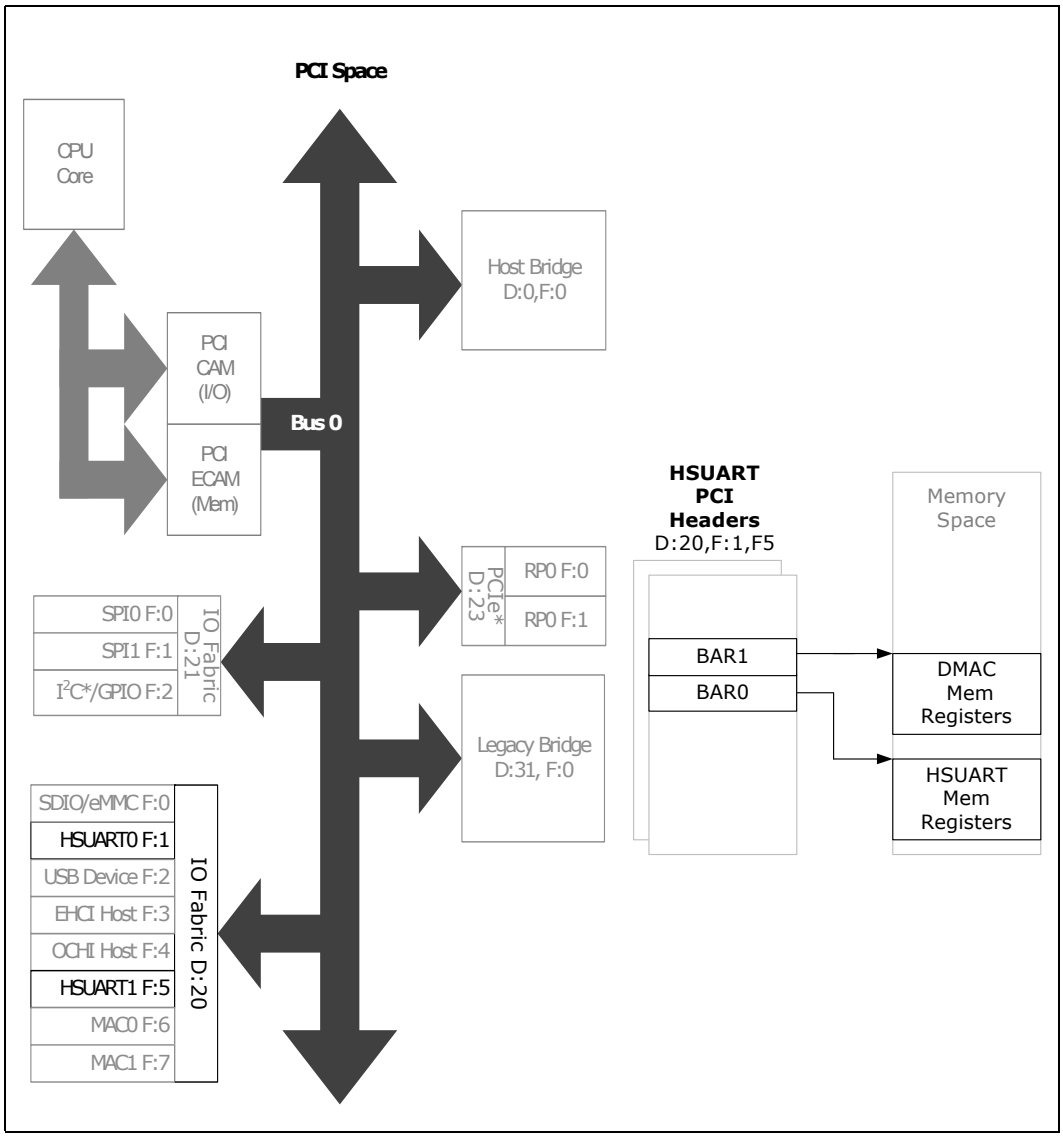
18.3.4.2 CTS (UART Input)

When in Full or Half-autoflow mode, nCTS is asserted by the remote receiver when the receiver is ready to receive data from the UART. The UART checks nCTS before sending the next byte of data and will not transmit the byte until nCTS is low. If nCTS goes high while the transfer of a byte is in progress, the transmitter completes this byte.

18.4 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 34. HSUART Register Map



18.5 PCI Configuration Registers

Registers listed are for Function 1 (HS-UART0). Function 5 (HS-UART1) contains the same registers. Differences between the HS-UARTs are noted in individual registers.

Table 99. Summary of PCI Configuration Registers—0/20/1

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 655	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 656	0936h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 656	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 657	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 658	07000210h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 658	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 659	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 659	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 659	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 660	00000000h
14h	17h	"Base Address Register (BAR1)—Offset 14h" on page 660	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 661	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 661	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 662	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 662	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 662	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 663	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 663	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 664	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 664	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 664	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 665	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 665	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 666	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 667	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 667	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 667	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 668	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 668	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 668	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 669	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 669	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 670	00000000h

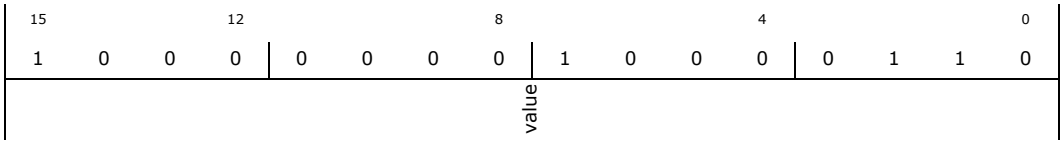
18.5.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:20, F:1] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

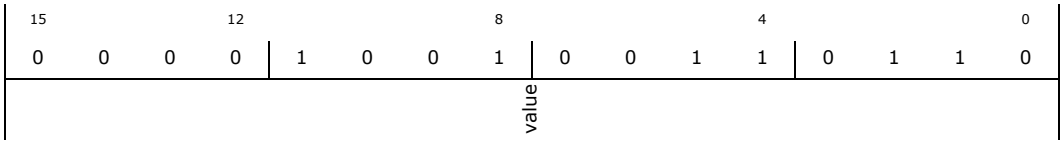
18.5.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:20, F:1] + 2h

Default: 0936h



Bit Range	Default & Access	Description
15: 0	0936h RO	**Device ID (value):** PCI Device ID

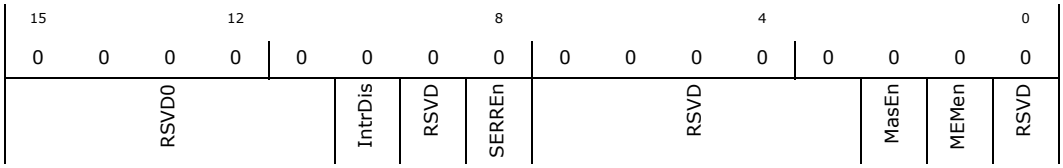
18.5.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:20, F:1] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

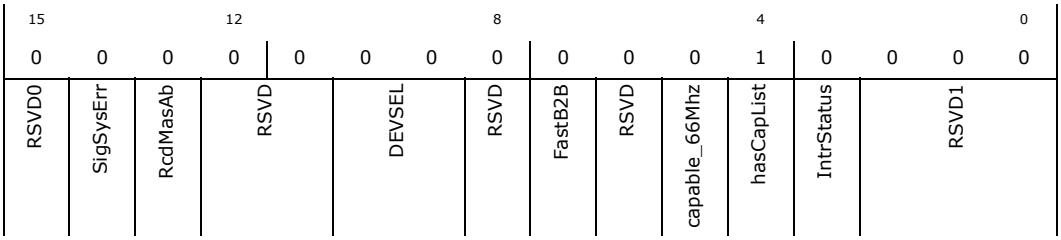
18.5.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:20, F:1] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.

Bit Range	Default & Access	Description
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

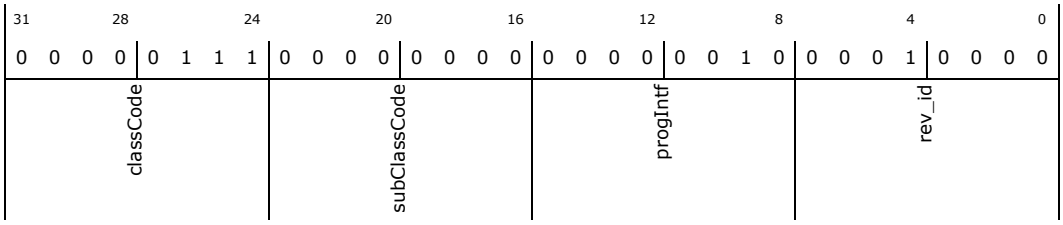
18.5.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:20, F:1] + 8h

Default: 07000210h



Bit Range	Default & Access	Description
31: 24	07h RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	00h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	02h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

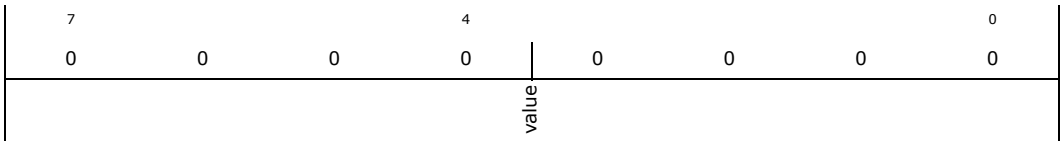
18.5.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:20, F:1] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

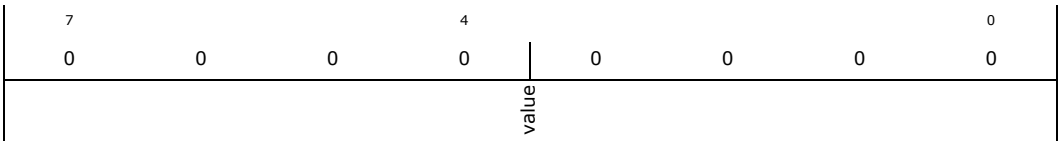
18.5.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:20, F:1] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

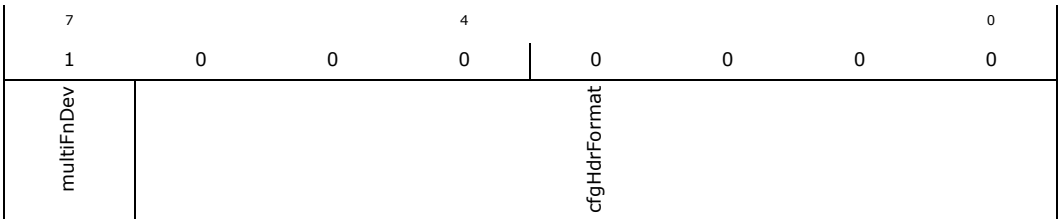
18.5.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:20, F:1] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

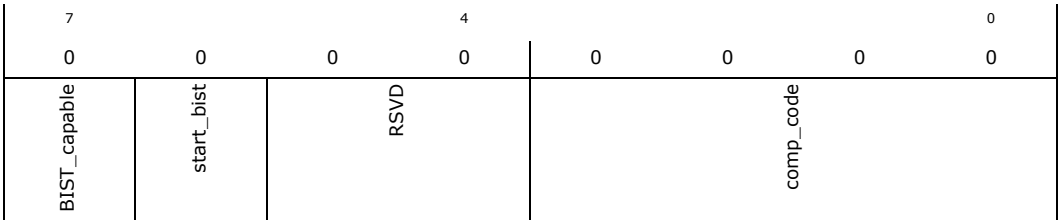
18.5.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:20, F:1] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

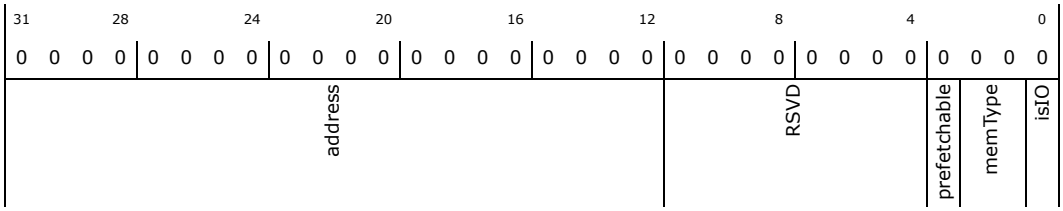
18.5.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

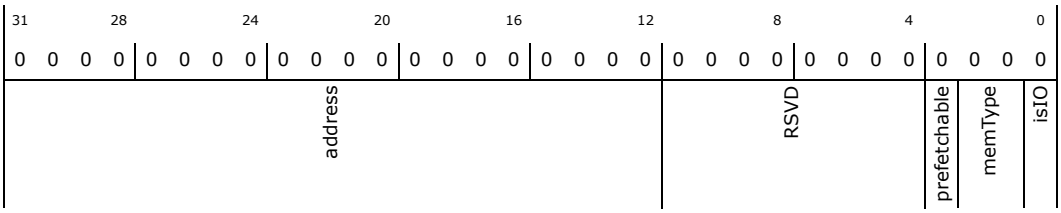
18.5.11 Base Address Register (BAR1)—Offset 14h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR1: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

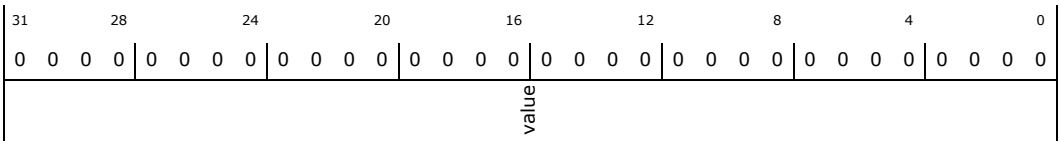
18.5.12 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:20, F:1] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

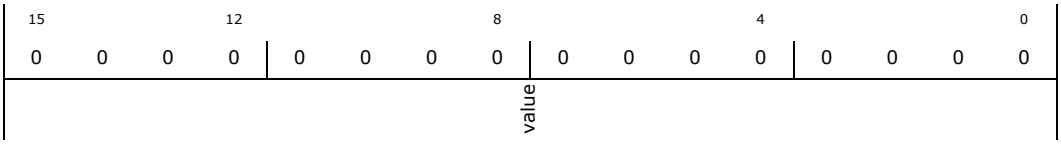
18.5.13 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:20, F:1] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

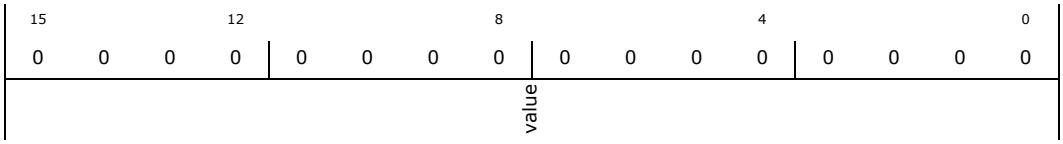
18.5.14 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:20, F:1] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

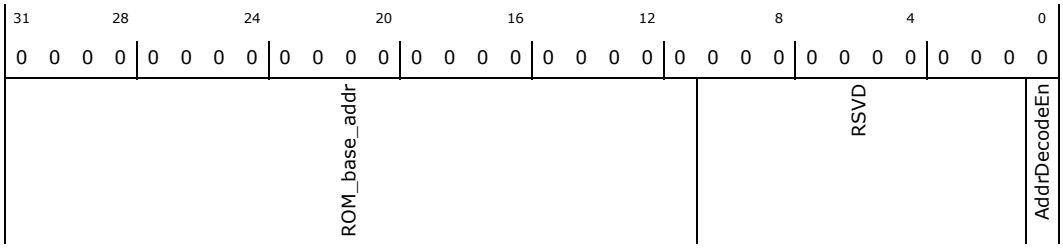
18.5.15 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:20, F:1] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

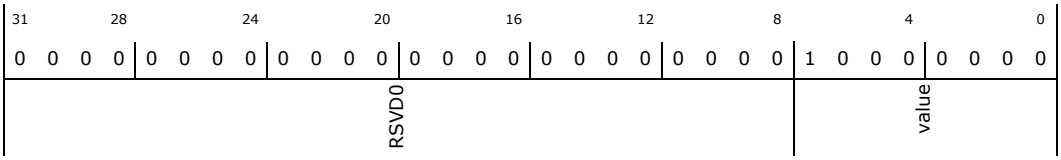
18.5.16 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

CAP_POINTER: [B:0, D:20, F:1] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

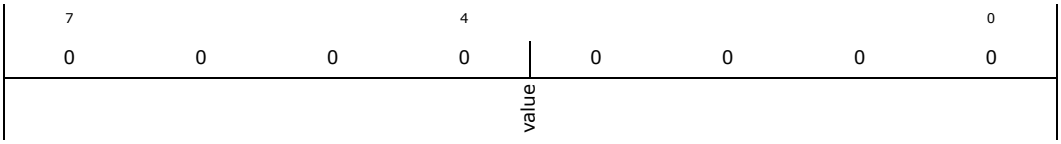
18.5.17 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register
(Size: 8 bits)

INTR_LINE: [B:0, D:20, F:1] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

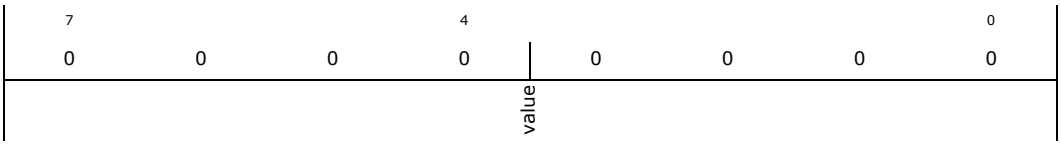
18.5.18 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register
(Size: 8 bits)

INTR_PIN: [B:0, D:20, F:1] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

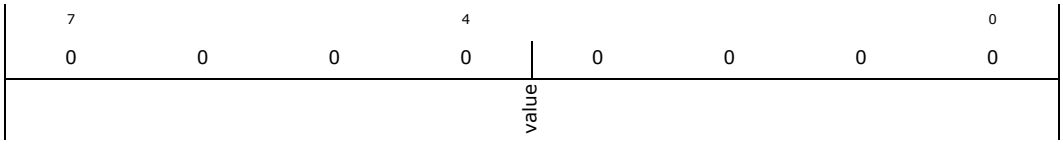
18.5.19 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:20, F:1] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

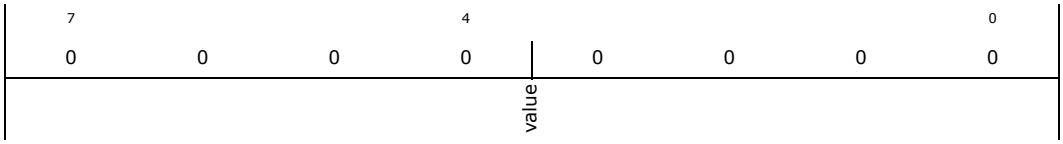
18.5.20 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:20, F:1] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

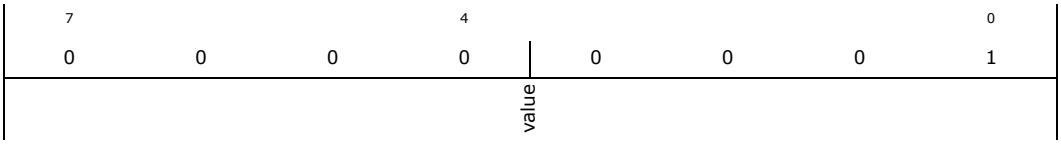
18.5.21 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:20, F:1] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

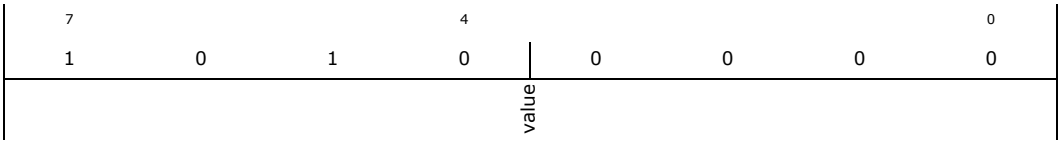
18.5.22 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:20, F:1] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

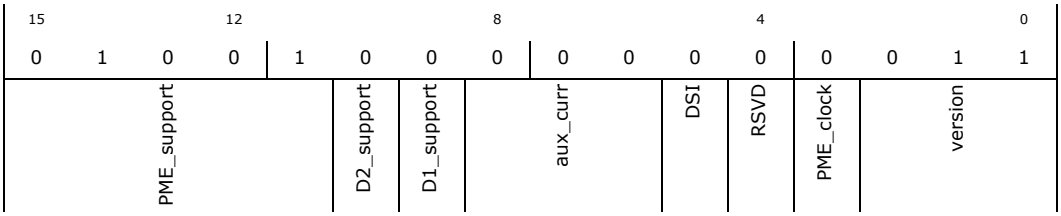
18.5.23 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:20, F:1] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported

Bit Range	Default & Access	Description
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

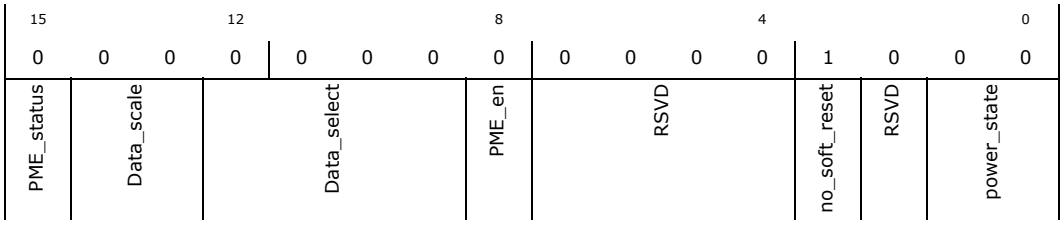
18.5.24 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:20, F:1] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

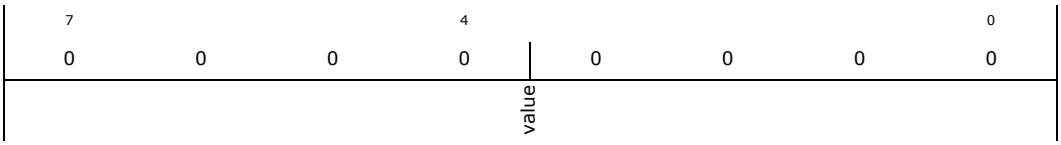
18.5.25 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:20, F:1] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

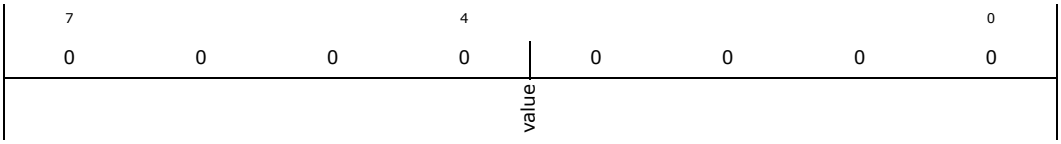
18.5.26 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:20, F:1] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

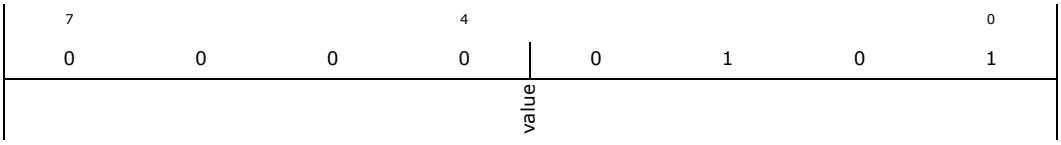
18.5.27 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:20, F:1] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

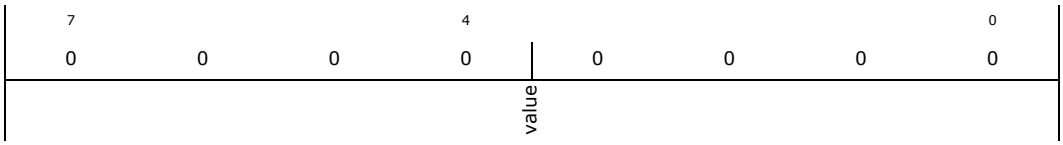
18.5.28 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:20, F:1] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

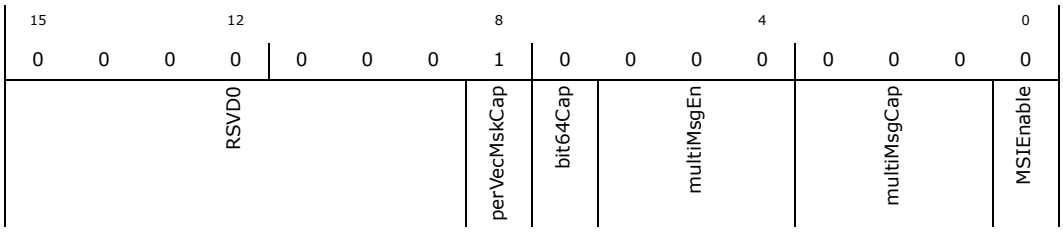
18.5.29 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:20, F:1] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

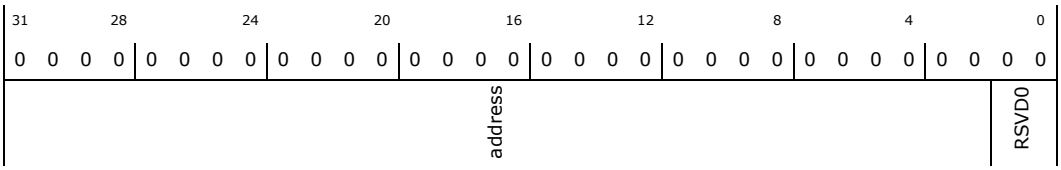
18.5.30 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:20, F:1] + A4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

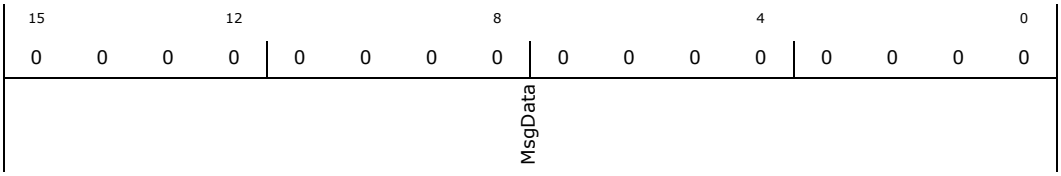
18.5.31 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:20, F:1] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

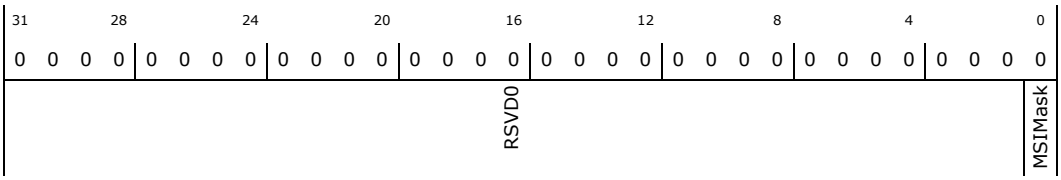
18.5.32 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:20, F:1] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

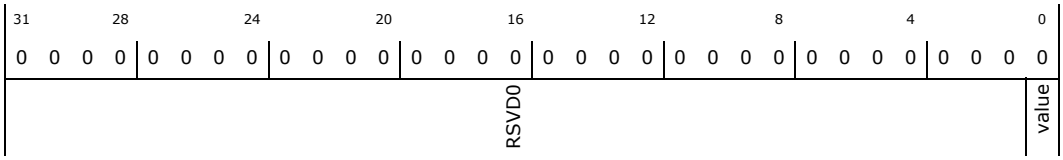
18.5.33 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:20, F:1] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

18.6 Memory Mapped Registers

18.6.1 UART Registers

Table 100. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Receive Buffer / Transmit Holding / Divisor Latch Low (RBR_THR_DLL)—Offset 0h" on page 671	00000000h
4h	7h	"Interrupt Enable / Divisor Latch High (IER_DLH)—Offset 4h" on page 671	00000000h
8h	Bh	"Interrupt Identification / FIFO Control (IIR_FCR)—Offset 8h" on page 672	00000001h
Ch	Fh	"Line Control (LCR)—Offset Ch" on page 674	00000000h
10h	13h	"MODEM Control (MCR)—Offset 10h" on page 674	00000000h
14h	17h	"Line Status (LSR)—Offset 14h" on page 675	00000060h
18h	1Bh	"MODEM Status (MSR)—Offset 18h" on page 677	00000000h
1Ch	1Fh	"Scratchpad (SCR)—Offset 1Ch" on page 678	00000000h
7Ch	7Fh	"UART Status (USR)—Offset 7Ch" on page 679	00000000h
A4h	A7h	"Halt Transmission (HTX)—Offset A4h" on page 679	00000000h
A8h	ABh	"DMA Software Acknowledge (DMASA)—Offset A8h" on page 680	00000000h

18.6.1.1 Receive Buffer / Transmit Holding / Divisor Latch Low (RBR_THR_DLL)—Offset 0h

Receive Buffer Register(RBR), reading this register when the DLAB bit (LCR[7]) is zero; Transmit Holding Register (THR), writing to this register when the DLAB is zero; Divisor Latch Low (DLL), when DLAB bit is one

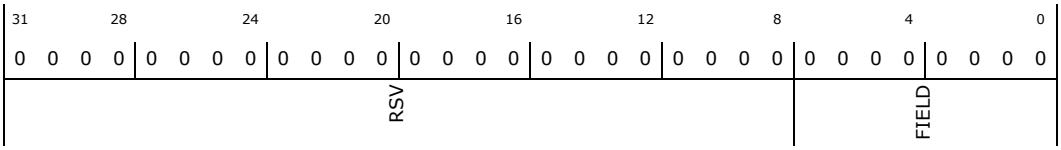
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	00h RW	**Receive Buffer / Transmit Holding / Divisor Latch Low (FIELD):** Different UART registers are accessed depending on read/write transfer type and Line control Register (LCR) DLAB bit value. RBR, Receive Buffer Register - Access - Read AND DLAB (LCR[7]) =0 Data byte received on the serial input port. Valid only if the Data Ready (DR) bit in the LSR is set. If FIFOs are disabled (FCR[0] set to zero), the data must be read before the next data arrives, otherwise it is overwritten, resulting in an over-run error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO is preserved, but any incoming data are lost and an over-run error occurs. THR, Transmit Holding Register. - Access - Write AND DLAB (LCR[7]) =0 Data to be transmitted on the serial output port. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFOs are disabled (FCR[0] = 0) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFOs are enabled (FCR[0] = 1) and THRE is set, a total of 16 characters (FIFO Depth) can be written to the THR before the FIFO is full. Any attempt to write data when the FIFO is full results in the write data being lost. DLL, Lower part of the Divisor Latch. - Access - Read/Write AND DLAB (LCR[7]) =1 Lower 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may be accessed only when the DLAB bit (LCR[7]) is set. Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of sclk should be allowed to pass before transmitting or receiving data.

18.6.1.2 Interrupt Enable / Divisor Latch High (IER_DLH)—Offset 4h

Interrupt Enable Register (IER), when the DLAB bit is zero; Divisor Latch High (DLH), when the DLAB bit is one

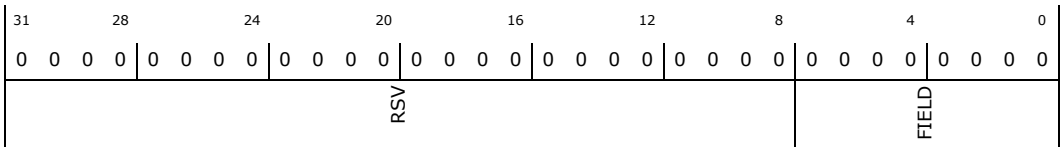
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	00h RW	**Interrupt Enable / Divisor Latch High (FIELD):** Different UART registers are accessed depending on the Line control Register (LCR) DLAB bit value. IER, Interrupt Enable Register Access - Read/write AND DLAB (LCR[7]) =0 Interrupt Enable Register: Each of the bits used has a different function (0 = disabled 1 = enabled): 0 - ERBFI, Enable Received Data Available Interrupt 1 - ETBEI, Enable Transmit Holding Register Empty Interrupt 2 - ELSI, Enable Receiver Line Status Interrupt 3 - EDSSI, Enable Modem Status Interrupt 4 - RESERVED 5 - RESERVED 6 - RESERVED 7 - PTIME, Programmable THRE Interrupt Mode Enable DLH, Divisor Latch (High) Access - Read/write AND DLAB (LCR[7]) =1 This register makes up the upper 8-bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register can be accessed only when the DLAB bit (LCR[7]) is set. NOTE: if the Divisor Latch Registers (DLL and DLH) are set to zero, the baud clock is disabled and no serial communications occur. Also, once the Divisor Latch Registers (DLL and DLH) are set, a mimimum of 50ns should be awaited before transmitting or receiving data as this time is required in order for changes to take effect due to internal synchronization processes.

18.6.1.3 Interrupt Identification / FIFO Control (IIR_FCR)—Offset 8h

Interrupt Identification Register (IIR) if reading; FIFO Control Register (FCR) if writing

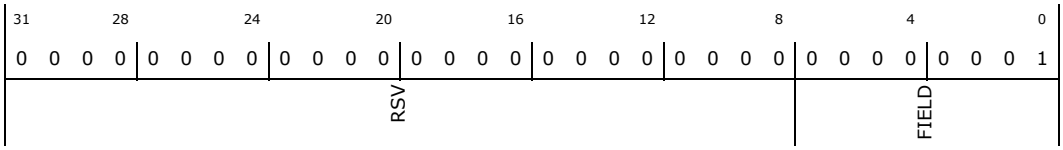
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000001h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	01h RW	**Interrupt Identification / FIFO Control (FIELD):** Different UART registers are accessed depending on read/write transfer type. IIR, Interrupt Identification Register - Access - Read only - Reset - 0x01 Each of the bits used has a different function: [3:0] - IID, Interrupt ID. This indicates the highest priority pending interrupt which can be one of the following types: 0000 = modem status. 0001 = no interrupt pending. 0010 = THR empty. 0100 = received data available. 0110 = receiver line status. 0111 = busy detect. NEVER INDICATED 1100 = character timeout. [5:3] - RESERVED read as zero [7:6] - FIFOSE, FIFOs Enabled. This is used to indicate whether the FIFO's are enabled or disabled: 00 = disabled 11 = enabled FCR, FIFO Control Register Access - Write only Used to control the FIFOs. Different functions: [0] - FIFOE, FIFO Enable. Enables/disables the transmit (XMIT) and receive (RCVR) FIFO's. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFO's will be reset. [1] - RFIFOR, RCVR FIFO Reset Resets the control portion of the receive FIFO and treats the FIFO as empty. This will also de-assert the DMA RX request and single signals. NOTE that this bit is 'self-clearing' and it is not necessary to clear this bit. [2] - XFIFOR, XMIT FIFO Reset Resets the control portion of the transmit FIFO and treats the FIFO as empty. This will also de-assert the DMA TX request and single signals. NOTE that this bit is 'self-clearing' and it is not necessary to clear this bit. [3] - DMAM, DMA Mode Not used in UART due to the use of extra DMA handshake interface signals [5:4] - TET, TX Empty Trigger Used to select the empty threshold level at which the THRE Interrupts will be generated when the mode is active. It also determines when the DMA support is requested if the DMA is enabled. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO 1/4 full 11 = FIFO 1/2 full [7:6] - RT, RCVR Trigger Used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt will be generated. In auto flow control mode it is used to determine when the rts_n signal will be de-asserted. It also determines when the DMA support is requested if the DMA is enabled. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO 1/4 full 10 = FIFO 1/2 full 11 = FIFO 2 less than full

18.6.1.4 Line Control (LCR)—Offset Ch

Used to specify the format of the asynchronous data communication exchange.

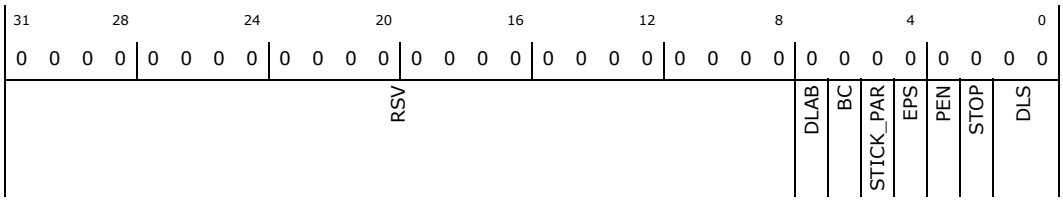
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7	0h RW	**Divisor Latch Access Bit (DLAB):** Used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.
6	0h RW	**Break Control Bit (BC):** Used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the serial output line is forced low until the Break bit is cleared.
5	0h RW	**Reserved for future use (STICK_PAR):** Reserved.
4	0h RW	**Even Parity Select (EPS):** Used to select between even and odd parity, when parity is enabled (PEN set to one). If set to one, an even number of logic '1's is transmitted or checked. If set to zero, an odd number of logic '1's is transmitted or checked.
3	0h RW	**Parity Enable (PEN):** Used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0 = parity disabled 1 = parity enabled
2	0h RW	**Number of stop bits (STOP):** Used to select the number of stop bits per character that the peripheral will transmit and receive. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits are transmitted. Otherwise, two stop bits are transmitted. NOTE: regardless of the number of stop bits selected the receiver will only check the first stop bit. 0 = 1 stop bit 1 = 1.5 stop bits when DLS (LCR[1:0]) == 00 1 = 2 stop bits when DLS (LCR[1:0]) different from 00
1:0	0h RW	**Data Length Select (DLS):** Used to select the number of data bits per character that the peripheral will transmit and receive. The number of bit that may be selected are as follows: 00 = 5 bits 01 = 6 bits 10 = 7 bits 11 = 8 bits

18.6.1.5 MODEM Control (MCR)—Offset 10h

Controls the interface with the MODEM or data set (or a peripheral device emulating a MODEM)

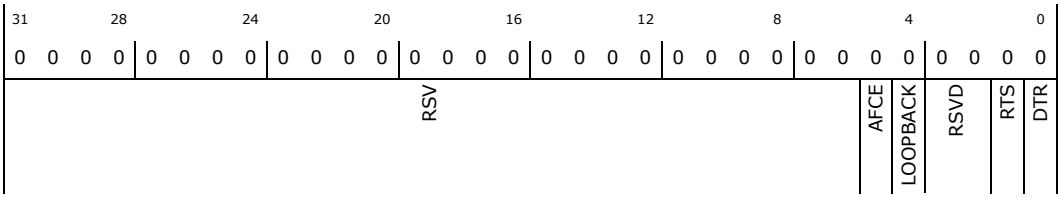
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5	0h RW	**Auto Flow Control Enable (AFCE):** When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, Auto Flow Control features are enabled 0 = Auto Flow Control Mode disabled 1 = Auto Flow Control Mode enabled
4	0h RW	**LoopBack Bit (LOOPBACK):** Used to put the UART into a diagnostic mode for test purposes. Data on the serial out line is held high, while serial data output is looped back to the serial in line, internally. In this mode all the interrupts are fully functional. Also, in loopback mode, the modem control inputs (dsr_n, cts_n, ri_n, dcd_n) are disconnected and the modem control outputs (dtr_n, rts_n, out1_n, out2_n) are looped back to the inputs, internally.
3:2	0h RO	**Reserved (RSVD):** Reserved.
1	0h RW	**Request to Send (RTS):** Used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto RTS Flow Control is not enabled (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. In Auto Flow Control, MCR[5] set to one and FIFO's enable (FCR[0] set to one, the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal will be de-asserted when MCR[1] is set low. Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive high while the value of this location is internally looped back to an input.
0	0h RW	**Data Terminal Ready (DTR):** Used to directly control the Data Terminal Ready (dtr_n) output. The value written to this location is inverted and driven out on dtr_n, that is: 0 = dtr_n de-asserted (logic 1) 1 = dtr_n asserted (logic 0) The Data Terminal Ready output is used to inform the modem or data set that the UART is ready to establish communications. Note that in Loopback mode (MCR[4] set to one), the dtr_n output is held inactive high while the value of this location is internally looped back to an input.

18.6.1.6 Line Status (LSR)—Offset 14h

Provides status information concerning the data transfer.

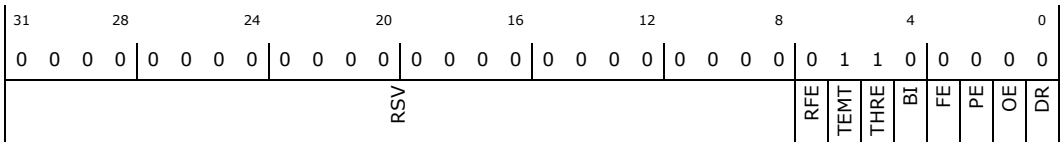
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000060h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7	0h RO	**Receiver FIFO Error bit (RFE):** This bit is only relevant when FIFO are enabled (FCR[0] set to one). This is used to indicate if there is at least one parity error, framing error, or break indication in the FIFO. That is: 0 = no error in RX FIFO 1 = error in RX FIFO This bit is cleared when the LSR is read and the character with the error is at the top of the receiver FIFO and there are no subsequent errors in the FIFO.
6	1h RO	**Transmitter Empty bit (TEMT):** If FIFO are enabled (FCR[0] set to one), this bit is set whenever the Transmitter Shift Register and the FIFO are both empty. If FIFO are disabled, this bit is set whenever the Transmitter Holding Register and the Transmitter Shift Register are both empty.
5	1h RO	**Transmit Holding Register Empty bit (THRE):** If THRE mode is disabled (IER[7] set to zero) and regardless of FIFO's being enabled or not, this bit indicates that the THR or TX FIFO is empty. This bit is set whenever data is transferred from the THR or TX FIFO to the transmitter shift register and no new data has been written to the THR or TX FIFO. This also causes a THRE Interrupt to occur, if the THRE Interrupt is enabled. If the THRE mode and FIFO are enabled (IER[7] and FCR[0] set to one), the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the FCR[5:4] threshold setting.
4	0h RO	**Break Interrupt bit (BI):** Used to indicate the detection of a break sequence on the serial input data. If in UART mode it is set whenever the serial input is held in a logic '0' state for longer than the sum of start time + data bits + parity + stop bits. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.
3	0h RO	**Framing Error bit (FE):** Used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data. In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs the UART will try resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) will be set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no framing error, 1 = framing error Reading the LSR clears the FE bit.
2	0h RO	**Parity Error bit (PE):** Used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) will be set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no parity error, 1 = parity error Reading the LSR clears the PE bit.

Bit Range	Default & Access	Field Name (ID): Description
1	0h RO	**Overrun error bit (OE):** Used to indicate the occurrence of an overrun error. This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost. 0 = no overrun error, 1 = overrun error Reading the LSR clears the OE bit.
0	0h RO	**Data Ready bit (DR):** Used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0 = no data ready, 1 = data ready This bit is cleared when the RBR is read in the non-FIFO mode, or when the receiver FIFO is empty, in the FIFO mode.

18.6.1.7 MODEM Status (MSR)—Offset 18h

Provides the current state of the control lines from the MODEM (or peripheral device)

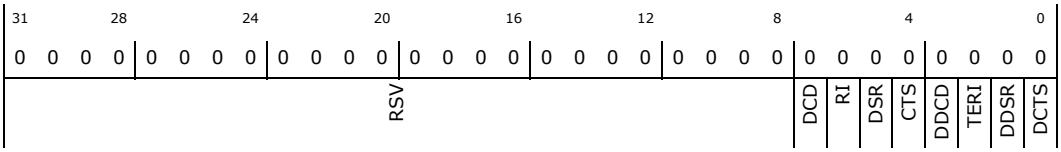
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7	0h RO	**Data Carrier Detect (DCD):** Used to indicate the current state of the modem control line dcd_n. That is this bit is the complement dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set. 0 = dcd_n input is de-asserted (logic 1) 1 = dcd_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DCD is the same as MCR[3] (Out2).
6	0h RO	**Ring Indicator (RI):** Used to indicate the current state of the modem control line ri_n. That is this bit is the complement ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0 = ri_n input is de-asserted (logic 1) 1 = ri_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), RI is the same as MCR[2] (Out1).
5	0h RO	**Data Set Ready (DSR):** Used to indicate the current state of the modem control line dsr_n. That is this bit is the complement dsr_n. When the Data Set Ready input (dsr_n) is asserted it is an indication that the modem or data set is ready to establish communications with the UART. 0 = dsr_n input is de-asserted (logic 1) 1 = dsr_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DSR is the same as MCR[0] (DTR).

Bit Range	Default & Access	Field Name (ID): Description
4	0h RO	**Clear to Send (CTS):** Used to indicate the current state of the modem control line cts_n. That is, this bit is the complement cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with the UART 0 = cts_n input is de-asserted (logic 1) 1 = cts_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), CTS is the same as MCR[1] (RTS).
3	0h RO	**Delta Data Carrier Detect (DDCD):** Used to indicate that the modem control line dcd_n has changed since the last time the MSR was read. That is: 0 = no change on dcd_n since last read of MSR 1 = change on dcd_n since last read of MSR Reading the MSR clears the DDCD bit. In Loopback Mode (MCR[4] set to one), DDCD reflects changes on MCR[3] (Out2). Note, if the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit will get set when the reset is removed if the dcd_n signal remains asserted.
2	0h RO	**Trailing Edge of Ring Indicator (TERI):** Used to indicate that a change on the input ri_n (from an active low, to an inactive high state) has occurred since the last time the MSR was read. That is: 0 = no change on ri_n since last read of MSR 1 = change on ri_n since last read of MSR Reading the MSR clears the TERI bit. In Loopback Mode (MCR[4] set to one), TERI reflects when MCR[2] (Out1) has changed state from a high to a low.
1	0h RO	**Delta Data Set Ready (DDSR):** Used to indicate that the modem control line dsr_n has changed since the last time the MSR was read. That is: 0 = no change on dsr_n since last read of MSR 1 = change on dsr_n since last read of MSR Reading the MSR clears the DDSR bit. In Loopback Mode (MCR[4] set to one), DDSR reflects changes on MCR[0] (DTR). Note, if the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDSR bit will get set when the reset is removed if the dsr_n signal remains asserted.
0	0h RO	**Delta Clear to Send (DCTS):** used to indicate that the modem control line cts_n has changed since the last time the MSR was read. That is: 0 = no change on cts_n since last read of MSR 1 = change on cts_n since last read of MSR Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] set to one), DCTS reflects changes on MCR[1] (RTS). Note, if the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit will get set when the reset is removed if the cts_n signal remains asserted.

18.6.1.8 Scratchpad (SCR)—Offset 1Ch

Used by the programmer to hold data temporarily

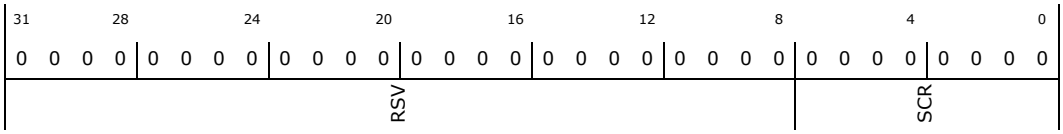
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0h RW	**Scratchpad Register (SCR):** This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART.

18.6.1.9 UART Status (USR)—Offset 7Ch

Provides internal status information

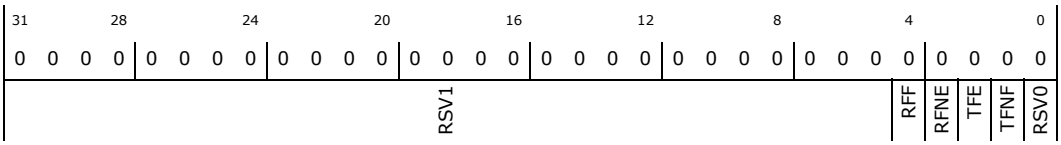
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 7Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	**Reserved (RSV1):** Reserved.
4	0h RO	**Receive FIFO Full (RFF):** Used to indicate that the receive FIFO is completely full. That is: 0 = Receive FIFO not full 1 = Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.
3	0h RO	**Receive FIFO Not Empty (RFNE):** Used to indicate that the receive FIFO contains one or more entries. 0 = Receive FIFO is empty 1 = Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.
2	0h RO	**Transmit FIFO Empty (TFE):** Used to indicate that the transmit FIFO is completely empty. 0 = Transmit FIFO is not empty 1 = Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty.
1	0h RO	**Transmit FIFO Not Full (TFNF):** Used to indicate that the transmit FIFO in not full. 0 = Transmit FIFO is full 1 = Transmit FIFO is not full This bit is cleared when the TX FIFO is full.
0	0h RO	**Reserved (RSV0):** Reserved.

18.6.1.10 Halt Transmission (HTX)—Offset A4h

Halt Transmission

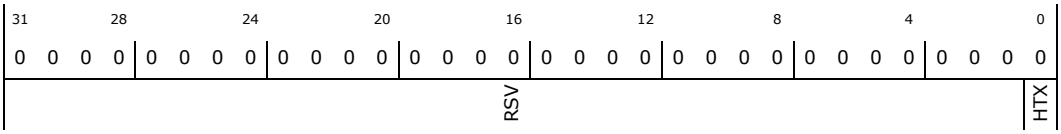
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + A4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0h RW	**Halt Transmission (HTX):** Used to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFO's are enabled. Note, if FIFO's are not enabled the setting of the halt TX register will have no effect on operation. 0 = Halt TX disabled 1 = Halt TX enabled

18.6.1.11 DMA Software Acknowledge (DMASA)—Offset A8h

DMA software acknowledge if a transfer needs to be terminated due to an error condition.

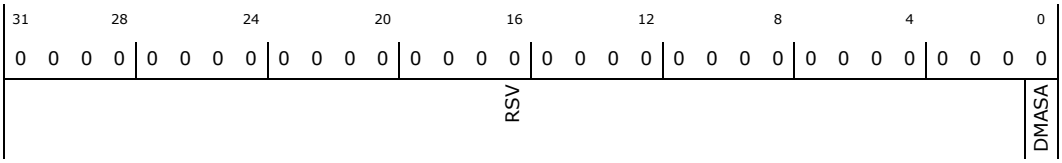
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + A8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:20, F:1] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0h RW	**DMA Software Acknowledge (DMASA):** Used to perform DMA software acknowledge if a transfer needs to be terminated due to an error condition. For example, if the DMA disables the channel, then the UART should clear its request. This will cause the DMA operation to be concluded. NOTE: this bit is 'self-clearing'. It is not necessary to clear it.

18.6.2 DMA Controller Registers

Table 101. Summary of Memory Mapped I/O Registers—BAR1

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Channel 0 Source Address (SAR0)—Offset 0h" on page 682	00000000h
8h	Bh	"Channel 0 Destination Address (DAR0)—Offset 8h" on page 682	00000000h
10h	13h	"Channel 0 Linked List Pointer (LLP0)—Offset 10h" on page 683	00000000h
18h	1Bh	"Channel 0 Control LOWER (CTL0_L)—Offset 18h" on page 683	00304801h
1Ch	1Fh	"Channel 0 Control UPPER (CTL0_U)—Offset 1Ch" on page 685	00000002h
20h	23h	"Channel 0 Source Status (SSTAT0)—Offset 20h" on page 686	00000000h
28h	2Bh	"Channel 0 Destination Status (DSTAT0)—Offset 28h" on page 686	00000000h
30h	33h	"Channel 0 Source Status Address (SSTATAR0)—Offset 30h" on page 687	00000000h

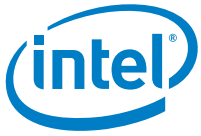


Table 101. Summary of Memory Mapped I/O Registers—BAR1 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
38h	3Bh	"Channel 0 Destination Status Address (DSTATAR0)—Offset 38h" on page 687	00000000h
40h	43h	"Channel 0 Configuration LOWER (CFG0_L)—Offset 40h" on page 688	00000E00h
44h	47h	"Channel 0 configuration UPPER (CFG0_U)—Offset 44h" on page 689	00000004h
48h	4Bh	"Channel 0 Source Gather (SGR0)—Offset 48h" on page 690	00000000h
50h	53h	"Channel 0 Destination Scatter (DSR0)—Offset 50h" on page 691	00000000h
58h	5Bh	"Channel 1 Source Address (SAR1)—Offset 58h" on page 691	00000000h
60h	63h	"Channel 1 Destination Address (DAR1)—Offset 60h" on page 692	00000000h
68h	6Bh	"Channel 1 Linked List Pointer (LLP1)—Offset 68h" on page 692	00000000h
70h	73h	"Channel 1 Control LOWER (CTL1_L)—Offset 70h" on page 693	00304801h
74h	77h	"Channel 1 Control UPPER (CTL1_U)—Offset 74h" on page 695	00000002h
78h	7Bh	"Channel 1 Source Status (SSTAT1)—Offset 78h" on page 696	00000000h
80h	83h	"Channel 1 Destination Status (DSTAT1)—Offset 80h" on page 696	00000000h
88h	8Bh	"Channel 1 Source Status Address (SSTATAR1)—Offset 88h" on page 697	00000000h
90h	93h	"Channel 1 Destination Status Address (DSTATAR1)—Offset 90h" on page 697	00000000h
98h	9Bh	"Channel 1 Configuration LOWER (CFG1_L)—Offset 98h" on page 698	00000E20h
9Ch	9Fh	"Channel 1 configuration UPPER (CFG1_U)—Offset 9Ch" on page 699	00000004h
A0h	A3h	"Channel 1 Source Gather (SGR1)—Offset A0h" on page 700	00000000h
A8h	ABh	"Channel 1 Destination Scatter (DSR1)—Offset A8h" on page 700	00000000h
2C0h	2C3h	"Raw Status for IntTfr Interrupt (RAW_TFR)—Offset 2C0h" on page 701	00000000h
2C8h	2CBh	"Raw Status for IntBlock Interrupt (RAW_BLOCK)—Offset 2C8h" on page 701	00000000h
2D0h	2D3h	"Raw Status for IntSrcTran Interrupt (RAW_SRC_TRAN)—Offset 2D0h" on page 702	00000000h
2D8h	2DBh	"Raw Status for IntDstTran Interrupt (RAW_DST_TRAN)—Offset 2D8h" on page 702	00000000h
2E0h	2E3h	"Raw Status for IntErr Interrupt (RAW_ERR)—Offset 2E0h" on page 703	00000000h
2E8h	2EBh	"Status for IntTfr Interrupt (STATUS_TFR)—Offset 2E8h" on page 703	00000000h
2F0h	2F3h	"Status for IntBlock Interrupt (STATUS_BLOCK)—Offset 2F0h" on page 704	00000000h
2F8h	2FBh	"Status for IntSrcTran Interrupt (STATUS_SRC_TRAN)—Offset 2F8h" on page 704	00000000h
300h	303h	"Status for IntDstTran Interrupt (STATUS_DST_TRAN)—Offset 300h" on page 705	00000000h
308h	30Bh	"Status for IntErr Interrupt (STATUS_ERR)—Offset 308h" on page 705	00000000h
310h	313h	"Mask for IntTfr Interrupt (MASK_TFR)—Offset 310h" on page 706	00000000h
318h	31Bh	"Mask for IntBlock Interrupt (MASK_BLOCK)—Offset 318h" on page 706	00000000h
320h	323h	"Mask for IntSrcTran Interrupt (MASK_SRC_TRAN)—Offset 320h" on page 707	00000000h
328h	32Bh	"Mask for IntDstTran Interrupt (MASK_DST_TRAN)—Offset 328h" on page 708	00000000h
330h	333h	"Mask for IntErr Interrupt (MASK_ERR)—Offset 330h" on page 708	00000000h
338h	33Bh	"Clear for IntTfr Interrupt (CLEAR_TFR)—Offset 338h" on page 709	00000000h
340h	343h	"Clear for IntBlock Interrupt (CLEAR_BLOCK)—Offset 340h" on page 709	00000000h
348h	34Bh	"Clear for IntSrcTran Interrupt (CLEAR_SRC_TRAN)—Offset 348h" on page 710	00000000h
350h	353h	"Clear for IntDstTran Interrupt (CLEAR_DST_TRAN)—Offset 350h" on page 710	00000000h
358h	35Bh	"Clear for IntErr Interrupt (CLEAR_ERR)—Offset 358h" on page 711	00000000h
360h	363h	"Combined Interrupt Status (STATUS_INT)—Offset 360h" on page 711	00000000h
368h	36Bh	"Source Software Transaction Request (REQ_SRC_REG)—Offset 368h" on page 712	00000000h

Table 101. Summary of Memory Mapped I/O Registers—BAR1 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
370h	373h	"Destination Software Transaction Request register (REQ_DST_REG)—Offset 370h" on page 712	00000000h
378h	37Bh	"Source Single Transaction Request (SGL_REQ_SRC_REG)—Offset 378h" on page 713	00000000h
380h	383h	"Destination Single Software Transaction Request (SGL_REQ_DST_REG)—Offset 380h" on page 714	00000000h
388h	38Bh	"Source Last Transaction Request (LST_SRC_REG)—Offset 388h" on page 714	00000000h
390h	393h	"Destination Single Transaction Request (LST_DST_REG)—Offset 390h" on page 715	00000000h
398h	39Bh	"DMA Configuration (DMA_CFG_REG)—Offset 398h" on page 715	00000000h
3A0h	3A3h	"Channel Enable (CH_EN_REG)—Offset 3A0h" on page 716	00000000h

18.6.2.1 Channel 0 Source Address (SAR0)—Offset 0h

Source Address of DMA transfer The starting source address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the source address of the current transfer.

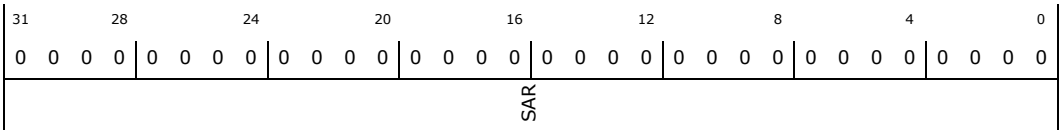
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Current Source Address of DMA transfer (SAR):** Updated after each source transfer. The SINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every source transfer throughout the block transfer. NOTE: Channel 0 is dedicated to the UART controller Received data. Source Address is the UART controller RBR_THR_DLL register address: 0xFFFF_F000

18.6.2.2 Channel 0 Destination Address (DAR0)—Offset 8h

Destination address of DMA transfer The starting destination address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the destination address of the current transfer.

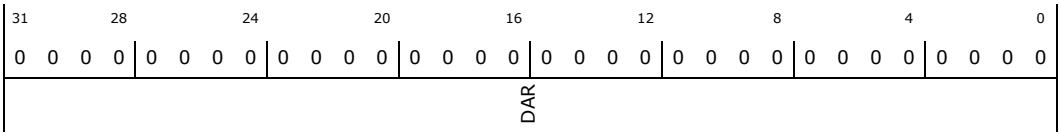
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Current Destination address of DMA transfer (DAR):** Updated after each destination transfer. The DINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every destination transfer throughout the block transfer. NOTE: Channel 0 is dedicated to the UART controller Received data. Destination Address is a memory address

18.6.2.3 Channel 0 Linked List Pointer (LLP0)—Offset 10h

Program this register to point to the first Linked List Item (LLI) in memory prior to enabling the channel if block chaining is enabled

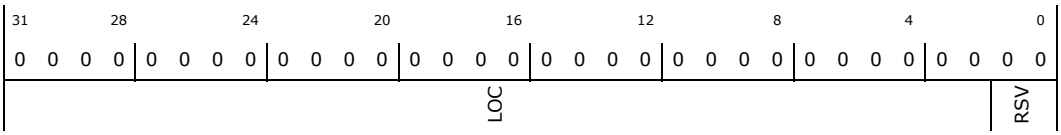
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 10h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RW	**Starting Address In Memory (LOC):** Starting Address In Memory of next LLI if block chaining is enabled. Note that the two LSBs of the starting address are not stored because the address is assumed to be aligned to a 32-bit boundary. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
1:0	0b RO	**Reserved (RSV):** Reserved.

18.6.2.4 Channel 0 Control LOWER (CTL0_L)—Offset 18h

Contains fields that control the DMA transfer. Part of the block descriptor (linked list item - LLI) when block chaining is enabled. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled.

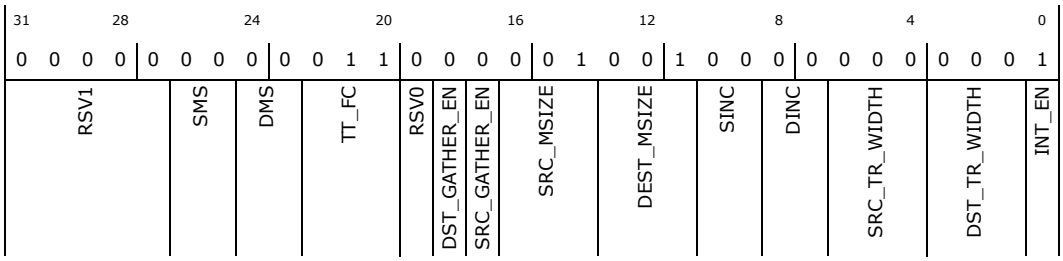
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 18h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00304801h



Bit Range	Default & Access	Field Name (ID): Description
31:27	0b RO	**Reserved (RSV1):** Reserved.
26:25	0b RO	**Source AMBA Layer (SMS):** Hardcoded the Master interface attached to the source of channel 0.
24:23	0b RO	**Destination AMBA Layer (DMS):** Hardcoded the Master interface attached to the destination of channel 0
22:20	011b RW	**Transfer Type and Flow Control (TT_FC):** The following transfer types are supported: Code - Type - Flow Controller --- 000 - Memory to Memory - DMAC 001 - Memory to Peripheral - DMAC 010 - Peripheral to Memory - DMAC 011 - Peripheral to Peripheral - DMAC 100 - Peripheral to Memory - Peripheral 101 - Peripheral to Peripheral - Source Peripheral 110 - Memory to Peripheral - Peripheral 111 - Peripheral to Peripheral - Destination Peripheral
19	0b RO	**Reserved (RSV0):** Reserved.
18	0b RW	**Destination scatter enable (DST_GATHER_EN):** 0 = Scatter disabled 1 = Scatter enabled Scatter on the destination side is applicable only when the CTL0_L.DINC bit indicates an incrementing or decrementing address control.
17	0b RW	**Source gather enable (SRC_GATHER_EN):** 0 = Gather disabled 1 = Gather enabled Gather on the source side is applicable only when the CTL0_L.SINC bit indicates an incrementing or decrementing address control.
16:14	001b RW	**Source Burst Transaction Length (SRC_MSIZE):** Number of data items, each of width CTL0_L.SRC_TR_WIDTH, to be read from the source every time a source burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WITDH) --- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256

Bit Range	Default & Access	Field Name (ID): Description
13:11	001b RW	**Destination Burst Transaction Length (DEST_MSIZE):** Destination Burst Transaction Length. Number of data items, each of width CTL0_L.DST_TR_WIDTH, to be written to the destination every time a destination burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WITDH) --- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
10:9	0b RW	**Source Address Increment (SINC):** Indicates whether to increment or decrement the source address on every source transfer. If the device is fetching data from a source peripheral FIFO with a fixed address, then set this field to No change. 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next SRC_TR_WIDTH boundary.
8:7	0b RW	**Destination Address Increment (DINC):** Indicates whether to increment or decrement the destination address on every destination transfer. If your device is writing data to a destination peripheral FIFO with a fixed address, then set this field to No change 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next DST_TR_WIDTH boundary.
6:4	0b RW	**Source transfer width (SRC_TR_WIDTH):** Decoding for this field: Value - Size(bits) --- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
3:1	0b RW	**Destination transfer width (DST_TR_WIDTH):** Decoding for this field: Value - Size(bits) --- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
0	1b RW	**Interrupt enable (INT_EN):** If set, then all interrupt-generating sources are enabled. Functions as a global mask bit for all interrupts for the channel. RAW interrupt registers still assert if INT_EN = 0.

18.6.2.5 Channel 0 Control UPPER (CTL0_U)—Offset 1Ch

Contains fields that control the DMA transfer. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled. If status write-back is enabled, the content is written to the control register location of the LLI in system memory at the end of the block transfer.

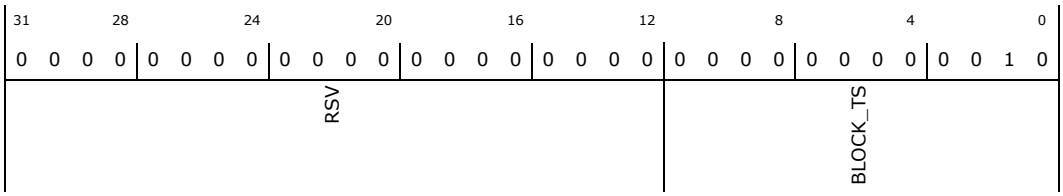
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 1Ch

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000002h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV):** Reserved.
11:0	002h RW	**Block length (BLOCK_TS):** When the DMAC is the flow controller, the user writes this field before the channel is enabled in order to indicate the block size. The number programmed into BLOCK_TS indicates the total number of single transactions to perform for every block transfer. A single transaction is mapped to a single AMBA beat. Width: The width of the single transaction is determined by CTL0_L.SRC_TR_WIDTH.

18.6.2.6 Channel 0 Source Status (SSTAT0)—Offset 20h

This register is a temporary placeholder for the source status information on its way to the SSTAT0 register location of the LLI. The source status information should be retrieved by software from the SSTAT0 register location of the LLI, and not by a read of this register over the DMAC slave interface.

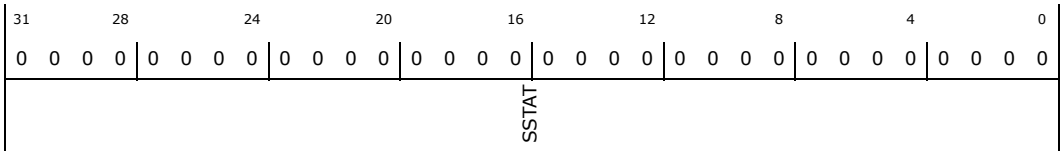
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 20h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Source Status (SSTAT):** Source status information retrieved by hardware from the address pointed to by the contents of the SSTATAR0 register.

18.6.2.7 Channel 0 Destination Status (DSTAT0)—Offset 28h

This register is a temporary placeholder for the destination status information on its way to the DSTAT0 register location of the LLI. The destination status information should be retrieved by software from the DSTAT0 register location of the LLI and not by a read of this register over the DMAC slave interface

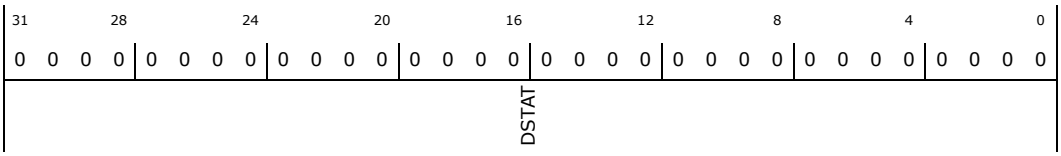
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 28h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Destination Status (DSTAT):** Destination status information retrieved by hardware from the address pointed to by the contents of the DSTATAR0 register

18.6.2.8 Channel 0 Source Status Address (SSTATAR0)—Offset 30h

After the completion of each block transfer, hardware can retrieve the source status information from the address pointed to by the contents of this register

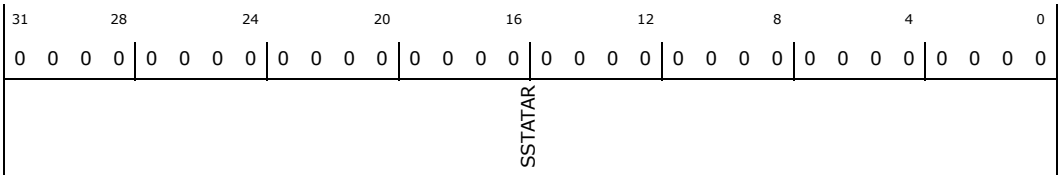
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 30h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Source Status Address (SSTATAR):** Pointer from where hardware can fetch the source status information, which is registered in the SSTAT0 register and written out to the SSTAT0 register location of the LLI before the start of the next block.

18.6.2.9 Channel 0 Destination Status Address (DSTATAR0)—Offset 38h

After the completion of each block transfer, hardware can retrieve the destination status information from the address pointed to by the contents of this register

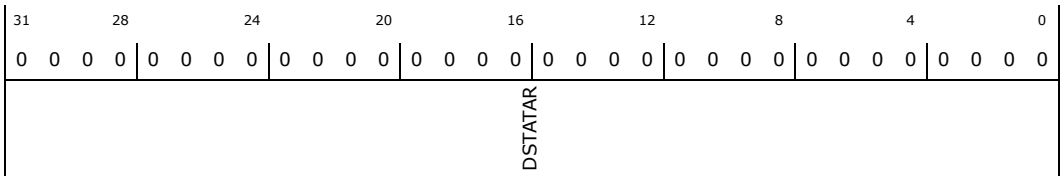
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 38h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Destination Status Address (DSTATAR):** Pointer from where hardware can fetch the destination status information, which is registered in the DSTAT0 register and written out to the DSTAT0 register location of the LLI before the start of the next block.

18.6.2.10 Channel 0 Configuration LOWER (CFG0_L)—Offset 40h

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. This register need to be programmed before enable the channel.

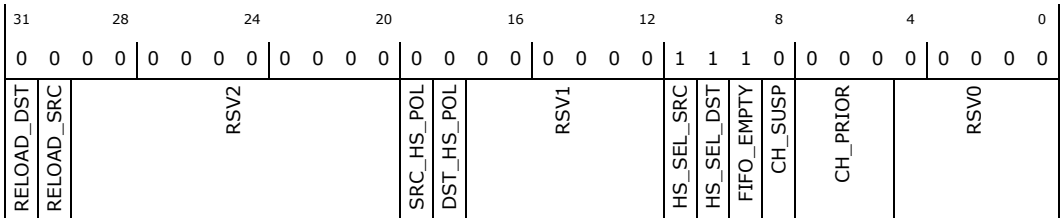
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 40h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000E00h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Reload destination enable (RELOAD_DST):** The DARx register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
30	0b RW	**Reload source enable (RELOAD_SRC):** The SAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
29:20	0b RO	**Reserved (RSV2):** Reserved.
19	0b RW	**Source handshake polarity (SRC_HS_POL):** 0 = Active high 1 = Active low
18	0b RW	**Destination handshake polarity (DST_HS_POL):** 0 = Active high 1 = Active low
17:12	0b RW	**Reserved (RSV1):** Reserved.
11	1b RW	**Source Handshake select (HS_SEL_SRC):** Used to select which handshake interface is active for source requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If source peripheral is memory this bit is ignored

Bit Range	Default & Access	Field Name (ID): Description
10	1b RW	**Destination Handshake select (HS_SEL_DST):** Used to select which handshake interface is active for destination requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If destination peripheral is memory this bit is ignored
9	1b RO	**Channel FIFO empty status (FIFO_EMPTY):** Indicates if there is data left in the channel FIFO. Can be used in conjunction with CH_SUSP to cleanly disable a channel. 1 = Channel FIFO empty 0 = Channel FIFO not empty
8	0b RW	**Channel Suspend control (CH_SUSP):** Suspends all DMA data transfers from the source until this bit is cleared. There is no guarantee that the current transaction will complete. Can also be used in conjunction with FIFO_EMPTY to cleanly disable a channel without losing any data. 0 = Not suspended. 1 = Suspend DMA transfer from the source.
7:5	0b RW	**Channel Priority (CH_PRIOR):** Priority value equal to 7 is the highest priority, and 0 is the lowest. This field must be programmed within the following range: 0: 1 A programmed value outside this range will cause erroneous behavior.
4:0	0b RO	**Reserved (RSV0):** Reserved.

18.6.2.11 Channel 0 configuration UPPER (CFG0_U)—Offset 44h

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. This register need to be programmed before enable the channel.

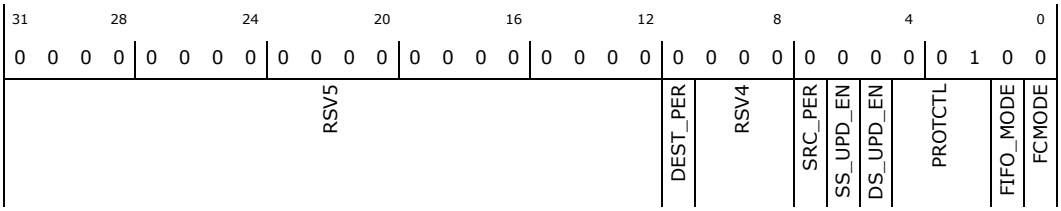
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 44h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000004h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV5):** Reserved.
11	0b RW	**Destination hardware interface (DEST_PER):** Assigns a hardware handshaking interface (0-1) to the channel destination if the CFG0_L.HS_SEL_DST field is 0; otherwise, this field is ignored. The channel can then communicate with the destination peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
10:8	0b RO	**Reserved (RSV4):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
7	0b RW	**Source hardware interface (SRC_PER):** Assigns a hardware handshaking interface (0-1) to the channel source if the CFG0_L.HS_SEL_SRC field is 0; otherwise, this field is ignored. The channel can then communicate with the source peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
6	0b RW	**Source Status Update Enable (SS_UPD_EN):** Source status information is fetched only from the location pointed to by the SSTATAR0 register, stored in the SSTAT0 register and written out to the SSTAT0 location of the LLI if this field is high
5	0b RW	**Destination Status Update Enable (DS_UPD_EN):** Destination status information is fetched only from the location pointed to by the DSTATAR0 register, stored in the DSTAT0 register and written out to the DSTAT0 location of the LLI if this field is high
4:2	001b RW	**AHB bus protocol bus control (PROTCTL):** Protection Control bits used to drive the AHB HPROT[3:1] bus. The AMBA Specification recommends that the default value of HPROT indicates a non-cached, non-buffered, privileged data access. The reset value is used to indicate such an access. HPROT[0] is tied high because all transfers are data accesses, as there are no opcode fetches. There is a one-to-one mapping of these register bits to the HPROT[3:1] master interface signals.
1	0b RW	**Channel FIFO mode control (FIFO_MODE):** Determines how much space or data needs to be available in the FIFO before a burst transaction request is serviced. 0 = Space/data available for single AHB transfer of the specified transfer width. 1 = Data available is greater than or equal to half the FIFO depth for destination transfers and space available is greater than half the fifo depth for source transfers. The exceptions are at the end of a burst transaction request or at the end of a block transfer.
0	0b RW	**Channel flow control mode (FCMODE):** Determines when source transaction requests are serviced when the Destination Peripheral is the flow controller. 0 = Source transaction requests are serviced when they occur. Data pre-fetching is enabled. 1 = Source transaction requests are not serviced until a destination transaction request occurs. In this mode, the amount of data transferred from the source is limited so that it is guaranteed to be transferred to the destination prior to block termination by the destination. Data pre-fetching is disabled.

18.6.2.12 Channel 0 Source Gather (SGR0)—Offset 48h

The CTL0_L.SINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

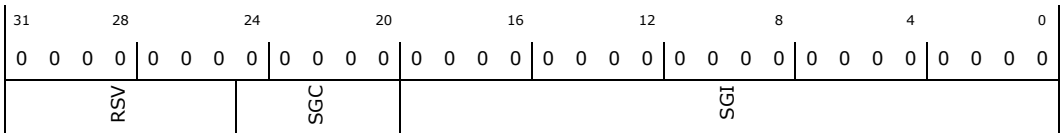
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 48h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0b RO	**Reserved (RSV):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
24:20	0b RW	**Source Gather Count (SGC):** Source contiguous transfer count between successive gather boundaries. Specifies the number of contiguous source transfers of CTL0_L.SRC_TR_WIDTH between successive gather intervals. This is defined as a gather boundary
19:0	0b RW	**Source Gather Interval (SGI):** Specifies the source address increment/decrement in multiples of CTL0_L.SRC_TR_WIDTH on a gather boundary when gather mode is enabled for the source transfer.

18.6.2.13 Channel 0 Destination Scatter (DSR0)—Offset 50h

The CTL0_L.DINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

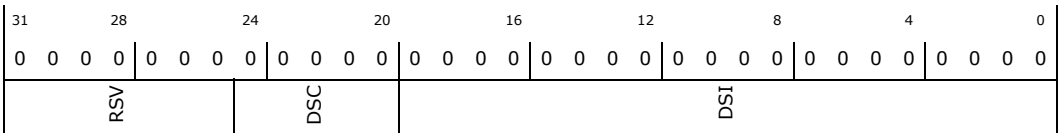
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 50h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0b RO	**Reserved (RSV):** Reserved.
24:20	0b RW	**Destination Scatter Count (DSC):** Source contiguous transfer count between successive scatter boundaries. Specifies the number of contiguous destination transfers of CTL0_L.DST_TR_WIDTH between successive scatter intervals. This is defined as a scatter boundary
19:0	0b RW	**Destination Scatter Interval (DSI):** Specifies the destination address increment/ decrement in multiples of CTL0_L.DST_TR_WIDTH on a scatter boundary when scatter mode is enabled for the destination transfer.

18.6.2.14 Channel 1 Source Address (SAR1)—Offset 58h

Source Address of DMA transfer The starting source address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the source address of the current transfer.

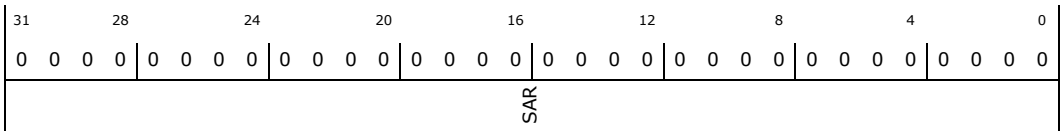
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 58h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Current Source Address of DMA transfer (SAR):** Updated after each source transfer. The SINC field in the CTL1_L register determines whether the address increments, decrements, or is left unchanged on every source transfer throughout the block transfer. NOTE: Channel 1 is dedicated to the UART controller Transmitted data. Source Address is a memory address

18.6.2.15 Channel 1 Destination Address (DAR1)—Offset 60h

Destination address of DMA transfer The starting destination address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the destination address of the current transfer.

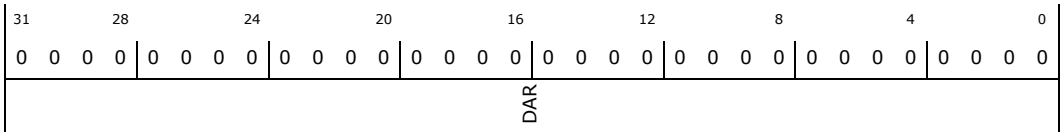
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 60h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Current Destination address of DMA transfer (DAR):** Updated after each destination transfer. The DINC field in the CTL1_L register determines whether the address increments, decrements, or is left unchanged on every destination transfer throughout the block transfer. NOTE: Channel 0 is dedicated to the UART controller Transmitted data. Destination Address is the UART controller RBR_THR_DLL register address: 0xFFFF_F000

18.6.2.16 Channel 1 Linked List Pointer (LLP1)—Offset 68h

Program this register to point to the first Linked List Item (LLI) in memory prior to enabling the channel if block chaining is enabled

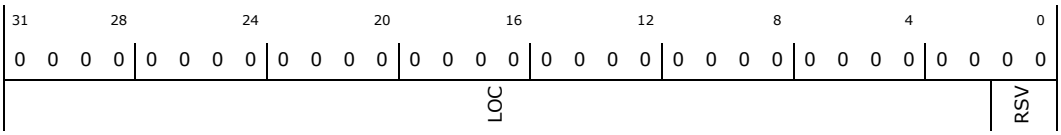
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 68h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RW	**Starting Address In Memory (LOC):** Starting Address In Memory of next LLI if block chaining is enabled. Note that the two LSBs of the starting address are not stored because the address is assumed to be aligned to a 32-bit boundary. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
1:0	0b RO	**Reserved (RSV):** Reserved.

18.6.2.17 Channel 1 Control LOWER (CTL1_L)—Offset 70h

Contains fields that control the DMA transfer. Part of the block descriptor (linked list item - LLI) when block chaining is enabled. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled.

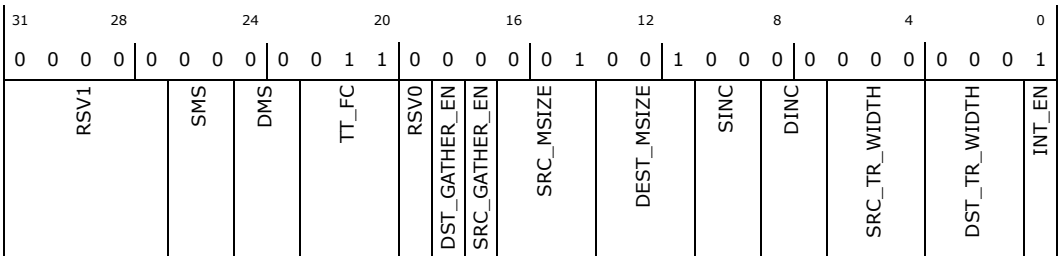
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 70h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00304801h



Bit Range	Default & Access	Field Name (ID): Description
31:27	0b RO	**Reserved (RSV1):** Reserved.
26:25	0b RO	**Source AMBA Layer (SMS):** Hardcoded the Master interface attached to the source of channel 1
24:23	0b RO	**Destination AMBA Layer (DMS):** Hardcoded the Master interface attached to the destination of channel 1

Bit Range	Default & Access	Field Name (ID): Description
22:20	011b RW	**Transfer Type and Flow Control (TT_FC):** The following transfer types are supported: Code - Type - Flow Controller --- 000 - Memory to Memory - DMAC 001 - Memory to Peripheral - DMAC 010 - Peripheral to Memory - DMAC 011 - Peripheral to Peripheral - DMAC 100 - Peripheral to Memory - Peripheral 101 - Peripheral to Peripheral - Source Peripheral 110 - Memory to Peripheral - Peripheral 111 - Peripheral to Peripheral - Destination Peripheral
19	0b RO	**Reserved (RSV0):** Reserved.
18	0b RW	**Destination scatter enable (DST_GATHER_EN):** 0 = Scatter disabled 1 = Scatter enabled Scatter on the destination side is applicable only when the CTL1_L.DINC bit indicates an incrementing or decrementing address control.
17	0b RW	**Source gather enable (SRC_GATHER_EN):** 0 = Gather disabled 1 = Gather enabled Gather on the source side is applicable only when the CTL1_L.SINC bit indicates an incrementing or decrementing address control.
16:14	001b RW	**Source Burst Transaction Length (SRC_MSIZE):** Number of data items, each of width CTL1_L.SRC_TR_WIDTH, to be read from the source every time a source burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WITDH) --- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
13:11	001b RW	**Destination Burst Transaction Length (DEST_MSIZE):** Destination Burst Transaction Length. Number of data items, each of width CTL1_L.DST_TR_WIDTH, to be written to the destination every time a destination burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WITDH) --- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
10:9	0b RW	**Source Address Increment (SINC):** Indicates whether to increment or decrement the source address on every source transfer. If the device is fetching data from a source peripheral FIFO with a fixed address, then set this field to No change. 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next SRC_TR_WIDTH boundary.

Bit Range	Default & Access	Field Name (ID): Description
8:7	0b RW	**Destination Address Increment (DINC):** Indicates whether to increment or decrement the destination address on every destination transfer. If your device is writing data to a destination peripheral FIFO with a fixed address, then set this field to No change 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next DST_TR_WIDTH boundary.
6:4	0b RW	**Source transfer width (SRC_TR_WIDTH):** Decoding for this field: Value - Size(bits) ---------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
3:1	0b RW	**Destination transfer width (DST_TR_WIDTH):** Decoding for this field: Value - Size(bits) ---------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
0	1b RW	**Interrupt enable (INT_EN):** If set, then all interrupt-generating sources are enabled. Functions as a global mask bit for all interrupts for the channel. RAW interrupt registers still assert if INT_EN = 0.

18.6.2.18 Channel 1 Control UPPER (CTL1_U)—Offset 74h

Contains fields that control the DMA transfer. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled. If status write-back is enabled, the content is written to the control register location of the LLI in system memory at the end of the block transfer.

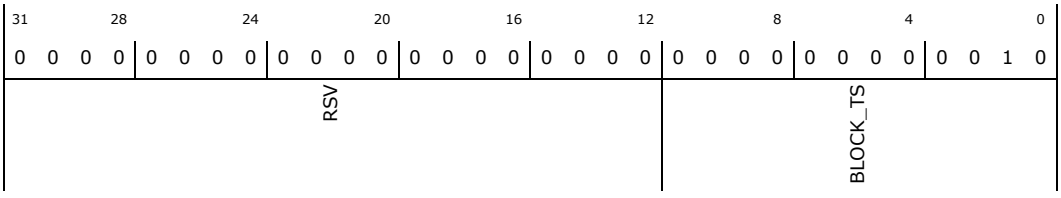
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 74h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000002h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11:0	002h RW	**Block length (BLOCK_TS):** When the DMAC is the flow controller, the user writes this field before the channel is enabled in order to indicate the block size. The number programmed into BLOCK_TS indicates the total number of single transactions to perform for every block transfer. A single transaction is mapped to a single AMBA beat. Width: The width of the single transaction is determined by CTL1_L.SRC_TR_WIDTH.

18.6.2.19 Channel 1 Source Status (SSTAT1)—Offset 78h

This register is a temporary placeholder for the source status information on its way to the SSTAT1 register location of the LLI. The source status information should be retrieved by software from the SSTAT1 register location of the LLI, and not by a read of this register over the DMAC slave interface.

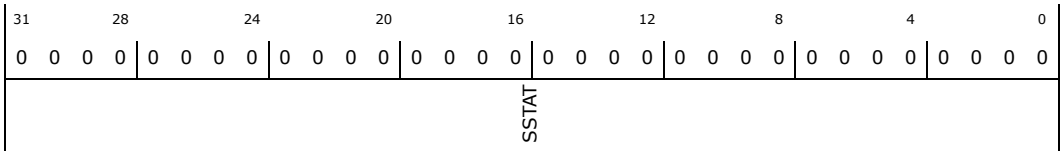
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 78h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Source Status (SSTAT):** Source status information retrieved by hardware from the address pointed to by the contents of the SSTATAR1 register.

18.6.2.20 Channel 1 Destination Status (DSTAT1)—Offset 80h

This register is a temporary placeholder for the destination status information on its way to the DSTAT1 register location of the LLI. The destination status information should be retrieved by software from the DSTAT1 register location of the LLI and not by a read of this register over the DMAC slave interface

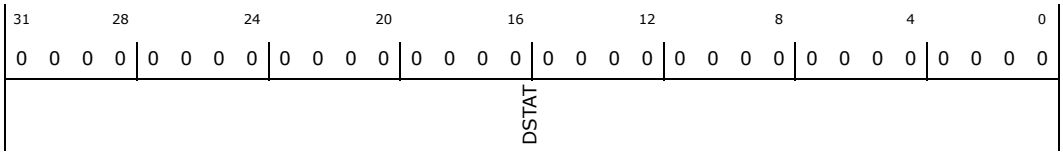
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 80h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Destination Status (DSTAT):** Destination status information retrieved by hardware from the address pointed to by the contents of the DSTATAR1 register

18.6.2.21 Channel 1 Source Status Address (SSTATAR1)—Offset 88h

After the completion of each block transfer, hardware can retrieve the source status information from the address pointed to by the contents of this register

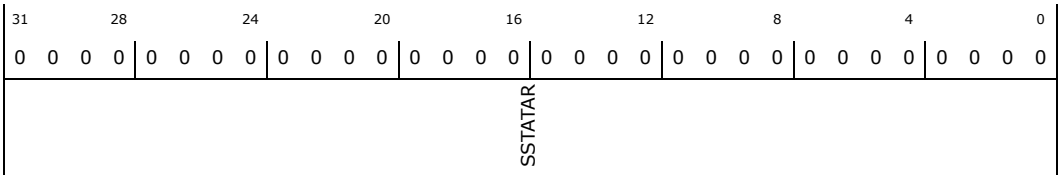
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 88h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Source Status Address (SSTATAR):** Pointer from where hardware can fetch the source status information, which is registered in the SSTAT1 register and written out to the SSTAT1 register location of the LLI before the start of the next block.

18.6.2.22 Channel 1 Destination Status Address (DSTATAR1)—Offset 90h

After the completion of each block transfer, hardware can retrieve the destination status information from the address pointed to by the contents of this register

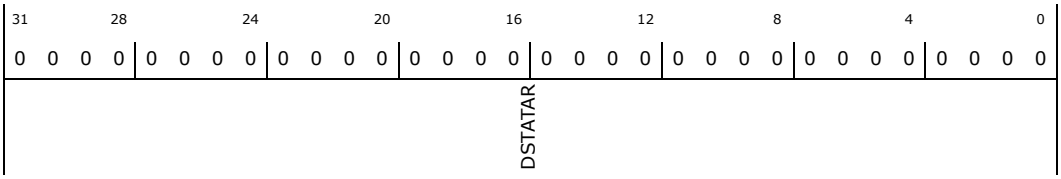
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 90h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0b RW	**Channel Destination Status Address (DSTATAR):** Pointer from where hardware can fetch the destination status information, which is registered in the DSTAT1 register and written out to the DSTAT1 register location of the LLI before the start of the next block.

18.6.2.23 Channel 1 Configuration LOWER (CFG1_L)—Offset 98h

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. This register need to be programmed before enable the channel.

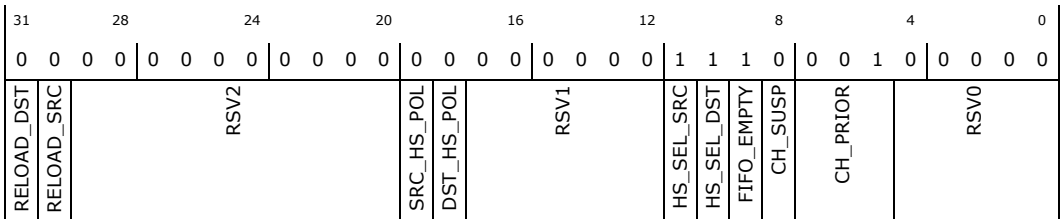
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 98h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000E20h



Bit Range	Default & Access	Field Name (ID): Description
31	0b RW	**Reload destination enable (RELOAD_DST):** The DARx register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
30	0b RW	**Reload source enable (RELOAD_SRC):** The SAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
29:20	0b RO	**Reserved (RSV2):** Reserved.
19	0b RW	**Source handshake polarity (SRC_HS_POL):** 0 = Active high 1 = Active low
18	0b RW	**Destination handshake polarity (DST_HS_POL):** 0 = Active high 1 = Active low
17:12	0b RW	**Reserved (RSV1):** Reserved.
11	1b RW	**Source Handshake select (HS_SEL_SRC):** Used to select which handshake interface is active for source requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If source peripheral is memory this bit is ignored
10	1b RW	**Destination Handshake select (HS_SEL_DST):** Used to select which handshake interface is active for destination requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If destination peripheral is memory this bit is ignored
9	1b RO	**Channel FIFO empty status (FIFO_EMPTY):** Indicates if there is data left in the channel FIFO. Can be used in conjunction with CH_SUSP to cleanly disable a channel. 1 = Channel FIFO empty 0 = Channel FIFO not empty
8	0b RW	**Channel Suspend control (CH_SUSP):** Suspends all DMA data transfers from the source until this bit is cleared. There is no guarantee that the current transaction will complete. Can also be used in conjunction with FIFO_EMPTY to cleanly disable a channel without losing any data. 0 = Not suspended. 1 = Suspend DMA transfer from the source.

Bit Range	Default & Access	Field Name (ID): Description
7:5	001b RW	**Channel Priority (CH_PRIOR):** Priority value equal to 7 is the highest priority, and 0 is the lowest. This field must be programmed within the following range: 0: 1 A programmed value outside this range will cause erroneous behavior.
4:0	0b RO	**Reserved (RSV0):** Reserved.

18.6.2.24 Channel 1 configuration UPPER (CFG1_U)—Offset 9Ch

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. This register need to be programmed before enable the channel.

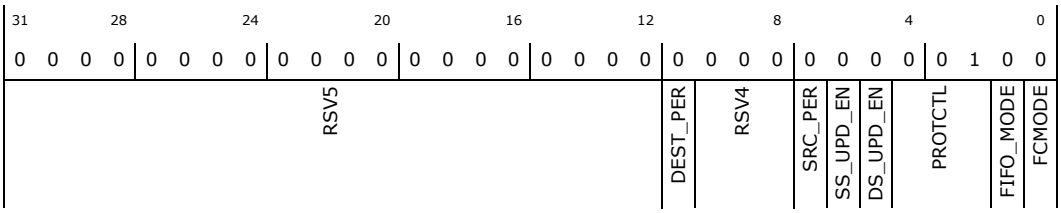
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 9Ch

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000004h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV5):** Reserved.
11	0b RW	**Destination hardware interface (DEST_PER):** Assigns a hardware handshaking interface (0-1) to the channel destination if the CFG1_L.HS_SEL_DST field is 0; otherwise, this field is ignored. The channel can then communicate with the destination peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
10:8	0b RO	**Reserved (RSV4):** Reserved.
7	0b RW	**Source hardware interface (SRC_PER):** Assigns a hardware handshaking interface (0-1) to the channel source if the CFG1_L.HS_SEL_SRC field is 0; otherwise, this field is ignored. The channel can then communicate with the source peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
6	0b RW	**Source Status Update Enable (SS_UPD_EN):** Source status information is fetched only from the location pointed to by the SSTATAR1 register, stored in the SSTAT1 register and written out to the SSTAT1 location of the LLI if this field is high
5	0b RW	**Destination Status Update Enable (DS_UPD_EN):** Destination status information is fetched only from the location pointed to by the DSTATAR1 register, stored in the DSTAT1 register and written out to the DSTAT0 location of the LLI if this field is high

Bit Range	Default & Access	Field Name (ID): Description
4:2	001b RW	**AHB bus protocol bus control (PROTCTL):** Protection Control bits used to drive the AHB HPROT[3:1] bus. The AMBA Specification recommends that the default value of HPROT indicates a non-cached, non-buffered, privileged data access. The reset value is used to indicate such an access. HPROT[0] is tied high because all transfers are data accesses, as there are no opcode fetches. There is a one-to-one mapping of these register bits to the HPROT[3:1] master interface signals.
1	0b RW	**Channel FIFO mode control (FIFO_MODE):** Determines how much space or data needs to be available in the FIFO before a burst transaction request is serviced. 0 = Space/data available for single AHB transfer of the specified transfer width. 1 = Data available is greater than or equal to half the FIFO depth for destination transfers and space available is greater than half the fifo depth for source transfers. The exceptions are at the end of a burst transaction request or at the end of a block transfer.
0	0b RW	**Channel flow control mode (FCMODE):** Determines when source transaction requests are serviced when the Destination Peripheral is the flow controller. 0 = Source transaction requests are serviced when they occur. Data pre-fetching is enabled. 1 = Source transaction requests are not serviced until a destination transaction request occurs. In this mode, the amount of data transferred from the source is limited so that it is guaranteed to be transferred to the destination prior to block termination by the destination. Data pre-fetching is disabled.

18.6.2.25 Channel 1 Source Gather (SGR1)—Offset A0h

The CTL1_L.SINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

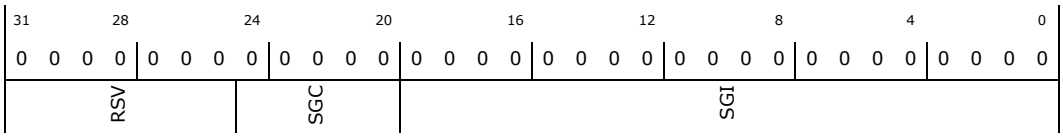
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + A0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0b RO	**Reserved (RSV):** Reserved.
24:20	0b RW	**Source Gather Count (SGC):** Source contiguous transfer count between successive gather boundaries. Specifies the number of contiguous source transfers of CTL1_L.SRC_TR_WIDTH between successive gather intervals. This is defined as a gather boundary
19:0	0b RW	**Source Gather Interval (SGI):** Specifies the source address increment/decrement in multiples of CTL1_L.SRC_TR_WIDTH on a gather boundary when gather mode is enabled for the source transfer.

18.6.2.26 Channel 1 Destination Scatter (DSR1)—Offset A8h

The CTL1_L.DINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

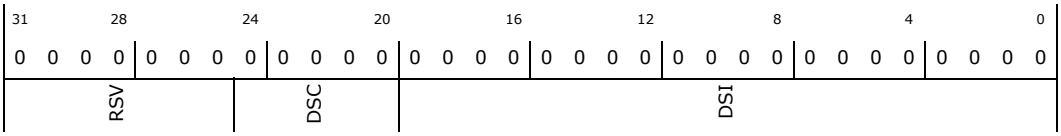
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + A8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:25	0b RO	**Reserved (RSV):** Reserved.
24:20	0b RW	**Destination Scatter Count (DSC):** Source contiguous transfer count between successive scatter boundaries. Specifies the number of contiguous destination transfers of CTL1_L.DST_TR_WIDTH between successive scatter intervals. This is defined as a scatter boundary
19:0	0b RW	**Destination Scatter Interval (DSI):** Specifies the destination address increment/ decrement in multiples of CTL1_L.DST_TR_WIDTH on a scatter boundary when scatter mode is enabled for the destination transfer.

18.6.2.27 Raw Status for IntTfr Interrupt (RAW_TFR)—Offset 2C0h

DMA Transfer Complete Interrupt. This interrupt is generated on DMA transfer completion to the destination peripheral

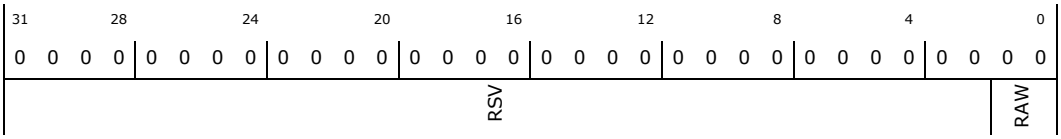
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2C0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Raw Status for IntTfr Interrupt (RAW):** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

18.6.2.28 Raw Status for IntBlock Interrupt (RAW_BLOCK)—Offset 2C8h

Block Transfer Complete Interrupt. This interrupt is generated on DMA block transfer completion to the destination peripheral.

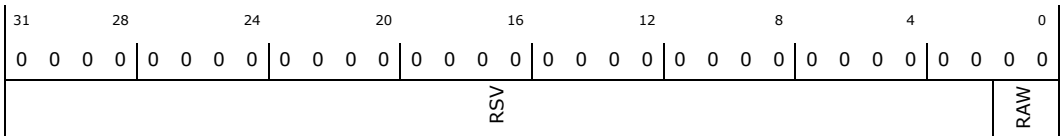
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2C8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Raw Status for IntBlock Interrupt (RAW):** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

18.6.2.29 Raw Status for IntSrcTran Interrupt (RAW_SRC_TRAN)—Offset 2D0h

Source Transaction Complete Interrupt. Generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the source side. NOTE: If the source is memory, then IntSrcTran interrupt should be ignored, as there is no concept of a DMA transaction level for memory

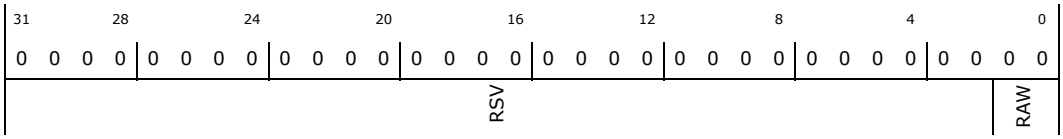
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2D0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Raw Status for IntSrcTran Interrupt (RAW):** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

18.6.2.30 Raw Status for IntDstTran Interrupt (RAW_DST_TRAN)—Offset 2D8h

Destination Transaction Complete Interrupt. Generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the destination side. NOTE: If the destination for a channel is memory, then that channel will never generate the IntDstTran interrupt. Because of this, the corresponding bit in this field will not be set.

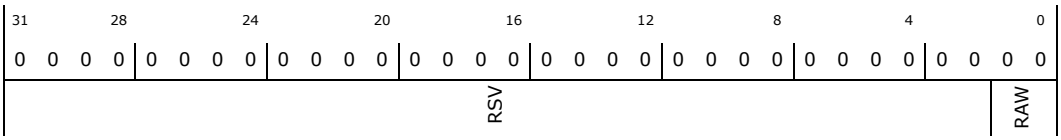
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2D8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Raw Status for IntDstTran Interrupt (RAW):** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

18.6.2.31 Raw Status for IntErr Interrupt (RAW_ERR)—Offset 2E0h

Error Interrupt. This interrupt is generated when an ERROR response is received from an AHB slave on the HRESP bus during a DMA transfer. In addition, the DMA transfer is cancelled and the channel is disabled.

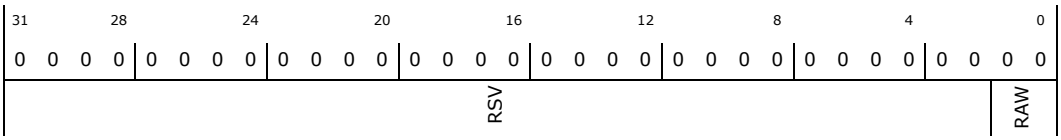
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2E0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Raw Status for IntErr Interrupt (RAW):** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

18.6.2.32 Status for IntTfr Interrupt (STATUS_TFR)—Offset 2E8h

DMA Transfer Complete Interrupt status

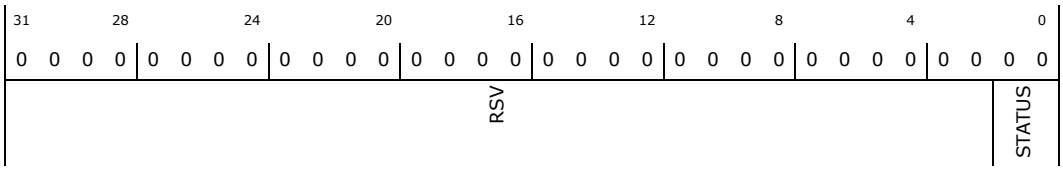
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2E8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RO	**Status for IntTfr Interrupt (STATUS):** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

18.6.2.33 Status for IntBlock Interrupt (STATUS_BLOCK)—Offset 2F0h

Block Transfer Complete Interrupt status

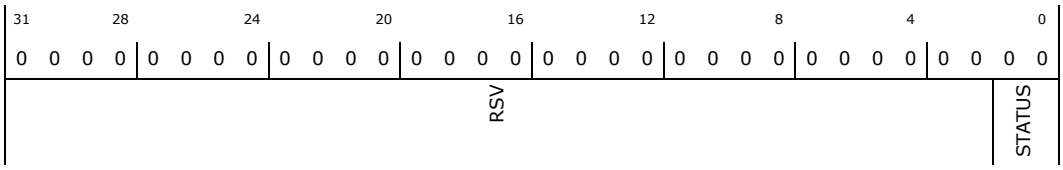
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2F0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RO	**Status for IntBlock Interrupt (STATUS):** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

18.6.2.34 Status for IntSrcTran Interrupt (STATUS_SRC_TRAN)—Offset 2F8h

Source Transaction Complete Interrupt status

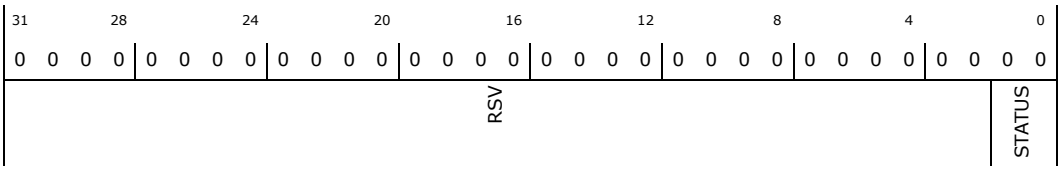
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 2F8h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RO	**Status for IntSrcTran Interrupt (STATUS):** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

18.6.2.35 Status for IntDstTran Interrupt (STATUS_DST_TRAN)—Offset 300h

Destination Transaction Complete Interrupt status

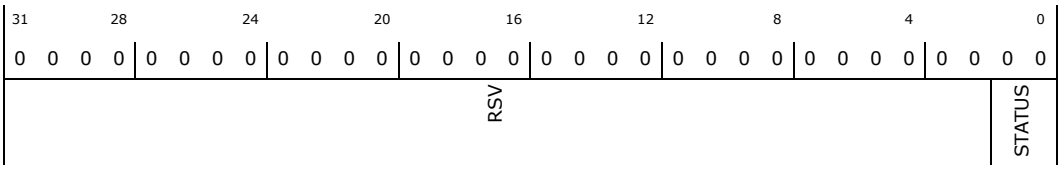
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 300h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RO	**Status for IntDstTran Interrupt (STATUS):** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

18.6.2.36 Status for IntErr Interrupt (STATUS_ERR)—Offset 308h

Error Interrupt status

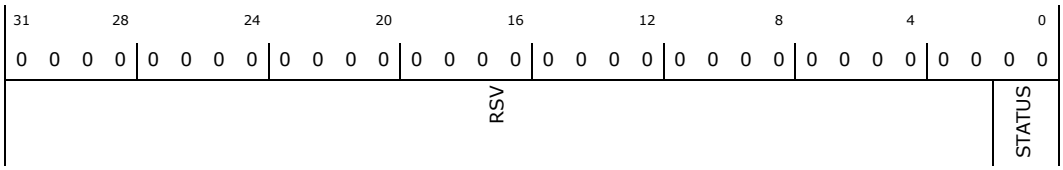
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 308h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RO	**Status for IntErr Interrupt (STATUS):** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

18.6.2.37 Mask for IntTfr Interrupt (MASK_TFR)—Offset 310h

DMA Transfer Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

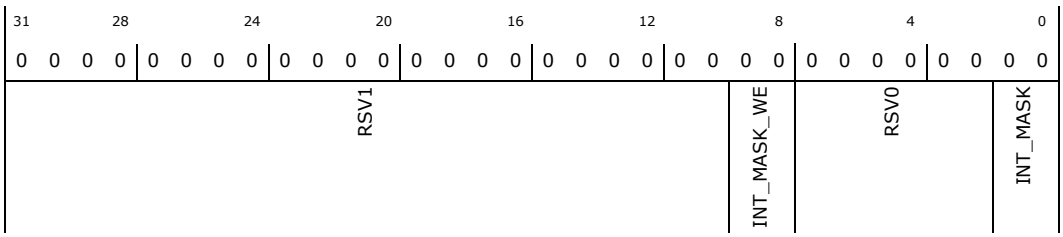
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 310h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Interrupt Mask Write Enable (INT_MASK_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Mask for the interrupt (INT_MASK):** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

18.6.2.38 Mask for IntBlock Interrupt (MASK_BLOCK)—Offset 318h

Block Transfer Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

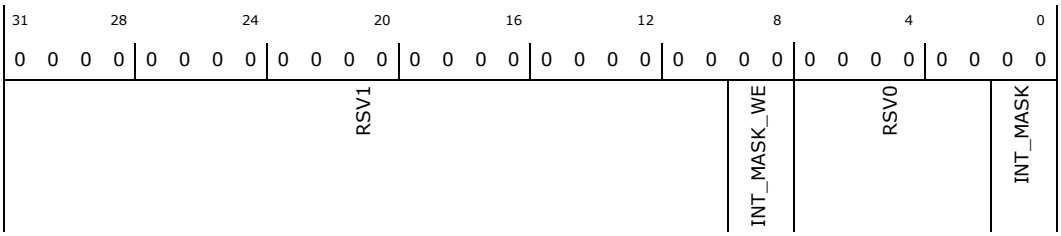
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 318h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Interrupt Mask Write Enable (INT_MASK_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Mask for the interrupt (INT_MASK):** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

18.6.2.39 Mask for IntSrcTran Interrupt (MASK_SRC_TRAN)—Offset 320h

Source Transaction Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

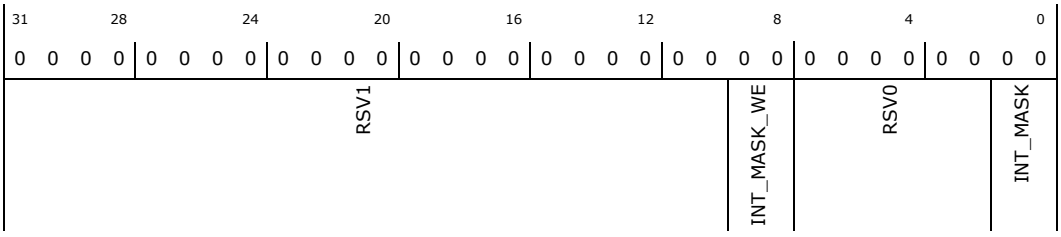
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 320h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
9:8	0b RW	**Interrupt Mask Write Enable (INT_MASK_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Mask for the interrupt (INT_MASK):** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

18.6.2.40 Mask for IntDstTran Interrupt (MASK_DST_TRAN)—Offset 328h

Destination Transaction Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

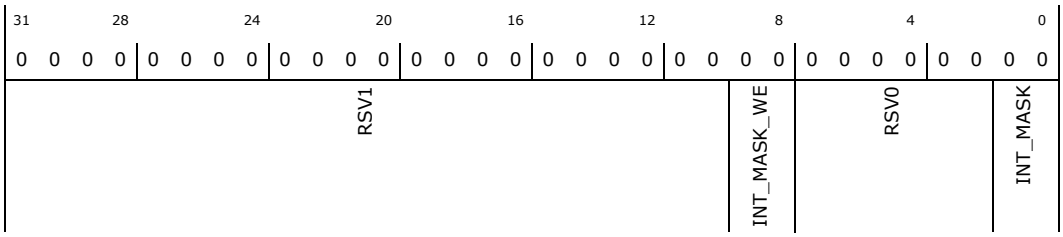
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 328h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Interrupt Mask Write Enable (INT_MASK_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Mask for the interrupt (INT_MASK):** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

18.6.2.41 Mask for IntErr Interrupt (MASK_ERR)—Offset 330h

Error Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

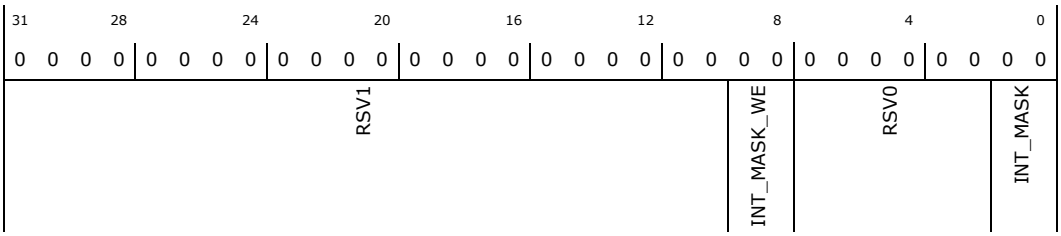
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 330h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Interrupt Mask Write Enable (INT_MASK_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Mask for the interrupt (INT_MASK):** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

18.6.2.42 Clear for IntTfr Interrupt (CLEAR_TFR)—Offset 338h

DMA Transfer Complete Interrupt clear

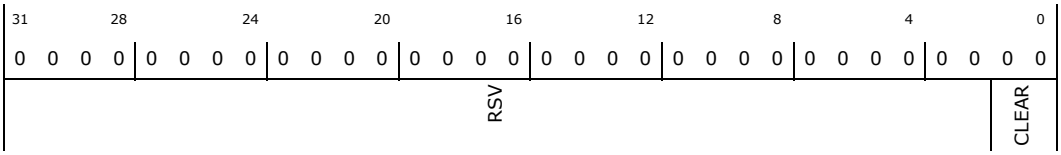
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 338h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b WO	**Clear for Interrupt (CLEAR):** 0 = no effect 1 = clear interrupt

18.6.2.43 Clear for IntBlock Interrupt (CLEAR_BLOCK)—Offset 340h

Block Transfer Complete Interrupt clear

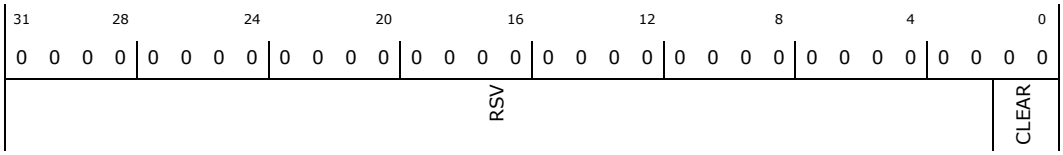
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 340h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b WO	**Clear for Interrupt (CLEAR):** 0 = no effect 1 = clear interrupt

18.6.2.44 Clear for IntSrcTran Interrupt (CLEAR_SRC_TRAN)—Offset 348h

Source Transaction Complete Interrupt clear

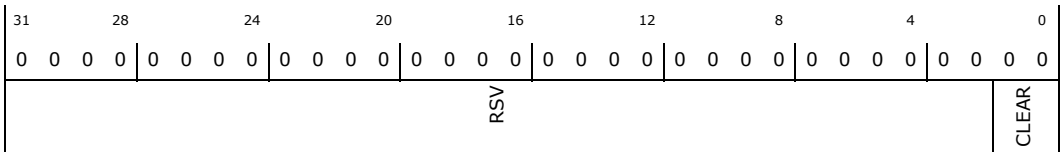
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 348h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b WO	**Clear for Interrupt (CLEAR):** 0 = no effect 1 = clear interrupt

18.6.2.45 Clear for IntDstTran Interrupt (CLEAR_DST_TRAN)—Offset 350h

Destination Transaction Complete Interrupt clear

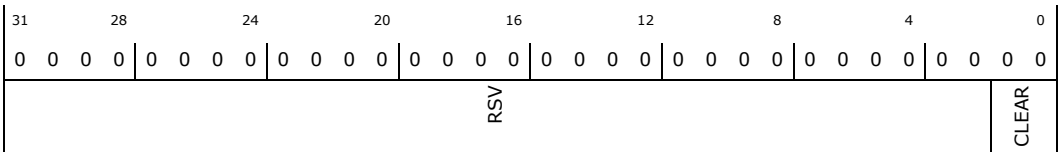
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 350h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b WO	**Clear for Interrupt (CLEAR):** 0 = no effect 1 = clear interrupt

18.6.2.46 Clear for IntErr Interrupt (CLEAR_ERR)—Offset 358h

Error Interrupt clear

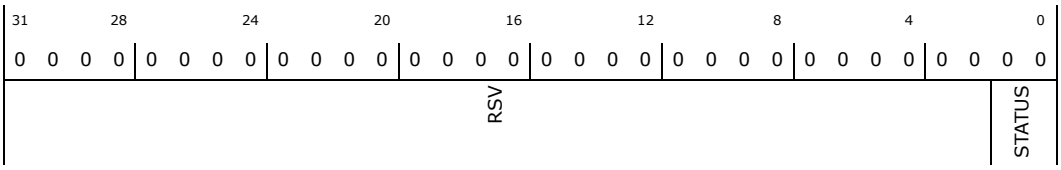
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 358h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b WO	**Clear for Interrupt (STATUS):** 0 = no effect 1 = clear interrupt

18.6.2.47 Combined Interrupt Status (STATUS_INT)—Offset 360h

The contents of each of the Status registers is ORed to produce a single bit for each interrupt type in this Combined Interrupt Status register.

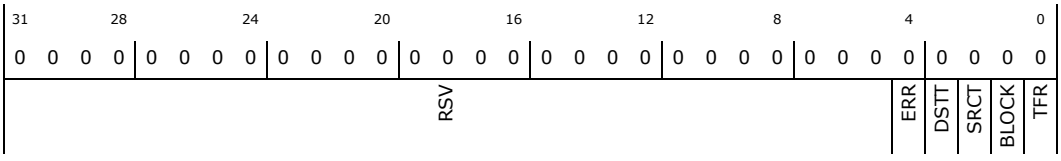
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 360h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0b RO	**Reserved (RSV):** Reserved.
4	0b RO	**OR of the contents of STATUS_ERR (ERR):** Reserved.
3	0b RO	**OR of the contents of STATUS_DSTT (DSTT):** Reserved.
2	0b RO	**OR of the contents of STATUS_SRCT (SRCT):** Reserved.
1	0b RO	**OR of the contents of STATUS_BLOCK (BLOCK):** Reserved.
0	0b RO	**OR of the contents of STATUS_TFR (TFR):** Reserved.

18.6.2.48 Source Software Transaction Request (REQ_SRC_REG)—Offset 368h

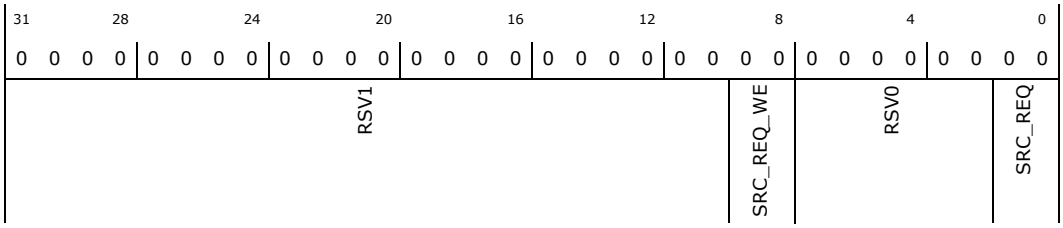
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 368h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Source Software Transaction Request write enable (SRC_REQ_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Source Software Transaction Request register (SRC_REQ):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.49 Destination Software Transaction Request register (REQ_DST_REG)—Offset 370h

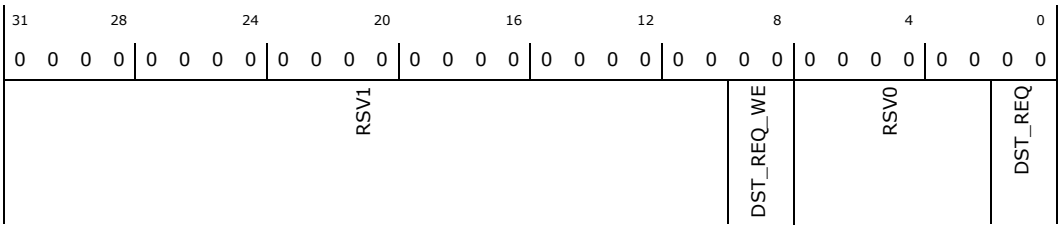
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 370h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Destination Software Transaction Request write enable (DST_REQ_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Destination Transaction Request register (DST_REQ):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.50 Source Single Transaction Request (SGL_REQ_SRC_REG)—Offset 378h

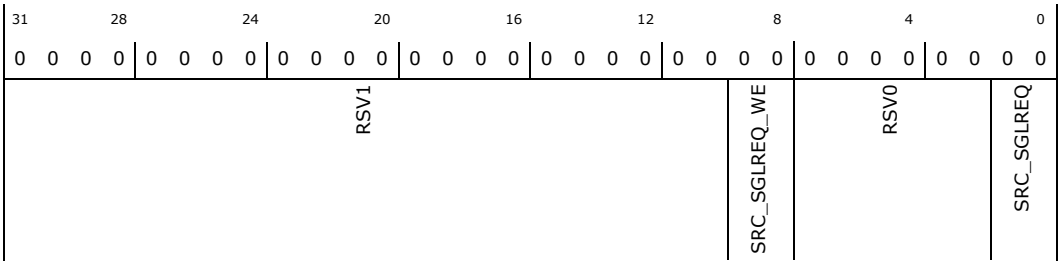
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 378h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Source Single Transaction Request write enable (SRC_SGLREQ_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Source Single Transaction Request register (SRC_SGLREQ):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.51 Destination Single Software Transaction Request (SGL_REQ_DST_REG)—Offset 380h

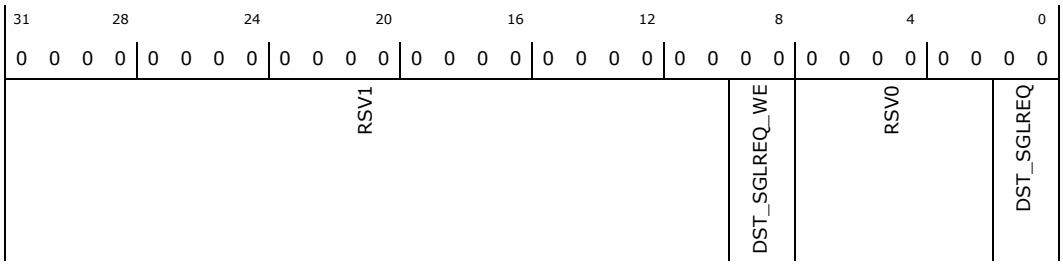
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 380h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Destination Single Transaction Request write enable (DST_SGLREQ_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Destination Single Transaction Request register (DST_SGLREQ):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.52 Source Last Transaction Request (LST_SRC_REG)—Offset 388h

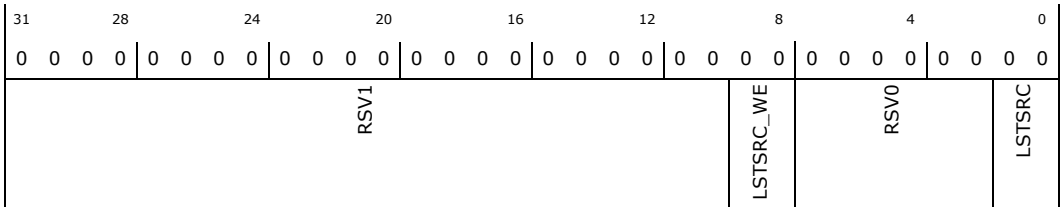
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 388h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
9:8	0b RW	**Source Last Transaction Request write enable (LSTSRC_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Source Last Transaction Request register (LSTSRC):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.53 Destination Single Transaction Request (LST_DST_REG)—Offset 390h

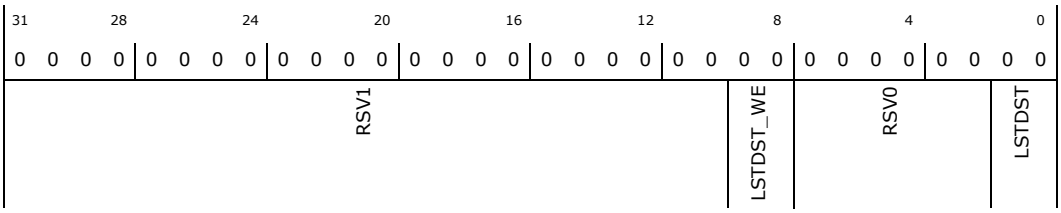
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 390h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Destination Last Transaction Request write enable (LSTDST_WE):** 0 = write disabled 1 = write enabled
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Destination Last Transaction Request register (LSTDST):** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

18.6.2.54 DMA Configuration (DMA_CFG_REG)—Offset 398h

Used to enable the DMA controller (DMAC), which must be done before any channel activity can begin. If the global channel enable bit is cleared while any channel is still active, then DMA_EN still returns 1 to indicate that there are channels still active until hardware has terminated all activity on all channels, at which point the DMA_EN bit returns 0

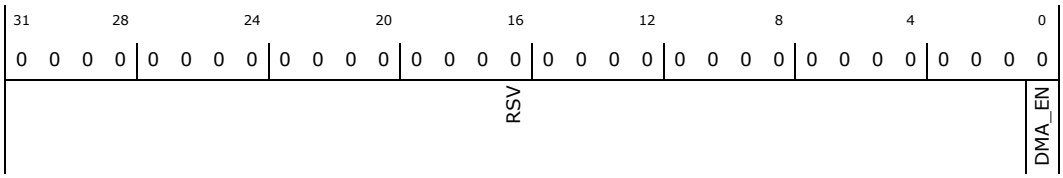
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 398h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RW	**DMA global enable (DMA_EN):** 0 = disabled 1 = enabled

18.6.2.55 Channel Enable (CH_EN_REG)—Offset 3A0h

Software can read this register in order to find out which channels are currently inactive if needs to set up a new channel. It can then enable an inactive channel with the required priority.

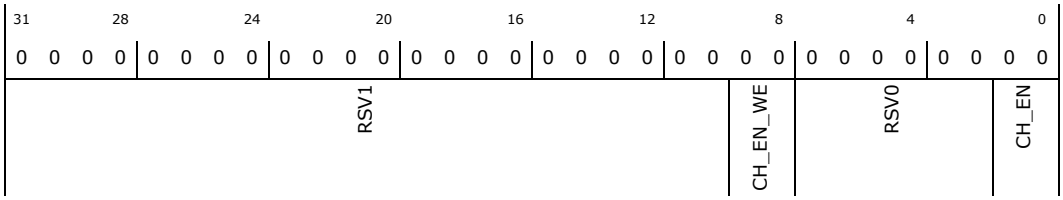
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 3A0h

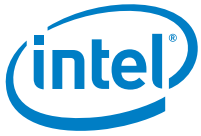
BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:20, F:1] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0b RO	**Reserved (RSV1):** Reserved.
9:8	0b RW	**Channel enable register (CH_EN_WE):** Reserved.
7:2	0b RO	**Reserved (RSV0):** Reserved.
1:0	0b RW	**Channel enable register (CH_EN):** Setting this bit enables a channel. Clearing this bit disables the channel. 0 = Disable the Channel 1 = Enable the Channel The CH_EN_REG.CH_EN bit is automatically cleared by hardware to disable the channel after the last AMBA transfer of the DMA transfer to the destination has completed. Software can therefore poll this bit to determine when this channel is free for a new DMA transfer.

§ §



19.0 I²C* Controller/GPIO Controller

The Intel® Quark SoC X1000 has an I²C* controller connected to the I/O Fabric. Both 7-bit and 10-bit addressing modes are supported. This controller operates in master mode only.

In addition, this PCI function also provides a GPIO controller. The SoC provides a total of 16 GPIOs that are split between the Legacy Bridge (D:31 F:0) and the GPIO controller (D:21 F:2). This chapter details the non-Legacy GPIOs provided by the GPIO controller.

19.1 I²C Controller

19.1.1 Signal Descriptions

I²C is a two-wire bus for inter-IC communication. Data and clock signals carry information between the connected devices.

Please see Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 102. I²C* Signals

Signal Name	Direction/ Type	Description
I2C_DATA	I/O	I²C Serial Data
I2C_CLK	I/O	I²C Serial Clock

19.1.2 Features

19.1.2.1 I²C* Protocol

The I²C bus is a two-wire serial interface, consisting of a serial data line and a serial clock. These wires carry information between the devices connected to the bus. Each device is recognized by a unique address and can operate as either a "transmitter" or "receiver," depending on the function of the device. Devices are considered slaves when performing data transfers, as the SoC is always a Master. A master is a device that initiates a data transfer on the bus and generates the clock signals to permit that transfer. At that time, any device addressed is considered a slave.

- The SoC is always the I²C master; and it supports multi-master mode.

- The SoC can support clock stretching by slave devices.
- The I2C_DATA line is a bidirectional signal and changes only while the I2C_CLK line is low, except for STOP, START, and RESTART conditions.
- The output drivers are open-drain or open-collector to perform wire-AND functions on the bus.
- The maximum number of devices on the bus is limited by the maximum capacitance specification:
  — 130 pF for standard and fast mode
  Refer to Chapter 4.0, "Electrical Characteristics" for details.
- Data is transmitted in byte packages.

19.1.2.2 I2C* Modes of Operation

The I2C module can operate in the following modes:

- Standard mode (with a bit rate up to 100 Kb/s)
- Fast mode (with a bit rate up to 400 Kb/s)

The I2C module can communicate with devices using only these modes as long as they are attached to the bus. Additionally, fast mode devices are downward compatible.

- Fast mode devices can communicate with standard mode devices in a 0–100 Kb/s I2C bus system.

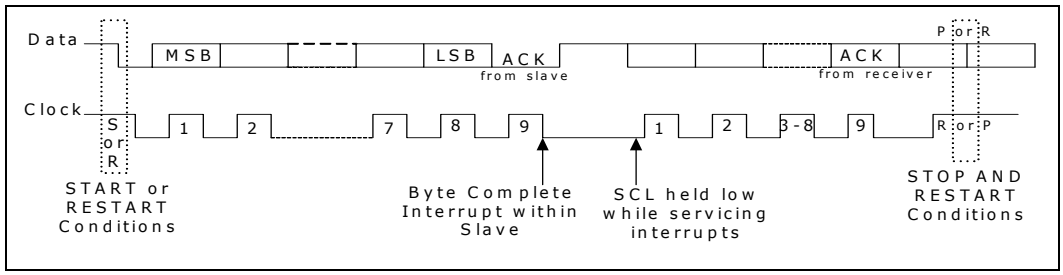
However, according to the I2C specification, standard mode devices are not upward compatible and should not be incorporated in a fast-mode I2C bus system since they cannot follow the higher transfer rate and unpredictable states would occur.

19.1.2.3 Functional Description

- The I2C master is responsible for generating the clock and controlling the transfer of data.
- The slave is responsible for either transmitting or receiving data to/from the master.
- The acknowledgement of data is sent by the device that is receiving data, which can be either a master or a slave.
- Each slave has a unique address that is determined by the system designer:
  — When a master wants to communicate with a slave, the master transmits a START/RESTART condition that is then followed by the slave's address and a control bit (R/W), to determine if the master wants to transmit data or receive data from the slave.
  — The slave then sends an acknowledge (ACK) pulse after the address.
- If the master (master-transmitter) is writing to the slave (slave-receiver):
  — The receiver gets one byte of data.
  — This transaction continues until the master terminates the transmission with a STOP condition.
- If the master is reading from a slave (master-receiver):
  — The slave transmits (slave-transmitter) a byte of data to the master, and the master then acknowledges the transaction with the ACK pulse.
  — This transaction continues until the master terminates the transmission by not acknowledging (NACK) the transaction after the last byte is received, and then

the master issues a STOP condition or addresses another slave after issuing a RESTART condition. This behavior is illustrated in Figure 35.

Figure 35. Data Transfer on the I²C* Bus



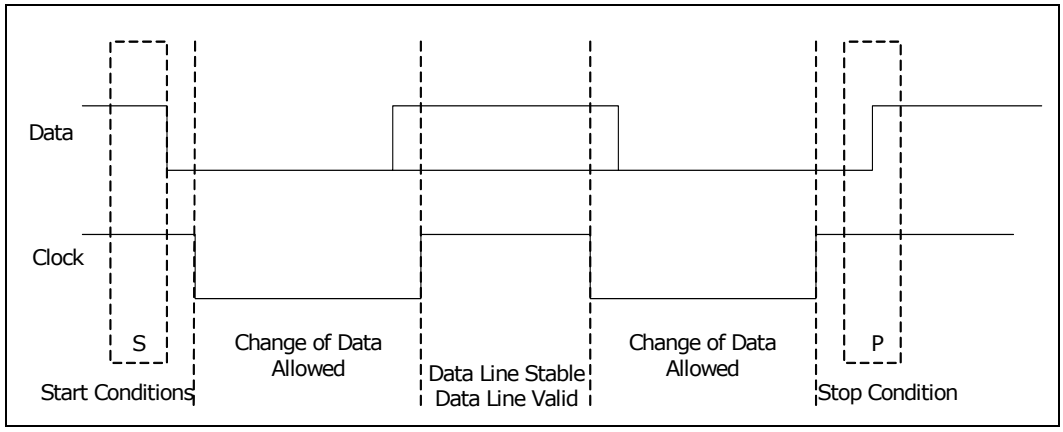
19.1.2.3.1 START and STOP Conditions

When the bus is idle, both the clock and data signals are pulled high through external pull-up resistors on the bus.

When the master wants to start a transmission on the bus, the master issues a START condition.

- This is defined to be a high-to-low transition of the data signal while clock is a '1'.
- When the master wants to terminate the transmission, the master issues a STOP condition. This is defined to be a low-to-high transition of the data line while clock is a 1. Figure 36 shows the timing of the START and STOP conditions.
- When data is being transmitted on the bus, the data line must be stable when clock is a 1.

Figure 36. START and STOP Conditions



The signal transitions for the START/STOP conditions, as depicted above, reflect those observed at the output of the Master driving the I²C bus. Care should be taken when observing the data/clock signals at the input of the Slave(s), because unequal line delays may result in an incorrect data/clock timing relationship.

19.1.2.3.2 Addressing Slave Protocol

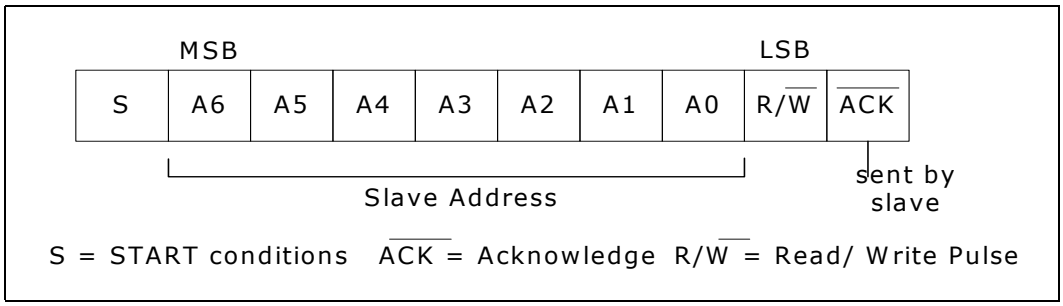
There are two address formats—7-bit address format and 10-bit address format.

7-bit Address Format

- During the seven-bit address format, the first seven bits (bits 7:1) of the first byte set the slave address and the LSB bit (bit 0) is the R/W bit as shown in Figure 37.

- When bit 0 (R/W) is set to 0, the master writes to the slave. When bit 0 (R/W) is set to 1, the master reads from the slave.

Figure 37. 7-Bit Address Format



10-bit Address Format

- During 10-bit addressing, 2 bytes are transferred to set the 10-bit address. The transfer of the first byte contains the following bit definition.
  - The first five bits (bits 7:3) notify the slaves that this is a 10-bit transfer. The next two bits (bits 2:1) set the slave's address bits 9:8. The LSB bit (bit 0) is the RW bit.
  - The second byte transferred sets bits 7:0 of the slave address.
  - Figure 38 shows the 10-bit address format, and Table 103 defines the special purpose and reserved first byte addresses.

Figure 38. 10-bit Address Format

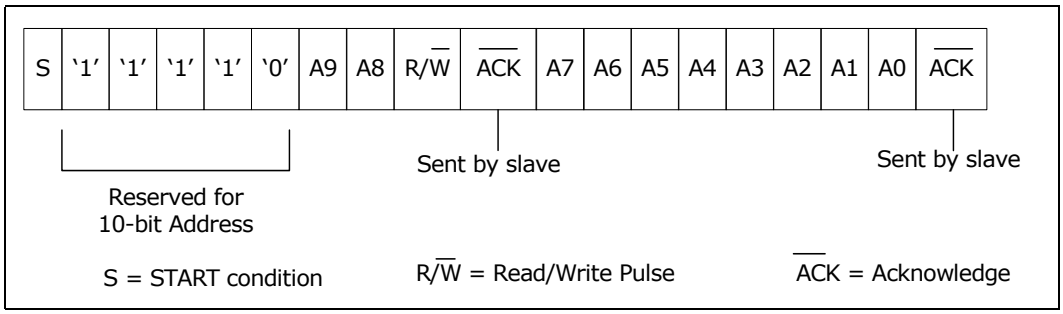


Table 103. I²C* Definition of Bits in First Byte

Slave Address	RW Bit	Description
0000 000	0	**General Call Address**—The I²C controller places the data in the receive buffer and issues a General Call interrupt.
0000 000	1	**START byte**—For more details, refer to I²C bus specification section 3.15.
0000 001	X	**CBUS address**—I²C controller ignores these accesses.
0000 010	X	**Reserved**
0000 011	X	**Reserved**
0000 1XX	X	**High-speed master code**
1111 1XX	X	**Reserved**
1111 0XX	X	**Ten (10)-bit slave addressing**

19.1.2.3.3 Transmit and Receive Protocol

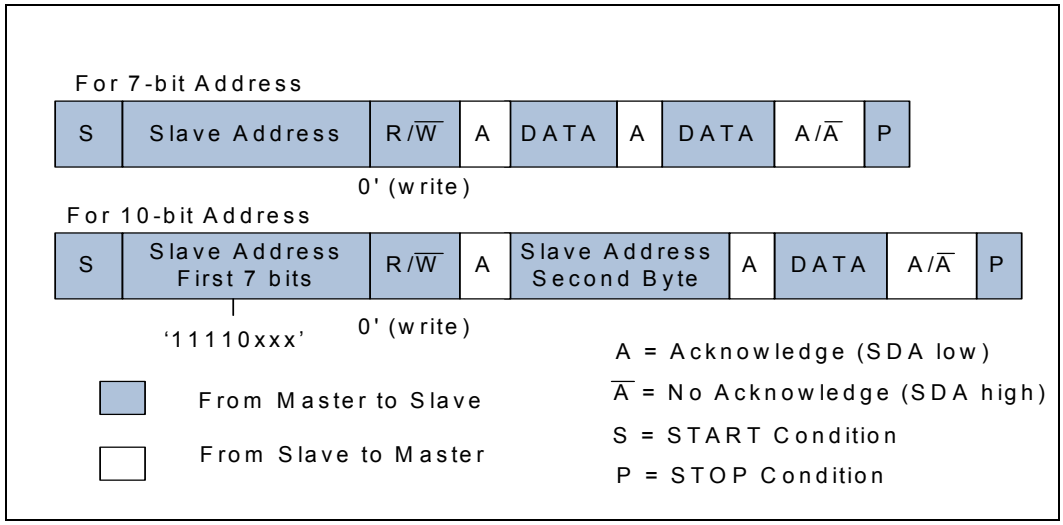
The master can initiate data transmission and reception to/from the bus, acting as either a master-transmitter or master-receiver. A slave responds to requests from the master by either transmitting data or receiving data to/from the bus, acting as either a slave-transmitter or slave-receiver, respectively.

Master-Transmitter and Slave-Receiver

All data is transmitted in byte format, with no limit on the number of bytes transferred per data transfer. After the master sends the address and RW bit or the master transmits a byte of data to the slave, the slave-receiver must respond with the acknowledge signal (ACK). When a slave-receiver does not respond with an ACK pulse, the master aborts the transfer by issuing a STOP condition. The slave must leave the SDA line high so that the master can abort the transfer.

If the master-transmitter is transmitting data as shown in Figure 39, then the slave-receiver responds to the master-transmitter with an acknowledge pulse after every byte of data is received.

Figure 39. Master Transmitter Protocol

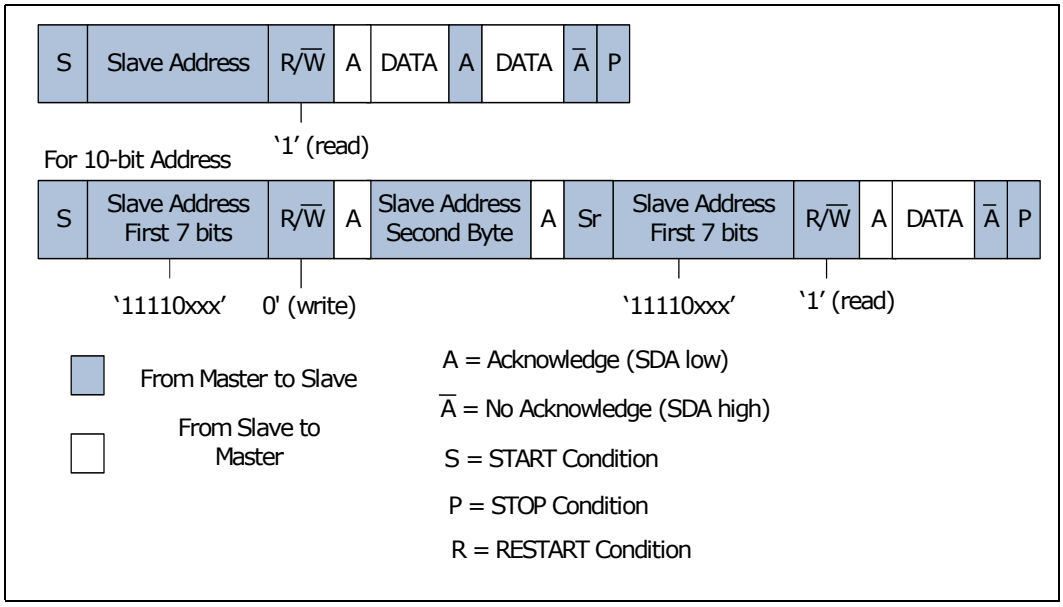


Master-Receiver and Slave-Transmitter

If the master is receiving data as shown in Figure 40, the master responds to the Slave-Transmitter with an acknowledge pulse after a byte of data has been received, except for the last byte. This is the way the Master-Receiver notifies the Slave-Transmitter that this is the last byte. The Slave-Transmitter relinquishes the SDA line after detecting the No Acknowledge (NACK) so that the master can issue a STOP condition.

When a master does not want to relinquish the bus with a STOP condition, the master can issue a RESTART condition. This is identical to a START condition except it occurs after the ACK pulse. The master can then communicate with the same slave or a different slave.

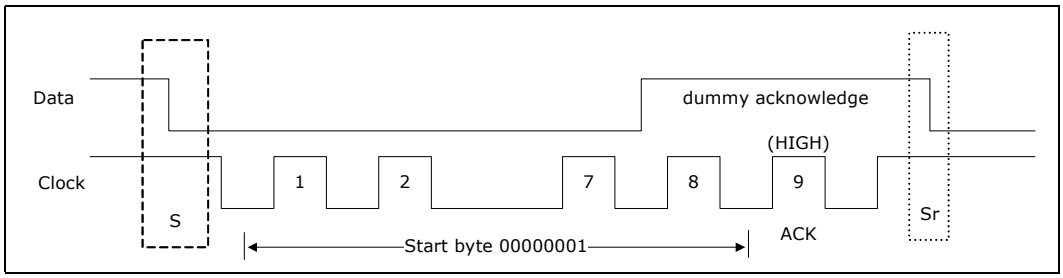
Figure 40. Master Receiver Protocol



19.1.2.3.4 START BYTE Transfer Protocol

The START BYTE Transfer protocol is set up for systems that do not have an on-board dedicated I²C hardware module. When the I²C controller is a master, it supports the generation of START BYTE transfers at the beginning of every transfer in case a slave device requires it. This protocol consists of 7 '0's being transmitted followed by a 1, as illustrated in Figure 41. This allows the processor that is polling the bus to under-sample the address phase until 0s are detected. Once the microcontroller detects a 0, it switches from the under sampling rate to the correct rate of the master.

Figure 41. START Byte Transfer



The START BYTE procedure is as follows:

1. Master generates a START condition.
2. Master transmits the START byte (0000 0001).
3. Master transmits the ACK clock pulse. (Present only to conform with the byte handling format used on the bus.)
4. No slave sets the ACK signal to 0.
5. Master generates a RESTART (R) condition.

A hardware receiver does not respond to the START BYTE because it is a reserved address and resets after the RESTART condition is generated.

19.1.3 Use

19.1.3.1 Master Mode Operation

To use the I²C controller as a master, perform the following steps:

1. Disable the I²C controller by writing 0 (zero) to IC_ENABLE.ENABLE.
2. Write to the IC_CON register to set the maximum speed mode supported (IC_CON.SPEED) and to specify whether the I²C controller starts its transfers in 7/10 bit addressing mode.
3. Write to the IC_TAR register the address of the I²C device to be addressed. The desired speed of the I²C controller master-initiated transfers, either 7-bit or 10-bit addressing, is controlled by the IC_TAR.IC_10BITADDR_MASTER bit field.
4. Enable the I²C controller by writing a 1 in IC_ENABLE.
5. Write the transfer direction and data to be sent to the IC_DATA_CMD register. If the IC_DATA_CMD register is written before the I²C controller is enabled, the data and commands are lost as the buffers are kept cleared when I²C controller is not enabled.

RESTART and STOP conditions are generated only under software control through specific IC_DATA_CMD fields. When the Transmit FIFO is empty the I²C controller does not automatically generate a STOP condition and pauses the I²C bus by holding SCL low. As part of the command definition (IC_DATA_CMD) software must specify if a particular command is going to be followed by a STOP or RESTART condition by setting either IC_DATA_CMD.STOP or IC_DATA_CMD.RESTART.

The I²C controller supports switching back and forth between reading and writing dynamically. To transmit data, write the data to be written to the lower byte of the I²C Rx/Tx Data Buffer and Command Register (IC_DATA_CMD). The IC_DATA_CMD.CMD should be written to 0 for I²C write operations. Subsequently, a read command may be issued by writing "don't cares" to IC_DATA_CMD.DAT register bits, and a 1 should be written to the IC_DATA_CMD.CMD bit.

19.1.3.2 Disabling I²C* Controller

The register IC_ENABLE is added to allow software to unambiguously determine when the hardware has completely shutdown in response to the IC_ENABLE.ENABLE register being set from 1 to 0.

Procedure

1. Define a timer interval (t_{i2c_poll}) equal to 10 times the signaling period for the highest I²C transfer speed used in the system and supported by the I²C controller. For example, if the highest I²C transfer mode is 400Kb/s, then this t_{i2c_poll} is 25 µs.
2. Define a maximum time-out parameter, MAX_T_POLL_COUNT, such that if any repeated polling operation exceeds this maximum value, an error is reported.
3. Execute a blocking thread/process/function that prevents any further I²C master transactions to be started by software, but allows any pending transfers to be completed.
4. The variable POLL_COUNT is initialized to zero (0).
5. Set IC_ENABLE.ENABLE to zero (0).
6. Read the IC_ENABLE_STATUS.IC_EN bit. Increment POLL_COUNT by one. If POLL_COUNT >= MAX_T_POLL_COUNT, exit with the relevant error code.
7. If IC_ENABLE_STATUS.IC_EN is 1, then sleep for t_{i2c_poll} and proceed to the previous step. Otherwise, exit with a relevant success code.

19.1.4 References

UM10204 I2C-Bus Specification and User Manual, Revision 03

19.2 GPIO Controller

19.2.1 Signal Descriptions

Please see Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 104. GPIO Signals

Signal Name	Direction/ Type	Description
GPIO[7:0]	I/O TBD	General Purpose IO available in the S0 state

19.2.2 Features

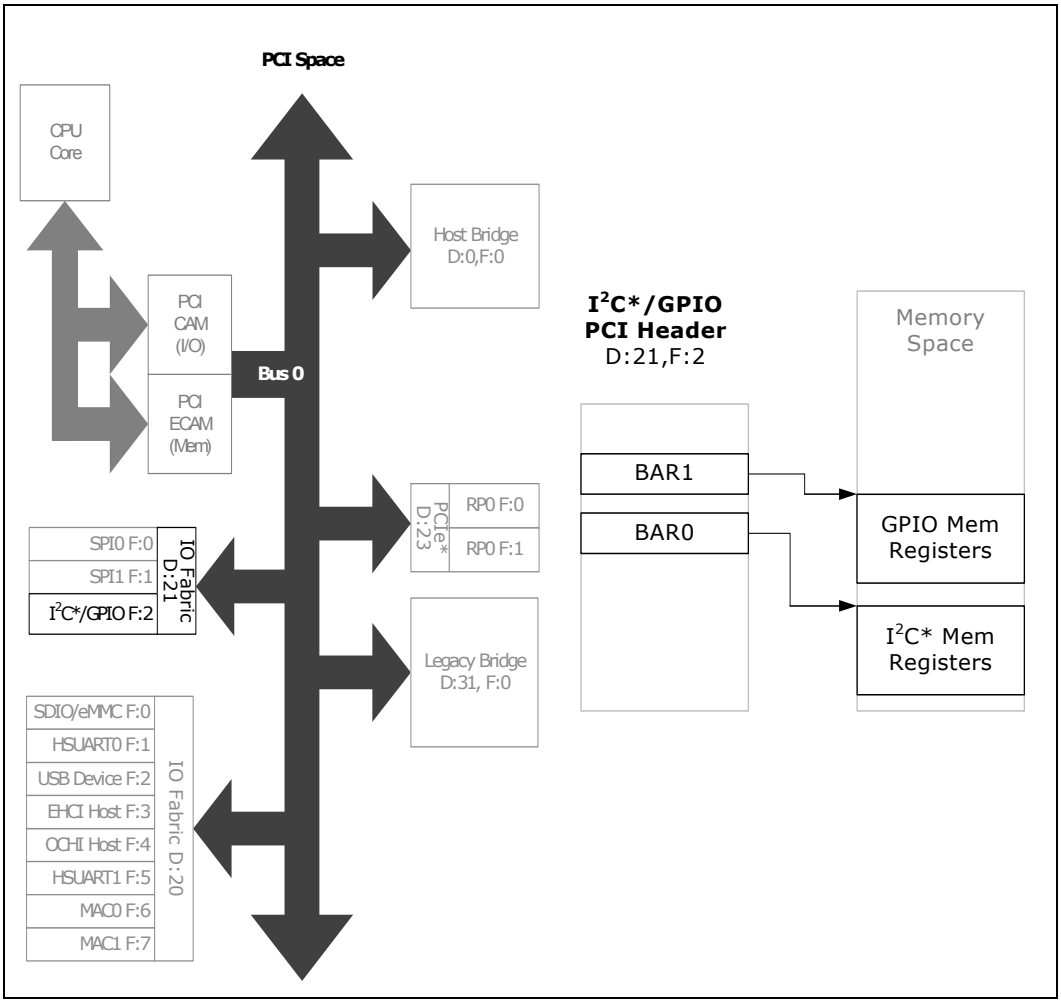
The GPIO Controller provides 8 GPIO pins via Port A of the controller.

- 8 Independently configurable GPIOs
- Separate data register and data direction for each GPIO
- Interrupt source mode supported for each GPIO
- Debounce logic for interrupt sources
- Metastability registers for GPIO read data

19.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 42. I2C*/GPIO Register Map



19.4 PCI Configuration Registers

Table 105. Summary of PCI Configuration Registers—0/21/2

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 726	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 727	0934h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 727	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 728	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 728	0C800010h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 729	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 729	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 730	80h

Table 105. Summary of PCI Configuration Registers—0/21/2 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
Fh	Fh	"BIST (BIST)—Offset Fh" on page 730	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 731	00000000h
14h	17h	"Base Address Register (BAR1)—Offset 14h" on page 731	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 732	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 732	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 733	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 733	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 733	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 734	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 734	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 735	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 735	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 735	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 736	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 736	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 737	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 738	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 738	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 738	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 739	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 739	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 740	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 740	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 740	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 741	00000000h

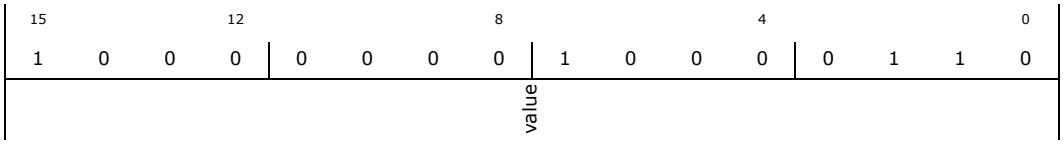
19.4.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:21, F:2] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

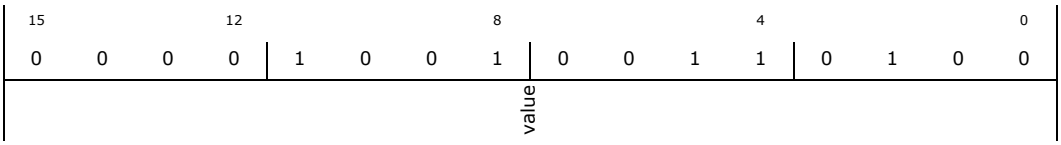
19.4.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:21, F:2] + 2h

Default: 0934h



Bit Range	Default & Access	Description
15: 0	0934h RO	**Device ID (value):** PCI Device ID

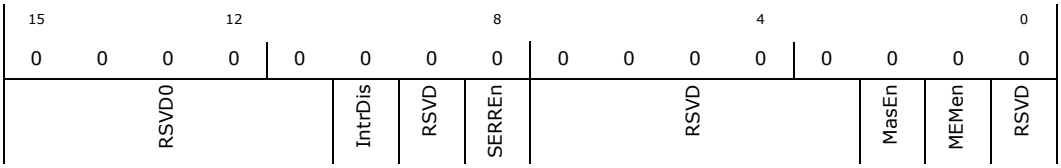
19.4.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:21, F:2] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

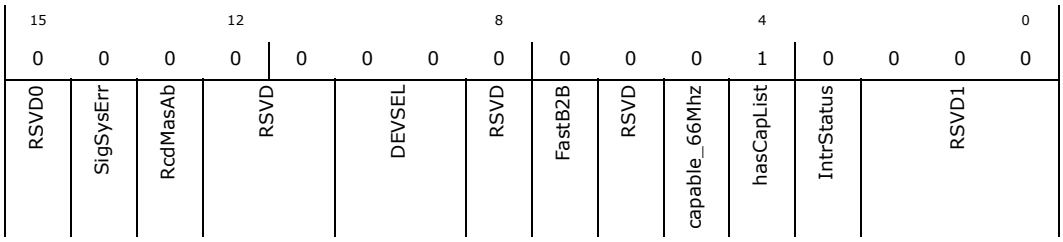
19.4.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:21, F:2] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

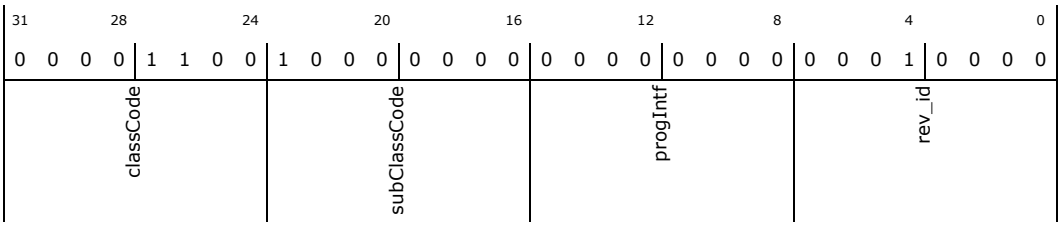
19.4.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:21, F:2] + 8h

Default: 0C800010h



Bit Range	Default & Access	Description
31: 24	0Ch RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	80h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	00h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

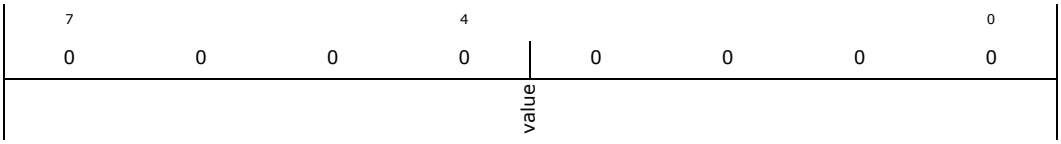
19.4.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:21, F:2] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

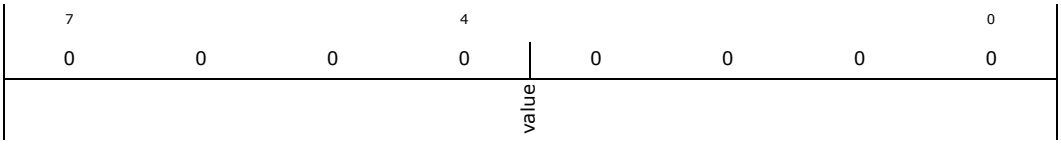
19.4.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:21, F:2] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

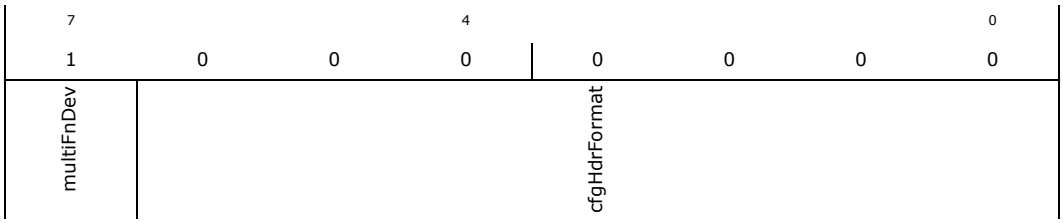
19.4.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:21, F:2] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

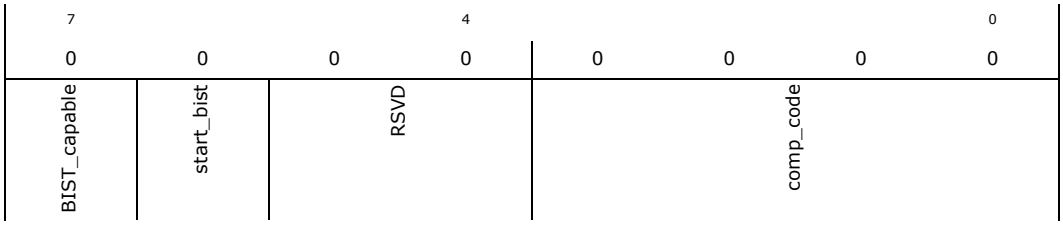
19.4.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:21, F:2] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

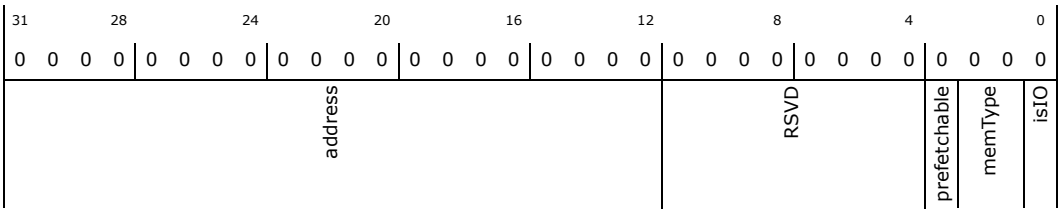
19.4.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

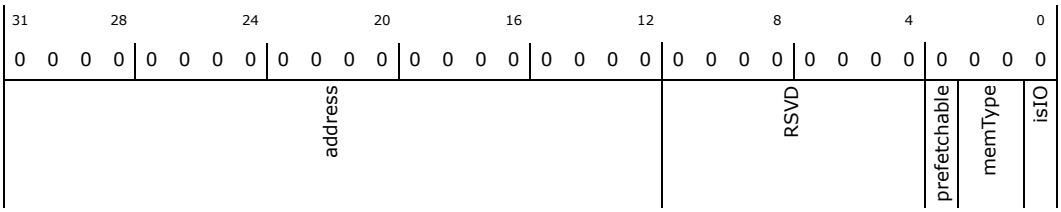
19.4.11 Base Address Register (BAR1)—Offset 14h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR1: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

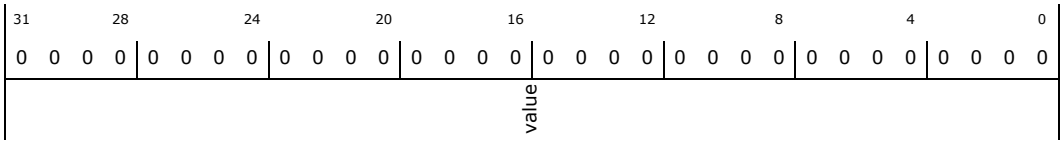
19.4.12 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:21, F:2] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

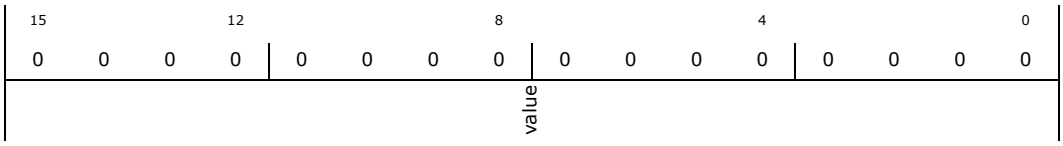
19.4.13 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:21, F:2] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

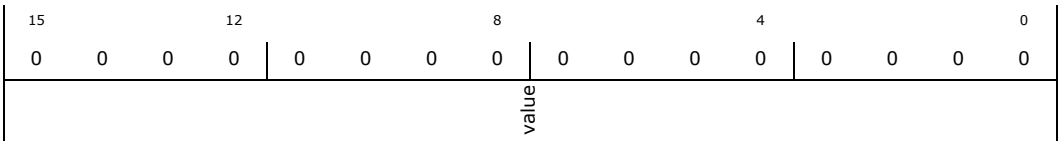
19.4.14 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:21, F:2] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

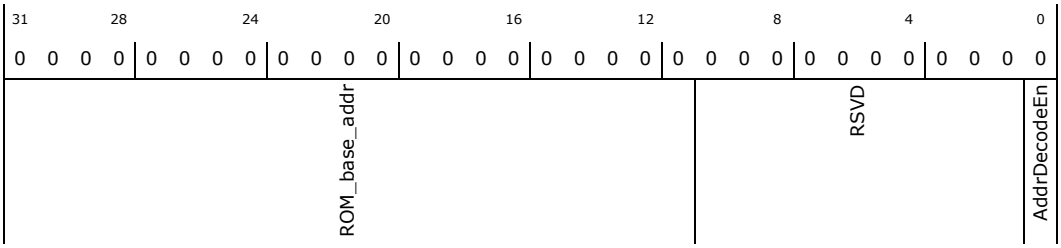
19.4.15 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:21, F:2] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

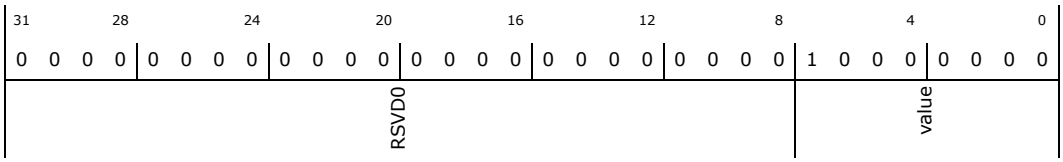
19.4.16 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:21, F:2] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

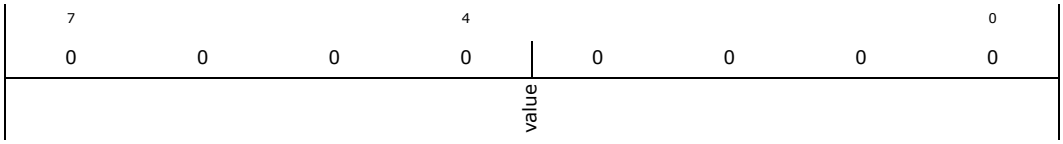
19.4.17 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:21, F:2] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

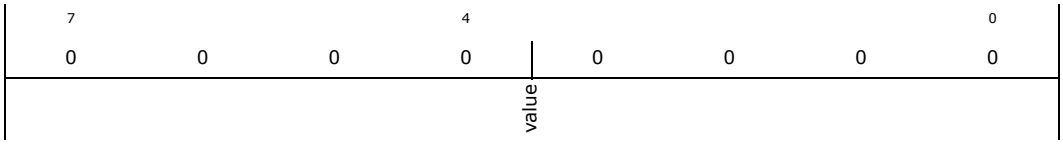
19.4.18 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:21, F:2] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

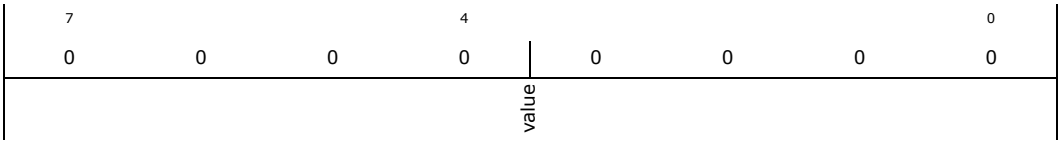
19.4.19 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:21, F:2] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

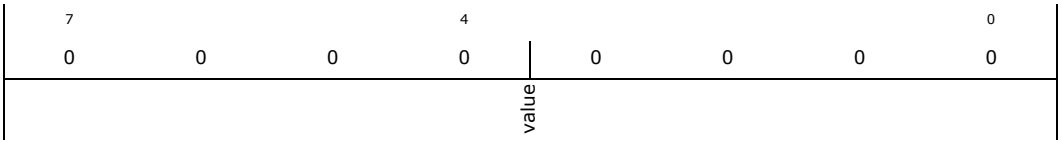
19.4.20 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:21, F:2] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

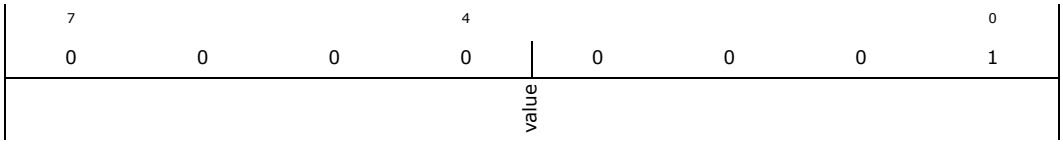
19.4.21 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:21, F:2] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

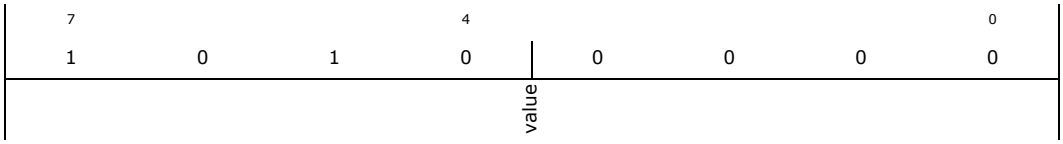
19.4.22 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:21, F:2] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

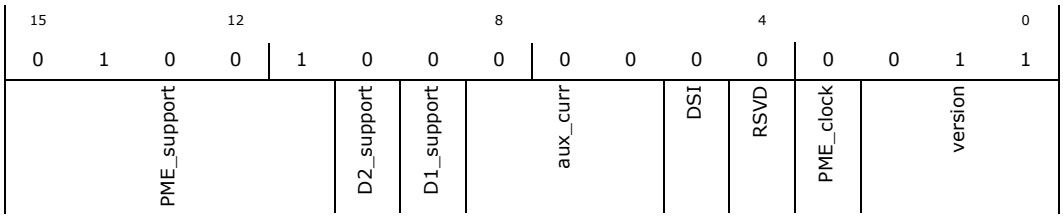
19.4.23 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:21, F:2] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported

Bit Range	Default & Access	Description
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

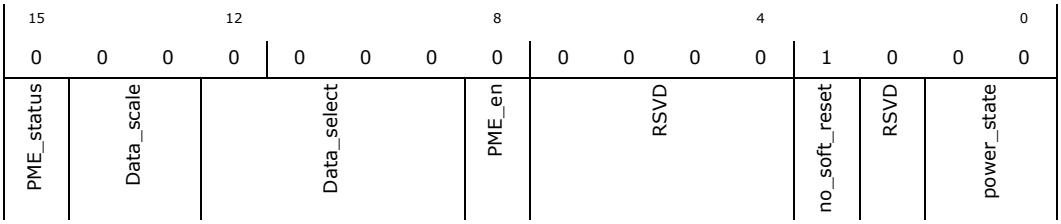
19.4.24 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:21, F:2] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

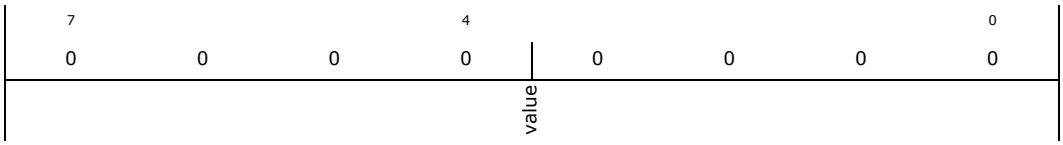
19.4.25 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:21, F:2] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

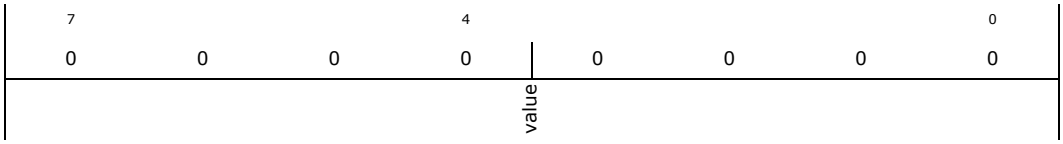
19.4.26 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:21, F:2] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

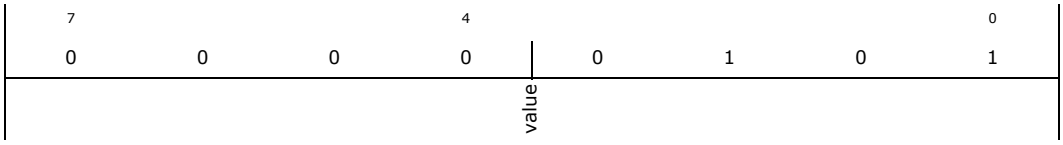
19.4.27 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:21, F:2] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

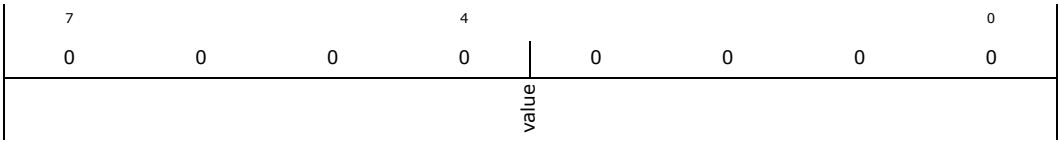
19.4.28 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:21, F:2] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

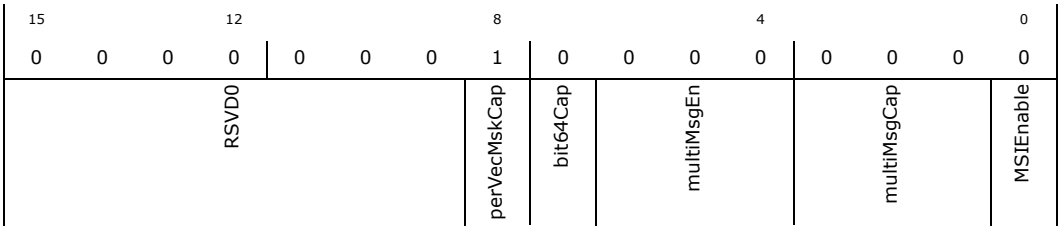
19.4.29 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:21, F:2] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector

Bit Range	Default & Access	Description
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

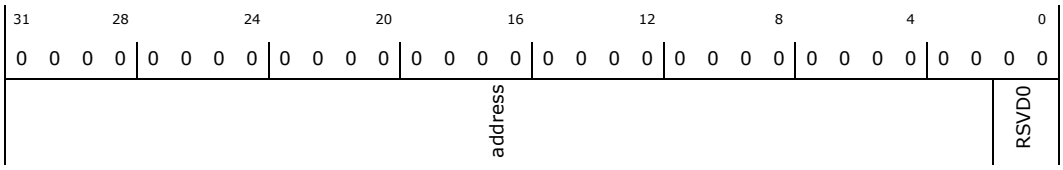
19.4.30 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:21, F:2] + A4h

Default: 00000000h



Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

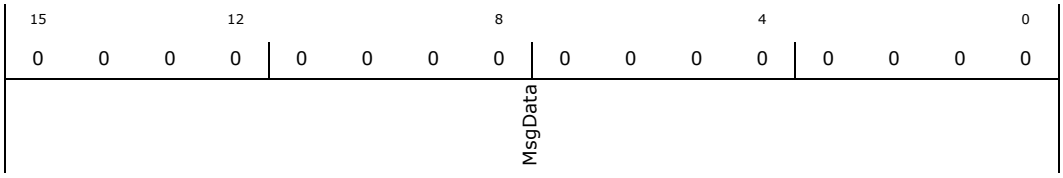
19.4.31 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:21, F:2] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

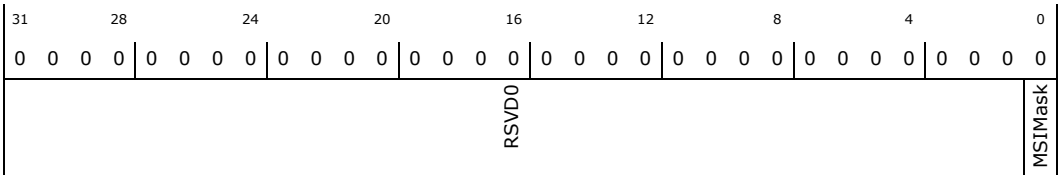
19.4.32 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

PER_VEC_MASK: [B:0, D:21, F:2] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

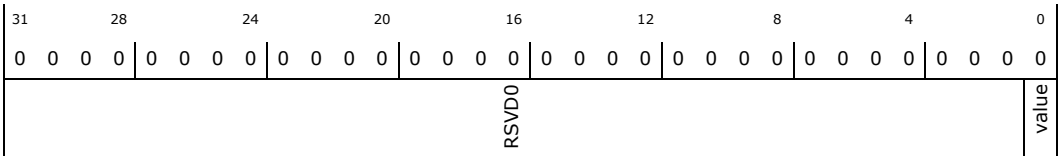
19.4.33 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

PER_VEC_PEND: [B:0, D:21, F:2] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

19.5 Memory Mapped Registers

19.5.1 I²C* Controller Memory Mapped Registers

Table 106. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Control Register (IC_CON)—Offset 0h" on page 742	00000037h
4h	7h	"Master Target Address (IC_TAR)—Offset 4h" on page 743	00000055h
10h	13h	"Data Buffer and Command (IC_DATA_CMD)—Offset 10h" on page 744	00000000h

Table 106. Summary of Memory Mapped I/O Registers—BAR0 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
14h	17h	"Standard Speed Clock SCL High Count (IC_SS_SCL_HCNT)—Offset 14h" on page 745	00000190h
18h	1Bh	"Standard Speed Clock SCL Low Count (IC_SS_SCL_LCNT)—Offset 18h" on page 746	000001D6h
1Ch	1Fh	"Fast Speed Clock SCL High Count (IC_FS_SCL_HCNT)—Offset 1Ch" on page 746	0000003Ch
20h	23h	"Fast Speed Clock SCL Low Count (IC_FS_SCL_LCNT)—Offset 20h" on page 747	00000082h
2Ch	2Fh	"Interrupt Status (IC_INTR_STAT)—Offset 2Ch" on page 747	00000000h
30h	33h	"Interrupt Mask (IC_INTR_MASK)—Offset 30h" on page 749	0000005Fh
34h	37h	"Raw Interrupt Status (IC_RAW_INTR_STAT)—Offset 34h" on page 750	00000000h
38h	3Bh	"Receive FIFO Threshold Level (IC_RX_TL)—Offset 38h" on page 752	0000000Fh
3Ch	3Fh	"Transmit FIFO Threshold Level (IC_TX_TL)—Offset 3Ch" on page 752	00000000h
40h	43h	"Clear Combined and Individual Interrupt (IC_CLR_INTR)—Offset 40h" on page 753	00000000h
44h	47h	"Clear RX_UNDER Interrupt (IC_CLR_RX_UNDER)—Offset 44h" on page 753	00000000h
48h	4Bh	"Clear RX_OVER Interrupt (IC_CLR_RX_OVER)—Offset 48h" on page 754	00000000h
4Ch	4Fh	"Clear TX_OVER Interrupt (IC_CLR_TX_OVER)—Offset 4Ch" on page 754	00000000h
50h	53h	"Clear RD_REQ Interrupt (IC_CLR_RD_REQ)—Offset 50h" on page 755	00000000h
54h	57h	"Clear TX_ABRT Interrupt (IC_CLR_TX_ABRT)—Offset 54h" on page 755	00000000h
5Ch	5Fh	"Clear ACTIVITY Interrupt (IC_CLR_ACTIVITY)—Offset 5Ch" on page 756	00000000h
60h	63h	"Clear STOP_DET Interrupt (IC_CLR_STOP_DET)—Offset 60h" on page 756	00000000h
64h	67h	"Clear START_DET Interrupt (IC_CLR_START_DET)—Offset 64h" on page 757	00000000h
6Ch	6Fh	"Enable (IC_ENABLE)—Offset 6Ch" on page 757	00000000h
70h	73h	"Status (IC_STATUS)—Offset 70h" on page 758	00000006h
74h	77h	"Transmit FIFO Level (IC_TXFLR)—Offset 74h" on page 759	00000000h
78h	7Bh	"Receive FIFO Level (IC_RXFLR)—Offset 78h" on page 760	00000000h
7Ch	7Fh	"SDA Hold (IC_SDA_HOLD)—Offset 7Ch" on page 760	00000001h
80h	83h	"Transmit Abort Source (IC_TX_ABRT_SOURCE)—Offset 80h" on page 761	00000000h
9Ch	9Fh	"Enable Status (IC_ENABLE_STATUS)—Offset 9Ch" on page 762	00000000h
A0h	A3h	"SS and FS Spike Suppression Limit (IC_FS_SPKLEN)—Offset A0h" on page 763	00000007h

19.5.1.1 Control Register (IC_CON)—Offset 0h

Can be written only when the I2C is disabled (IC_ENABLE==0). Writes while the I2C controller is enabled have no effect.

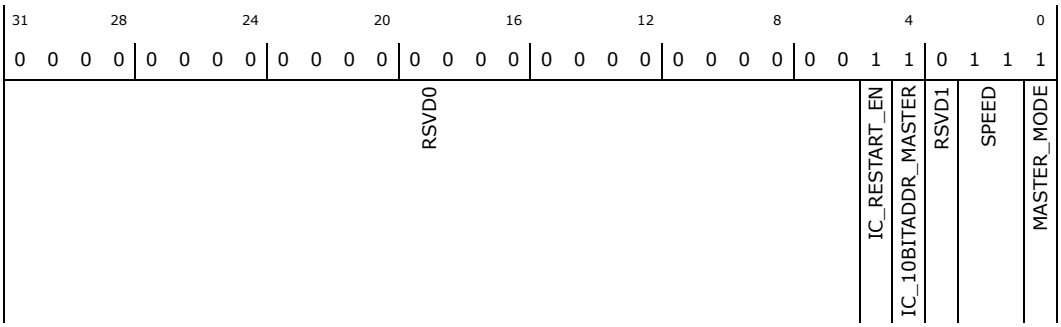
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000037h



Bit Range	Default & Access	Field Name (ID): Description
31:6	0b RO	**RSVD0:** Reserved
5	1b RW	**Restart Support (IC_RESTART_EN):** Determines whether RESTART conditions may be sent when acting as a master. Some older slaves do not support handling RESTART conditions; however, RESTART conditions are used in several I2C controller operations. 0: disable 1: enable When RESTART is disabled, the master is prohibited from performing the following functions: - Change direction within a transfer (split) - Send a START BYTE - Combined format transfers in 7-bit addressing modes - Read operation with a 10-bit address - Send multiple bytes per transfer By replacing RESTART condition followed by a STOP and a subsequent START condition, split operations are broken down into multiple I2C transfers. If the above operations are performed, it will result in setting TX_ABRT of the IC_RAW_INTR_STAT register
4	1b RW	**Master Addressing Mode (IC_10BITADDR_MASTER):** Controls whether the I2C controller starts its transfers in 7- or 10-bit addressing mode when acting as a master. 0: 7-bit addressing 1: 10-bit addressing
3	0b RO	**RSVD1:** Reserved
2:1	11b RW	**Speed Mode (SPEED):** I2C Master operational speed. Relevant only if Master is enabled (MASTER_MODE==1). 01: standard mode (100 kbit/s) 10: fast mode (400 kbit/s)
0	1b RW	**Master Mode Enable (MASTER_MODE):** controls whether the I2C master is enabled. 0: master disabled 1: master enabled

19.5.1.2 Master Target Address (IC_TAR)—Offset 4h

Can be written only when the I2C is disabled (IC_ENABLE==0). Writes while the I2C controller is enabled have no effect.

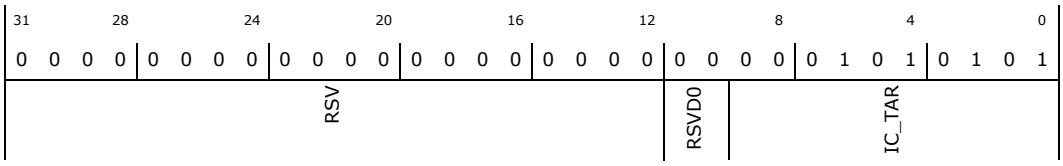
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000055h



Bit Range	Default & Access	Field Name (ID): Description
31:12	00000h RO	**Reserved (RSV):** Reserved.
11:10	0b RO	**RSVD0:** Reserved
9:0	055h RW	**Master Target Address (IC_TAR):** This is the target address for any master transaction. To generate a START BYTE, the CPU needs to write only once into these bits. If the IC_TAR and IC_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to only a slave.

19.5.1.3 Data Buffer and Command (IC_DATA_CMD)—Offset 10h

CPU writes to it when filling the TX FIFO and the reads from when retrieving bytes from RX FIFO.

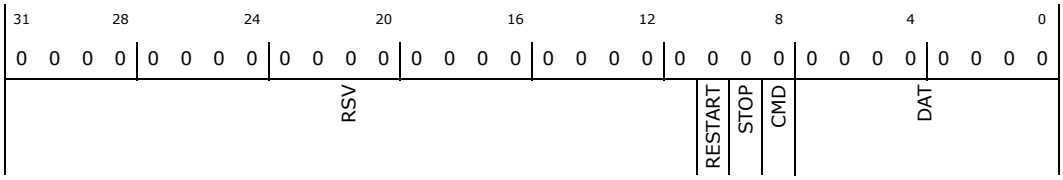
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0b RO	**Reserved (RSV):** Reserved.
10	0b WO	**Restart Bit Control (RESTART):** This bit controls whether a RESTART is issued before the byte is sent or received. - 1 if IC_RESTART_EN is 1, a RESTART is issued before the data is sent/received (according to the value of CMD), regardless of whether or not the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead. - 0 If IC_RESTART_EN is 1, a RESTART is issued only if the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead

Bit Range	Default & Access	Field Name (ID): Description
9	0b WO	**Stop Bit Control (STOP):** This bit controls whether a STOP is issued after the byte is sent or received: - 1 STOP is issued after this byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. - 0 STOP is not issued after this byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master continues the current transfer by sending/ receiving data bytes according to the value of the CMD bit. If the Tx FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the Tx FIFO.
8	0b WO	**Command (CMD):** This bit controls whether a read or a write is performed. This bit controls the direction only in I2C master mode. 0 = Write 1 = Read When a command is entered in the TX FIFO, this bit distinguishes the write and read commands. If a '1' is written to this bit after receiving a RD_REQ interrupt, then a TX_ABRT interrupt occurs. NOTE: It is possible that while attempting a master I2C read transfer, a RD_REQ interrupt may have occurred simultaneously due to a remote I2C master addressingI2C controller. In this type of scenario, the I2C controller ignores the IC_DATA_CMD write, generates a TX_ABRT interrupt, and waits to service the RD_REQ interrupt.
7:0	0b RW	**Data Buffer (DAT):** Contains the data to be transmitted or received on the I2C bus. When writing to this register and want to perform a read, DAT field is ignored by the I2C controller. When reading this register, DAT return the value of data received on the I2C controller interface.

19.5.1.4 Standard Speed Clock SCL High Count (IC_SS_SCL_HCNT)—Offset 14h

Sets the SCL clock high-period count for standard speed (SS). Can be written only when the I2C is disabled (IC_ENABLE==0). Writes at other times have no effect.

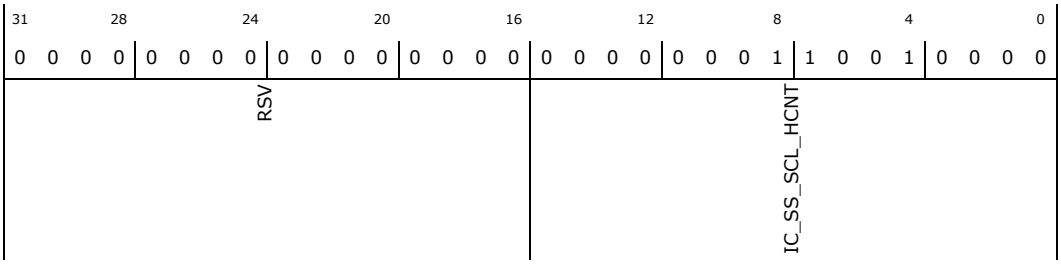
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 14h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000190h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
15:0	0190h RW	**SS SCL clock high-period count (IC_SS_SCL_HCNT):** Must be set before any I2C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. This register must not be programmed to a value higher than 65525, because I2C controller uses a 16-bit counter to flag an I2C bus idle condition when this counter reaches a value of IC_SS_SCL_HCNT + 10.

19.5.1.5 Standard Speed Clock SCL Low Count (IC_SS_SCL_LCNT)—Offset 18h

Sets the SCL clock low-period count for standard speed (SS). Can be written only when the I2C is disabled (IC_ENABLE==0). Writes at other times have no effect.

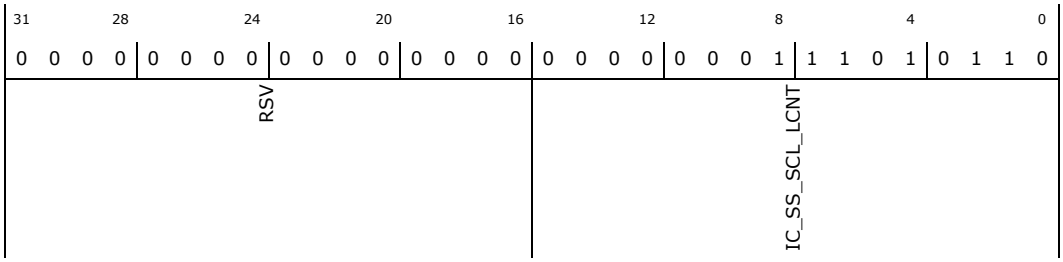
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 18h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 000001D6h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.
15:0	01d6h RW	**SS SCL clock low-period count (IC_SS_SCL_LCNT):** Must be set before any I2C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set.

19.5.1.6 Fast Speed Clock SCL High Count (IC_FS_SCL_HCNT)—Offset 1Ch

Sets the SCL clock high-period count for fast speed (FS). Can be written only when the I2C is disabled (IC_ENABLE==0). Writes at other times have no effect.

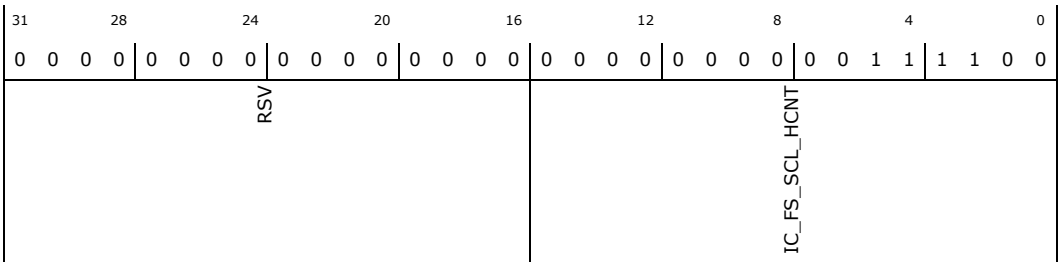
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 1Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 0000003Ch



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.
15:0	003ch RW	**FS SCL clock high-period count (IC_FS_SCL_HCNT):** Must be set before any I2C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.

19.5.1.7 Fast Speed Clock SCL Low Count (IC_FS_SCL_LCNT)—Offset 20h

Sets the SCL clock low-period count for fast speed (FS). Can be written only when the I2C is disabled (IC_ENABLE==0). Writes at other times have no effect.

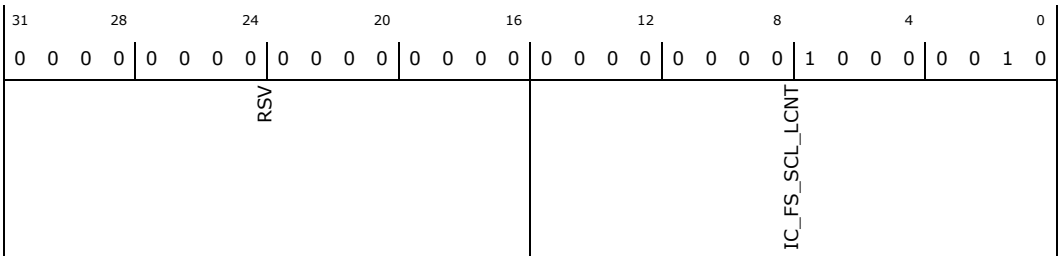
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 20h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000082h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.
15:0	0082h RW	**FS SCL clock low-period count (IC_FS_SCL_LCNT):** Must be set before any I2C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set.

19.5.1.8 Interrupt Status (IC_INTR_STAT)—Offset 2Ch

Each bit in this register has a corresponding mask bit in the IC_INTR_MASK register. These bits are cleared by reading the matching interrupt clear register. The unmasked raw versions of these bits are available in the IC_RAW_INTR_STAT register

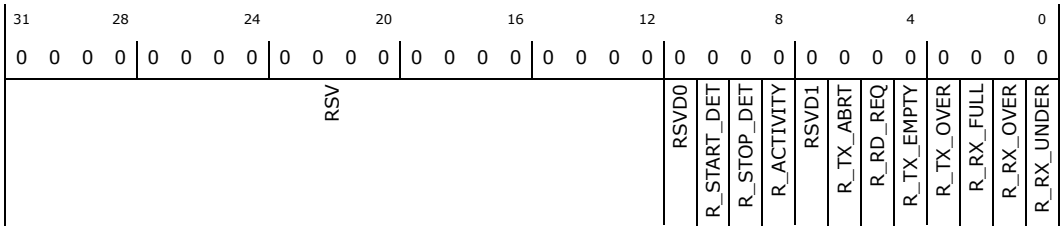
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 2Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV):** Reserved.
11	0b RO	**RSVD0:** Reserved
10	0b RO	**Start Detected (R_START_DET):** Indicates whether a START or RESTART condition has occurred on the I2C interface
9	0b RO	**Stop Detected (R_STOP_DET):** Indicates whether a STOP condition has occurred on the I2C interface.
8	0b RO	**Activity (R_ACTIVITY):** This bit captures I2C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus
7	0b RO	**RSVD1:** Reserved
6	0b RO	**TX Abort (R_TX_ABRT):** This bit indicates if the I2C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission
5	0b RO	**Read Requested (R_RD_REQ):** This bit is set to 1 when I2C controller is acting as a slave and another I2C master is attempting to read data from it. The controller holds the I2C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the IC_DATA_CMD register. This bit is set to 0 just after the processor reads the IC_CLR_RD_REQ register
4	0b RO	**TX Empty (R_TX_EMPTY):** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master state machine. When there is no longer activity, then with ic_en=0, this bit is set to 0.

Bit Range	Default & Access	Field Name (ID): Description
3	0b RO	**TX Overflow (R_TX_OVER):** Set during transmit if the transmit buffer is filled to 16 items and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master state machines goes into idle, and when ic_en goes to 0, this interrupt is cleared
2	0b RO	**RX Full (R_RX_FULL):** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues
1	0b RO	**RX Overflow (R_RX_OVER):** Set if the receive buffer is completely filled to 16 items and an additional byte is received from an external I2C device. The I2C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machines goes into idle, and when ic_en goes to 0, this interrupt is cleared.
0	0b RO	**RX Underflow (R_RX_UNDER):** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machines goes into idle, and when ic_en goes to 0, this interrupt is cleared.

19.5.1.9 Interrupt Mask (IC_INTR_MASK)—Offset 30h

These bits mask their corresponding interrupt status bits. They are active high; a value of 0 prevents a bit from generating an interrupt.

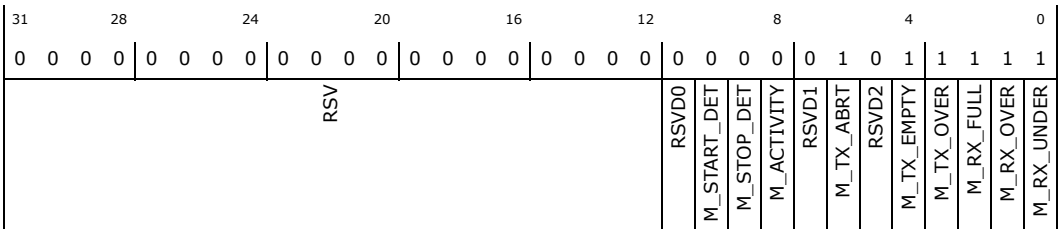
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 30h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 0000005Fh



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV):** Reserved.
11	0b RO	**RSVD0:** Reserved
10	0b RW	**Start Detected Mask (M_START_DET):** Indicates whether a START or RESTART condition has occurred on the I2C interface
9	0b RW	**Stop Detected Mask (M_STOP_DET):** Indicates whether a STOP condition has occurred on the I2C interface.

Bit Range	Default & Access	Field Name (ID): Description
8	0b RW	**Activity Mask (M_ACTIVITY):** This bit captures I2C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus
7	0b RO	**RSVD1:** Reserved
6	1b RW	**TX Abort Mask (M_TX_ABRT):** This bit indicates if the I2C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission
5	0b RO	**RSVD2:** Reserved
4	1b RW	**TX Empty Mask (M_TX_EMPTY):** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master state machine. When there is no longer activity, then with ic_en=0, this bit is set to 0. Reset value
3	1b RW	**TX Overflow Mask (M_TX_OVER):** Set during transmit if the transmit buffer is filled to 16 items and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared
2	1b RW	**RX Full Mask (M_RX_FULL):** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues
1	1b RW	**RX Overflow Mask (M_RX_OVER):** Set if the receive buffer is completely filled to 16 items and an additional byte is received from an external I2C device. The I2C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared.
0	1b RW	**RX Underflow Mask (M_RX_UNDER):** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared.

19.5.1.10 Raw Interrupt Status (IC_RAW_INTR_STAT)—Offset 34h

Unlike the IC_INTR_STAT register, these bits are not masked so they always show the true status of the I2C controller

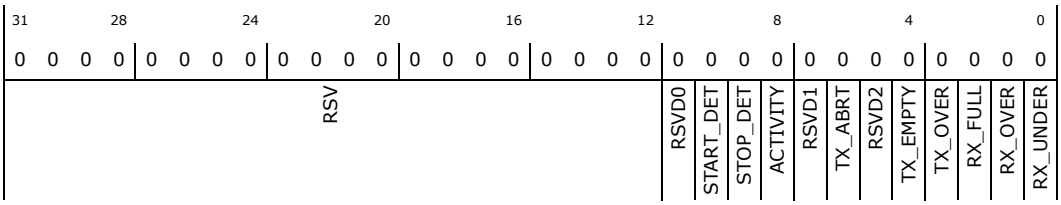
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 34h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0b RO	**Reserved (RSV):** Reserved.
11	0b RO	**RSVD0:** Reserved
10	0b RO	**Start Detected (START_DET):** Indicates whether a START or RESTART condition has occurred on the I2C interface
9	0b RO	**Stop Detected (STOP_DET):** Indicates whether a STOP condition has occurred on the I2C interface.
8	0b RO	**Activity (ACTIVITY):** This bit captures I2C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus
7	0b RO	**RSVD1:** Reserved
6	0b RO	**TX Abort (TX_ABRT):** This bit indicates if the I2C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission
5	0b RO	**RSVD2:** Reserved
4	0b RO	**TX Empty (TX_EMPTY):** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master state machine. When there is no longer activity, then with ic_en=0, this bit is set to 0.
3	0b RO	**TX Overflow (TX_OVER):** Set during transmit if the transmit buffer is filled to 16 items and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared
2	0b RO	**RX Full (RX_FULL):** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues

Bit Range	Default & Access	Field Name (ID): Description
1	0b RO	**RX Overflow (RX_OVER):** Set if the receive buffer is completely filled to 16 items and an additional byte is received from an external I2C device. The I2C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared.
0	0b RO	**RX Underflow (RX_UNDER):** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master state machine goes into idle, and when ic_en goes to 0, this interrupt is cleared.

19.5.1.11 Receive FIFO Threshold Level (IC_RX_TL)—Offset 38h

Controls the level of entries (or above) that triggers the RX_FULL interrupt (bit 2 in IC_RAW_INTR_STAT register)

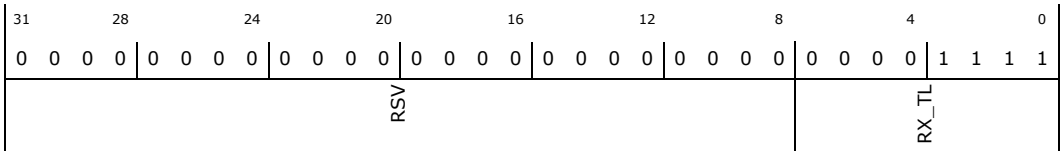
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 38h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 0000000Fh



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0Fh RW	**Receive FIFO Threshold Level (RX_TL):** The valid range is 0-255, with the additional restriction that hardware does not allow this value to be set to a value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 1 entry, and a value of 255 sets the threshold for 256 entries

19.5.1.12 Transmit FIFO Threshold Level (IC_TX_TL)—Offset 3Ch

Controls the level of entries (or below) that trigger the TX_EMPTY interrupt (bit 4 in IC_RAW_INTR_STAT register).

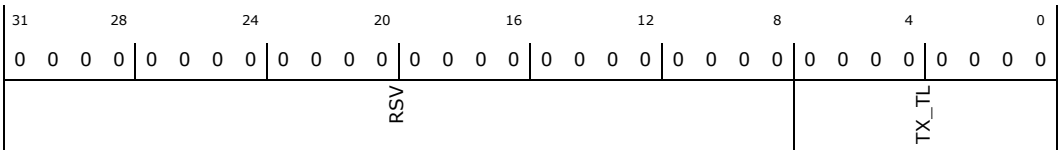
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 3Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	00h RW	**Transmit FIFO Threshold Level (TX_TL):** The valid range is 0-255, with the additional restriction that it may not be set to value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 0 entries, and a value of 255 sets the threshold for 255 entries

19.5.1.13 Clear Combined and Individual Interrupt (IC_CLR_INTR)—Offset 40h

Read this register to clear the combined interrupt, all individual interrupts, and the IC_TX_ABRT_SOURCE register.

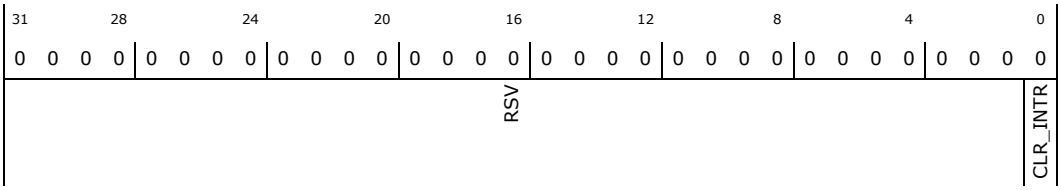
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 40h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear Combined and Individual Interrupt (CLR_INTR):** This bit does not clear hardware clearable interrupts but clears the software clearable interrupts. Refer to Bit 9 of the IC_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE

19.5.1.14 Clear RX_UNDER Interrupt (IC_CLR_RX_UNDER)—Offset 44h

Clear a single interrupt type

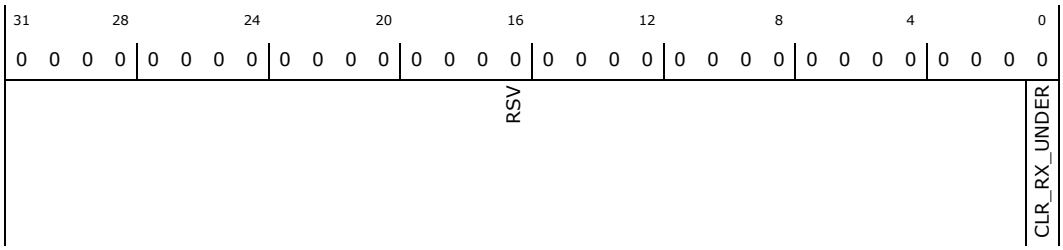
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 44h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear RX_UNDER (CLR_RX_UNDER):** Read this register to clear the RX_UNDER interrupt (bit 0) of the IC_RAW_INTR_STAT

19.5.1.15 Clear RX_OVER Interrupt (IC_CLR_RX_OVER)—Offset 48h

Clear a single interrupt type

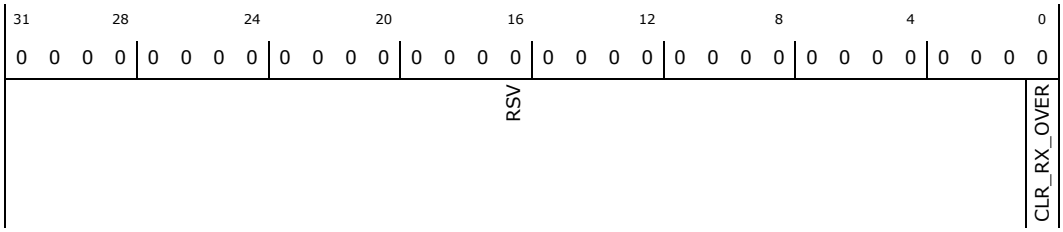
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 48h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear RX_OVER (CLR_RX_OVER):** Read this register to clear the RX_OVER interrupt (bit 1) of the IC_RAW_INTR_STAT

19.5.1.16 Clear TX_OVER Interrupt (IC_CLR_TX_OVER)—Offset 4Ch

Clear a single interrupt type

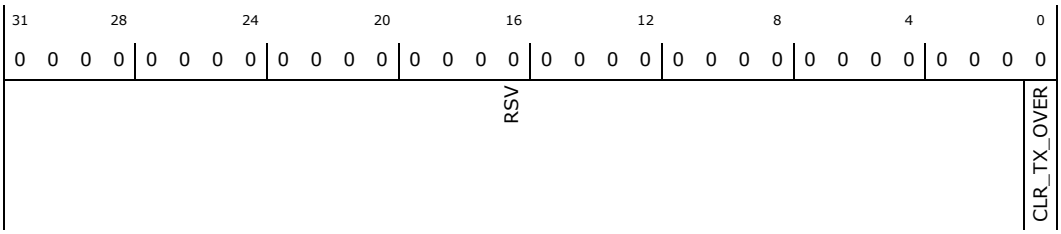
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 4Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear TX_OVER (CLR_TX_OVER):** Read this register to clear the TX_OVER interrupt (bit 3) of the IC_RAW_INTR_STAT

19.5.1.17 Clear RD_REQ Interrupt (IC_CLR_RD_REQ)—Offset 50h

Clear a single interrupt type

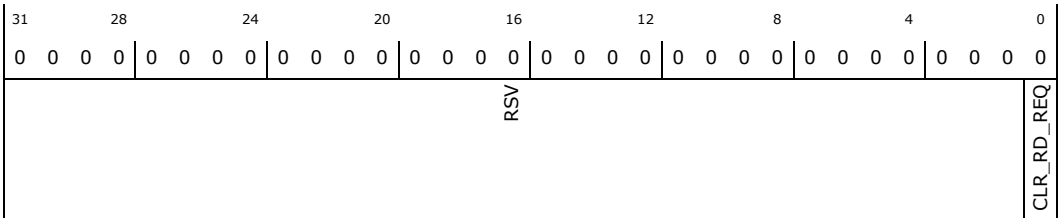
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 50h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear RD_REQ (CLR_RD_REQ):** Read this register to clear the RD_REQ interrupt (bit 5) of the IC_RAW_INTR_STAT

19.5.1.18 Clear TX_ABRT Interrupt (IC_CLR_TX_ABRT)—Offset 54h

Clear a single interrupt type

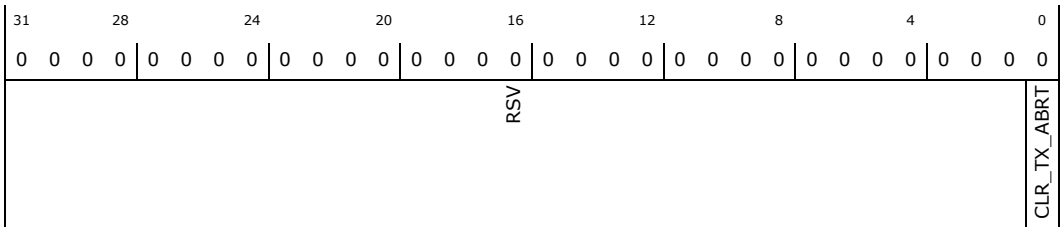
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 54h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear TX_ABRT (CLR_TX_ABRT):** Read this register to clear the TX_ABRT interrupt (bit 6) of the IC_RAW_INTR_STAT register, and the IC_TX_ABRT_SOURCE register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. Refer to Bit 9 of the IC_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE

19.5.1.19 Clear ACTIVITY Interrupt (IC_CLR_ACTIVITY)—Offset 5Ch

Clear a single interrupt type

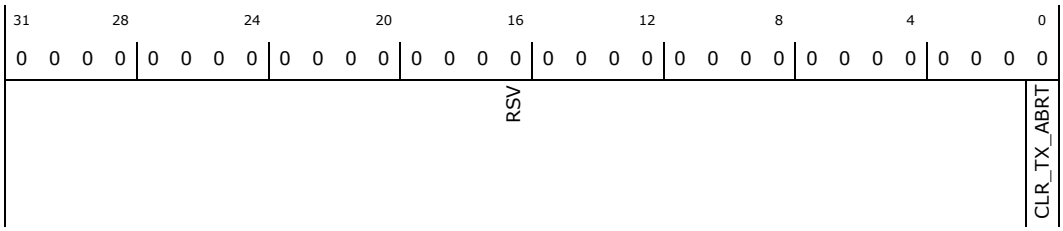
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 5Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear ACTIVITY (CLR_TX_ABRT):** Reading this register clears the ACTIVITY interrupt if the I2C is not active anymore. If the I2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT

19.5.1.20 Clear STOP_DET Interrupt (IC_CLR_STOP_DET)—Offset 60h

Clear a single interrupt type

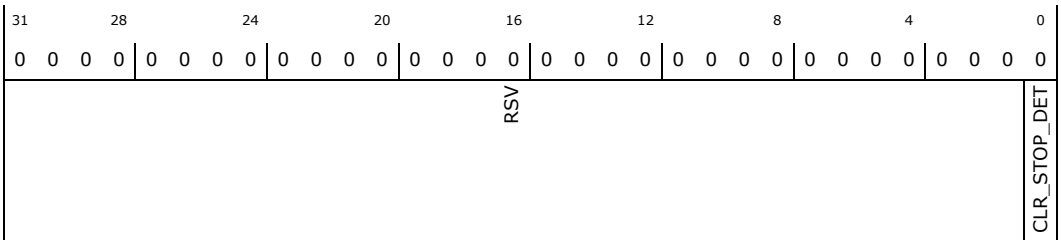
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 60h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear STOP_DET (CLR_STOP_DET):** Read this register to clear the STOP_DET interrupt (bit 9) of the IC_RAW_INTR_STAT

19.5.1.21 Clear START_DET Interrupt (IC_CLR_START_DET)—Offset 64h

Clear a single interrupt type

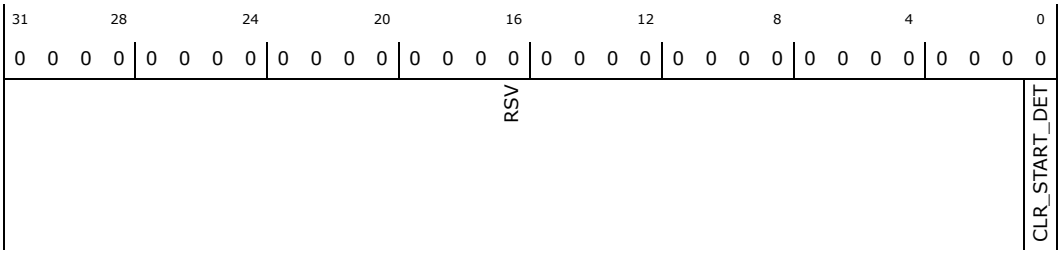
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 64h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RO	**Clear START_DET (CLR_START_DET):** Read this register to clear the START_DET interrupt (bit 10) of the IC_RAW_INTR_STAT

19.5.1.22 Enable (IC_ENABLE)—Offset 6Ch

Controls whether the I2C controller is enabled. Software can disable I2C controller while it is active.

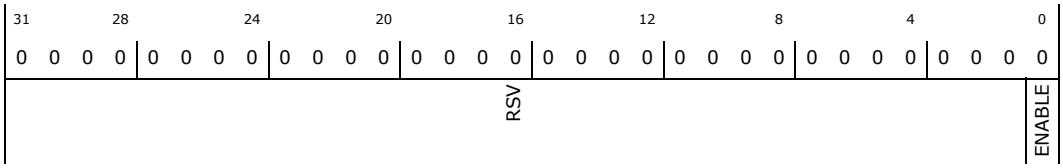
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 6Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RW	**Enable I2C Controller (ENABLE):** 0: Disabled (TX/RX FIFOs are held in an erased state) 1: Enabled NOTE: ensure that the controller is disabled properly. When disabled, the following occurs: - The TX FIFO and RX FIFO get flushed. - Status bits in the IC_INTR_STAT register are still active until the I2C Controller goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. There is a two I2C clocks delay when enabling or disabling the controller

19.5.1.23 Status (IC_STATUS)—Offset 70h

Read-only register used to indicate the current transfer status and FIFO status. The status register may be read at any time. None of the bits in this register request an interrupt. - When the I2C is disabled by writing 0 in bit 0 of the IC_ENABLE register: bits 1 and 2 are set to 1, bits 3 and 4 are set to 0 - When the master or slave state machines goes to idle and ic_en=0: bits 5 and 6 are set to 0

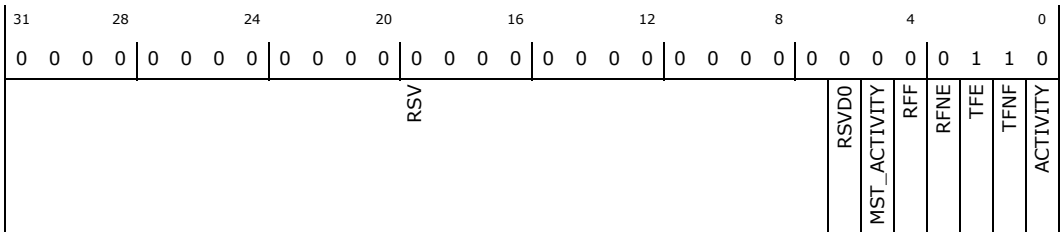
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 70h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000006h



Bit Range	Default & Access	Field Name (ID): Description
31:7	0b RO	**Reserved (RSV):** Reserved.
6	0b RO	**RSVD0:** Reserved
5	0b RO	**Master FSM Activity Status (MST_ACTIVITY):** When the Master Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0: Master FSM is in IDLE state so the Master part is not Active 1: Master FSM is not in IDLE state so the Master part is Active NOTE: IC_STATUS[0]-that is, ACTIVITY bit-is the OR of SLV_ACTIVITY and MST_ACTIVITY bits
4	0b RO	**Receive FIFO Completely Full (RFF):** When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0: Receive FIFO is not full 1: Receive FIFO is full
3	0b RO	**Receive FIFO Not Empty (RFNE):** This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0: Receive FIFO is empty 1: Receive FIFO is not empty
2	1b RO	**Transmit FIFO Completely Empty (TFE):** When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty
1	1b RO	**Transmit FIFO Not Full (TFNF):** Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0: Transmit FIFO is full 1: Transmit FIFO is not full
0	0b RO	**Activity (ACTIVITY):** I2C Activity Status. 0: Not Active 1: Active

19.5.1.24 Transmit FIFO Level (IC_TXFLR)—Offset 74h

Contains the number of valid data entries in the transmit FIFO buffer. It is cleared whenever: - The I2C is disabled - There is a transmit abort that is, TX_ABRT bit is set in the IC_RAW_INTR_STAT register - The slave bulk transmit mode is aborted. The register increments whenever data is placed into the transmit FIFO and decrements when data is taken from the transmit FIFO

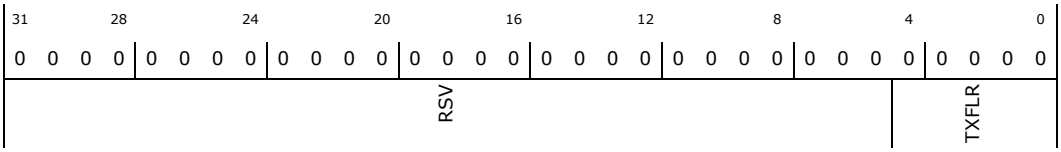
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 74h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0b RO	**Reserved (RSV):** Reserved.
4:0	0b RO	**Transmit FIFO Level (TXFLR):** Contains the number of valid data entries in the transmit FIFO

19.5.1.25 Receive FIFO Level (IC_RXFLR)—Offset 78h

Read This register contains the number of valid data entries in the receive FIFO buffer. It is cleared whenever: - The I2C is disabled - Whenever there is a transmit abort caused by any of the events tracked in IC_TX_ABRT_SOURCE. The register increments whenever data is placed into the receive FIFO and decrements when data is taken from the receive FIFO

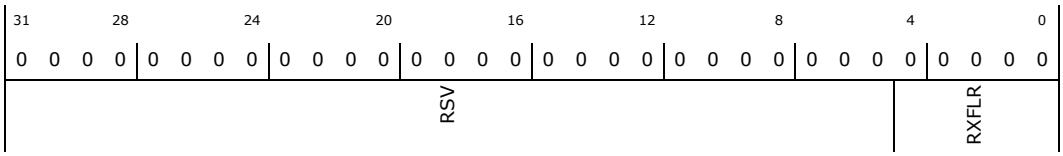
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 78h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0b RO	**Reserved (RSV):** Reserved.
4:0	0b RO	**Receive FIFO Level (RXFLR):** Contains the number of valid data entries in the receive FIFO

19.5.1.26 SDA Hold (IC_SDA_HOLD)—Offset 7Ch

This register controls the amount of hold time on the SDA signal after a negative edge of SCL line in units of I2C clock period. The value programmed must be greater than the minimum hold time in each mode for the value to be implemented: 1 cycle. Writes to this register succeed only when I2C controller is disabled (IC_ENABLE=0)

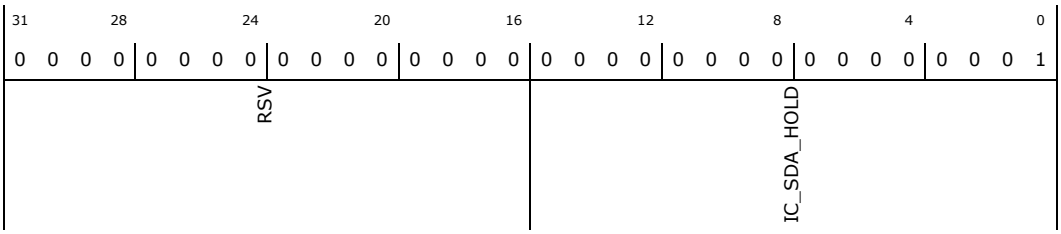
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 7Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000001h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.
15:0	0001h RW	**SDA Hold (IC_SDA_HOLD):** Sets the required SDA hold time in units of the I2C clock period.

19.5.1.27 Transmit Abort Source (IC_TX_ABRT_SOURCE)—Offset 80h

Used to indicate the source of the TX_ABRT interrupt. Except for Bit 9, this register is cleared whenever the IC_CLR_TX_ABRT register or the IC_CLR_INTR register is read. To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; RESTART must be enabled (IC_CON[5]=1), the SPECIAL bit must be cleared (IC_TAR[11]), or the GC_OR_START bit must be cleared (IC_TAR[10]). Once the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, Bit 9 clears for one cycle and is then re-asserted

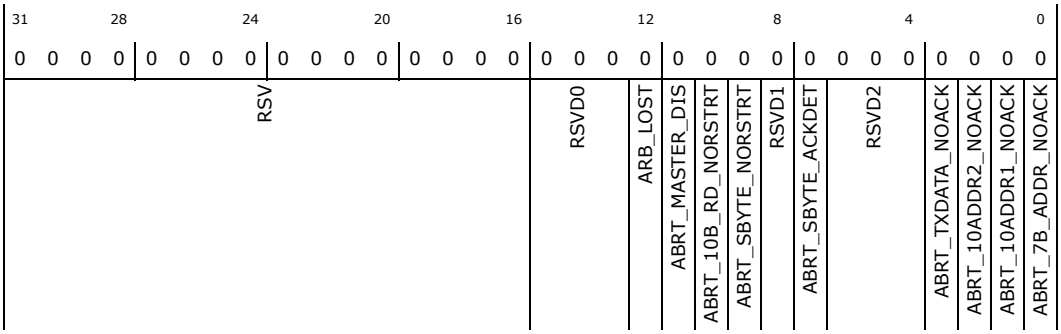
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 80h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0b RO	**Reserved (RSV):** Reserved.
15:13	0b RO	**RSVD0:** Reserved
12	0b RO	**Master Lost Arbitration (ARB_LOST):** Set if master has lost arbitration

Bit Range	Default & Access	Field Name (ID): Description
11	0b RO	**Master Disabled (ABRT_MASTER_DIS):** Set if user tries to initiate a Master operation with the Master mode disabled
10	0b RO	**10 Bit Address READ and RESTART Disabled (ABRT_10B_RD_NORSTRT):** Set if the restart is disabled (IC_RESTART_EN bit (IC_CON[5]) =0) and the master sends a read command in 10-bit addressing mode
9	0b RO	**START With RESTART Disabled (ABRT_SBYTE_NORSTRT):** To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; restart must be enabled (IC_CON[5]=1), the SPECIAL bit must be cleared (IC_TAR[11]), or the GC_OR_START bit must be cleared (IC_TAR[10]). Once the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets reasserted. Set if the restart is disabled (IC_RESTART_EN bit (IC_CON[5]) =0) and the user is trying to send a START Byte
8	0b RO	**RSVD1:** Reserved
7	0b RO	**START Acknowledged (ABRT_SBYTE_ACKDET):** Set if master has sent a START Byte and the START Byte was acknowledged (wrong behavior).
6:4	0b RO	**RSVD2:** Reserved
3	0b RO	**TX Data Not Acknowledged (ABRT_TXDATA_NOACK):** Set if master has received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s).
2	0b RO	**10 Bit Address Second Not Acknowledged (ABRT_10ADDR2_NOACK):** Set if master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave
1	0b RO	**10 Bit Address First Not Acknowledged (ABRT_10ADDR1_NOACK):** Set if master is in 10-bit address mode and the first 10-bit address byte was not acknowledged by any slave
0	0b RO	**7 Bit Address Not Acknowledged (ABRT_7B_ADDR_NOACK):** Set if master is in 7-bit addressing mode and the address sent was not acknowledged by any slave

19.5.1.28 Enable Status (IC_ENABLE_STATUS)—Offset 9Ch

Report the I2C hardware status

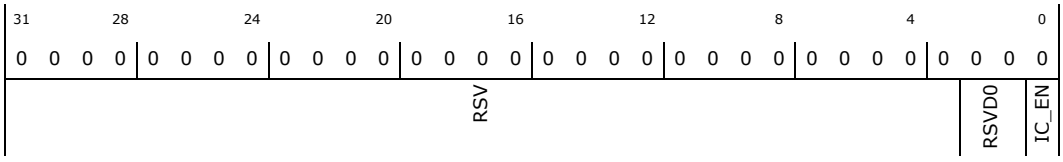
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + 9Ch

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:3	0b RO	**Reserved (RSV):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
2:1	0b RO	**RSVD0:** Reserved
0	0b RO	**I2C Enable Status (IC_EN):** When read as 1, the controller is deemed to be in an enabled state. When read as 0, the controller is deemed completely inactive. NOTE: The CPU can safely read this bit anytime.

19.5.1.29 SS and FS Spike Suppression Limit (IC_FS_SPKLEN)—Offset A0h

Used to store the duration, measured in I2C clock cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in Standard/Fast Speed modes. The relevant I2C requirement is detailed in the I2C Bus Specification. This register must be programmed with a minimum value of 2.

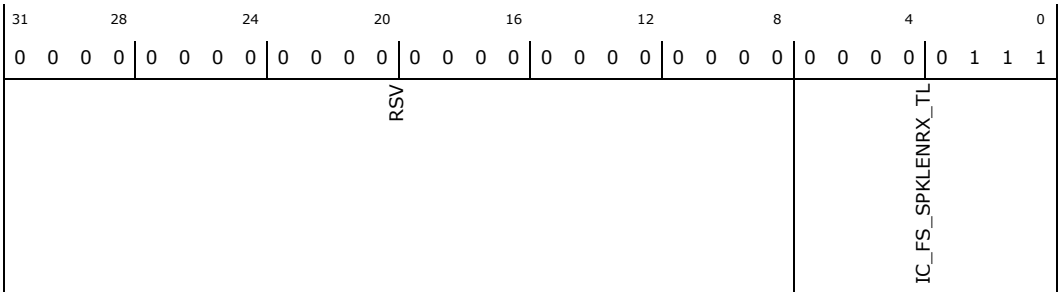
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR0] + A0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:2] + 10h

Default: 00000007h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	07h RW	**I2C SS and FS Spike Length (IC_FS_SPKLENRX_TL):** Must be set before any I2C bus transaction can take place to ensure stable operation.

19.5.2 GPIO Controller Memory Mapped Registers

Table 107. Summary of Memory Mapped I/O Registers—BAR1

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
0h	3h	"Port A Data (GPIO_SWPORTA_DR)—Offset 0h" on page 764	00000000h
4h	7h	"Port A Data Direction (GPIO_SWPORTA_DDR)—Offset 4h" on page 764	00000000h
30h	33h	"Interrupt Enable (GPIO_INTEN)—Offset 30h" on page 765	00000000h
34h	37h	"Interrupt Mask (GPIO_INTMASK)—Offset 34h" on page 765	00000000h

Table 107. Summary of Memory Mapped I/O Registers—BAR1 (Continued)

Offset Start	Offset End	Register Name (Register Symbol)	Default Value
38h	3Bh	"Interrupt Type (GPIO_INTTYPE_LEVEL)—Offset 38h" on page 766	00000000h
3Ch	3Fh	"Interrupt Polarity (GPIO_INT_POLARITY)—Offset 3Ch" on page 767	00000000h
40h	43h	"Interrupt Status (GPIO_INTSTATUS)—Offset 40h" on page 767	00000000h
44h	47h	"Raw Interrupt Status (GPIO_RAW_INTSTATUS)—Offset 44h" on page 768	00000000h
48h	4Bh	"Debounce Enable (GPIO_DEBOUNCE)—Offset 48h" on page 768	00000000h
4Ch	4Fh	"Clear Interrupt (GPIO_PORTA_EOI)—Offset 4Ch" on page 769	00000000h
50h	53h	"Port A External Port (GPIO_EXT_PORTA)—Offset 50h" on page 770	00000000h
60h	63h	"Synchronization Level (GPIO_LS_SYNC)—Offset 60h" on page 770	00000000h

19.5.2.1 Port A Data (GPIO_SWPORTA_DR)—Offset 0h

Contains the GPIO Port data bits

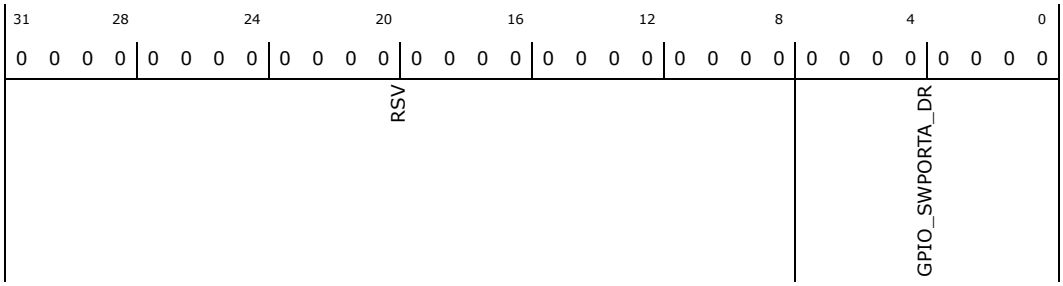
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 0h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Port Data (GPIO_SWPORTA_DR):** Values written to this register are output on the I/O signals for if the corresponding data direction bits are set to Output mode and the corresponding control bit for the Port is set to Software mode. The value read back is equal to the last value written to this register

19.5.2.2 Port A Data Direction (GPIO_SWPORTA_DDR)—Offset 4h

Used to control the GPIO Port bits data direction

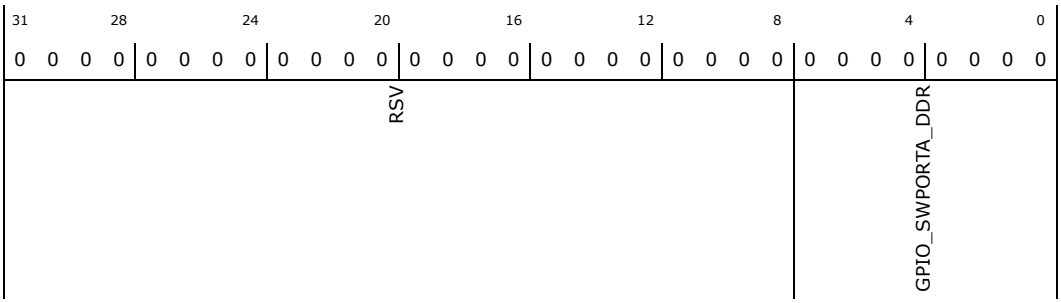
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 4h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Port Data Direction (GPIO_SWPORTA_DDR):** Values written to this register independently control the direction of the corresponding data bit in the Port - 0 Input (default) - 1 Output

19.5.2.3 Interrupt Enable (GPIO_INTEN)—Offset 30h

Used to configured Port A bits as interrupt sources

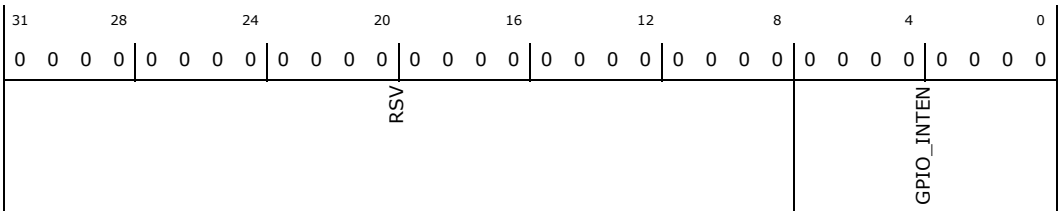
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 30h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Interrupt Enable (GPIO_INTEN):** Allows each bit of Port A to be configured for interrupts. By default the generation of interrupts is disabled. Whenever a 1 is written to a bit of this register, it configures the corresponding bit on Port A to become an interrupt; otherwise, Port A operates as a normal GPIO signal. Interrupts are disabled on the corresponding bits of Port A if the corresponding data direction register is set to Output. 0 Configure Port A bit as normal GPIO signal (default) 1 Configure Port A bit as interrupt

19.5.2.4 Interrupt Mask (GPIO_INTMASK)—Offset 34h

Controls masking for Port A bits configured as interrupt sources

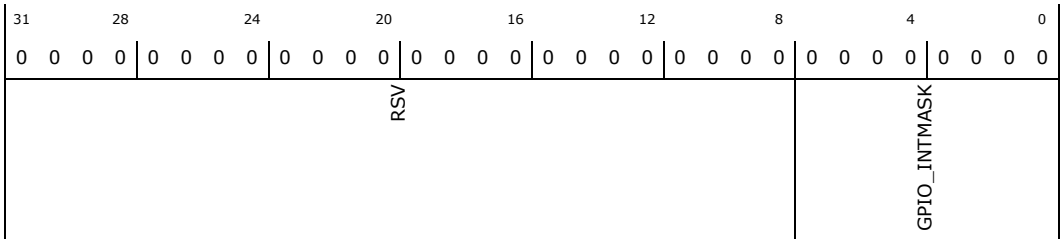
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 34h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Interrupt Mask (GPIO_INTMASK):** Controls whether an interrupt on Port A can create an interrupt for the interrupt controller by not masking it. By default, all interrupts bits are unmasked. Whenever a 1 is written to a bit in this register, it masks the interrupt generation capability for this signal; otherwise interrupts are allowed through. The unmasked status can be read as well as the resultant status after masking. 0 Interrupt bits are unmasked (default) 1 Mask interrupt

19.5.2.5 Interrupt Type (GPIO_INTTYPE_LEVEL)—Offset 38h

Controls the type of interrupt associated with Port A bits configured as interrupt source

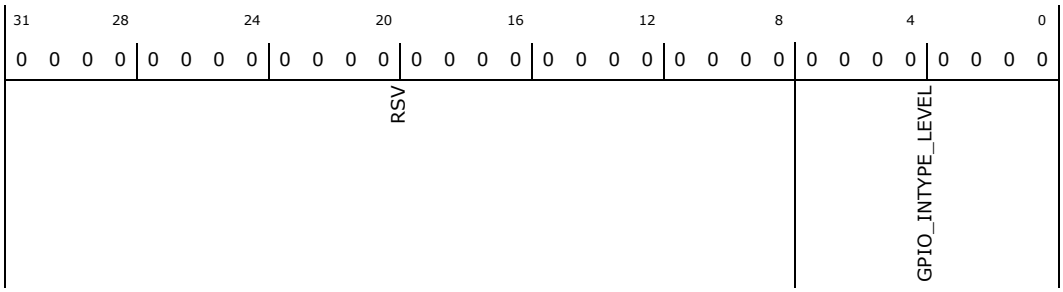
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 38h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**RSV:** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
7:0	0b RW	**Interrupt Type (GPIO_INTYPE_LEVEL):** Controls the type of interrupt that can occur on Port A. Whenever a 0 is written to a bit of this register, it configures the interrupt type to be level-sensitive; otherwise, it is edge-sensitive. 0 Level-sensitive (default) 1 Edge-sensitive

19.5.2.6 Interrupt Polarity (GPIO_INT_POLARITY)—Offset 3Ch

Controls the interrupt polarity associated with Port A bits configured as interrupt sources

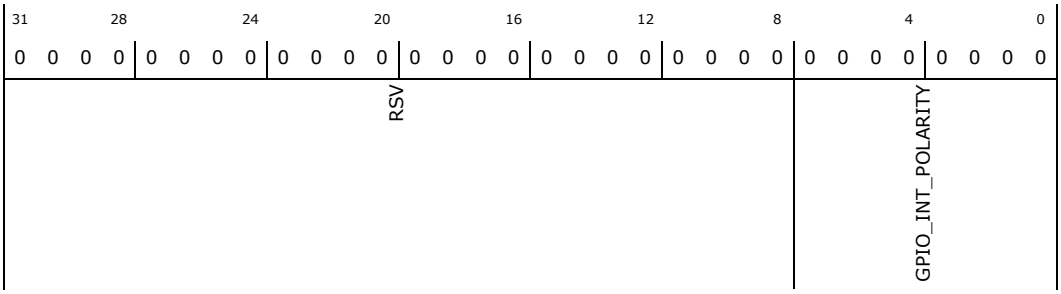
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 3Ch

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Interrupt Polarity (GPIO_INT_POLARITY):** Controls the polarity of edge or level sensitivity that can occur on input of Port A. Whenever a 0 is written to a bit of this register, it configures the interrupt type to falling-edge or active-low sensitive; otherwise, it is rising-edge or active-high sensitive. 0 Active-low (default) 1 Active-high

19.5.2.7 Interrupt Status (GPIO_INTSTATUS)—Offset 40h

Stores the interrupt status after masking for Port A bits configured as interrupt sources

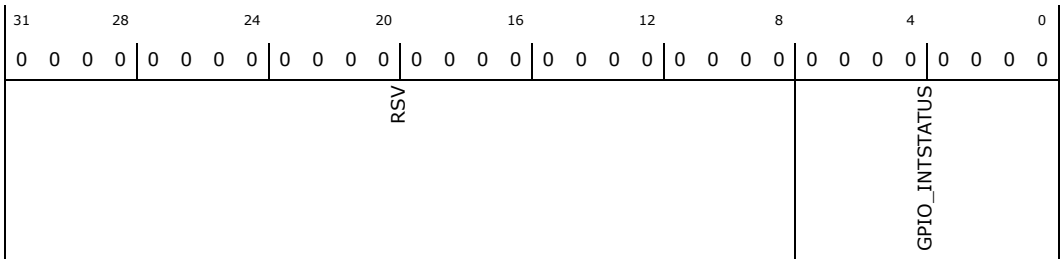
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 40h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Interrupt Status (GPIO_INTSTATUS):** After mask. See GPIO_RAW_INTSTATUS for raw interrupt values and GPIO_INTMASK for interrupt mask configuration

19.5.2.8 Raw Interrupt Status (GPIO_RAW_INTSTATUS)—Offset 44h

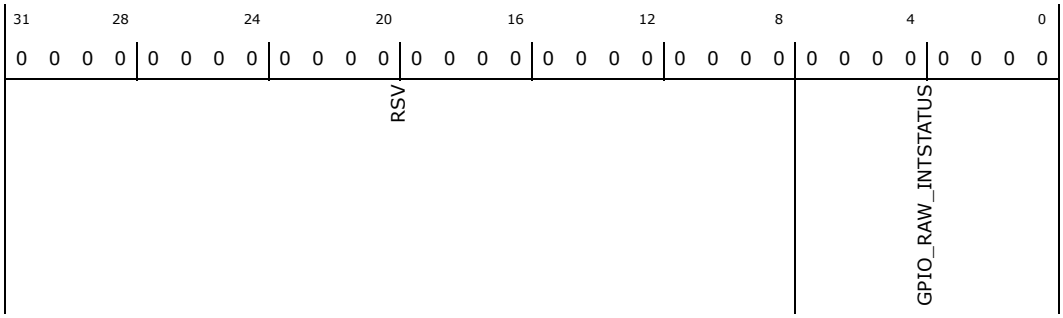
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 44h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Raw Interrupt Status (GPIO_RAW_INTSTATUS):** Raw interrupt of status of Port A (premasking bits)

19.5.2.9 Debounce Enable (GPIO_DEBOUNCE)—Offset 48h

Controls the debounce logic associated to a Port A bit configured as interrupt source

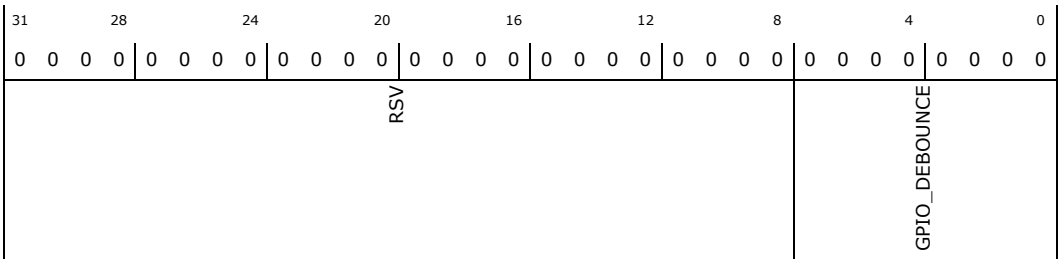
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 48h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**Debounce Enable (GPIO_DEBOUNCE):** Controls whether an external signal that is the source of an interrupt needs to be debounced to remove any spurious glitches. Writing a 1 to a bit in this register enables the debouncing circuitry. A signal must be valid for two periods of an external clock before it is internally processed. 0 No debounce (default) 1 Enable debounce

19.5.2.10 Clear Interrupt (GPIO_PORTA_EOI)—Offset 4Ch

Controls edge-type interrupt clearing

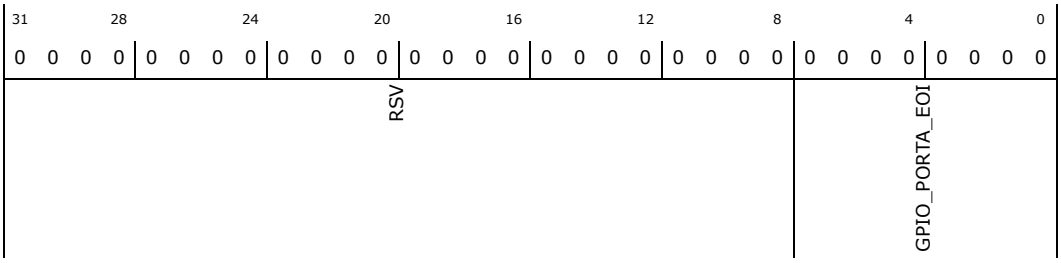
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 4Ch

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.

Bit Range	Default & Access	Field Name (ID): Description
7:0	0b RW	**Clear Interrupt (GPIO_PORTA_EOI):** Controls the clearing of edge type interrupts from Port A. When a 1 is written into a corresponding bit of this register, the interrupt is cleared. All interrupts are cleared when Port A is not configured for interrupts. 0 No interrupt clear (default) 1 Clear interrupt

19.5.2.11 Port A External Port (GPIO_EXT_PORTA)—Offset 50h

Used by the software to read values from the GPIO Port bits

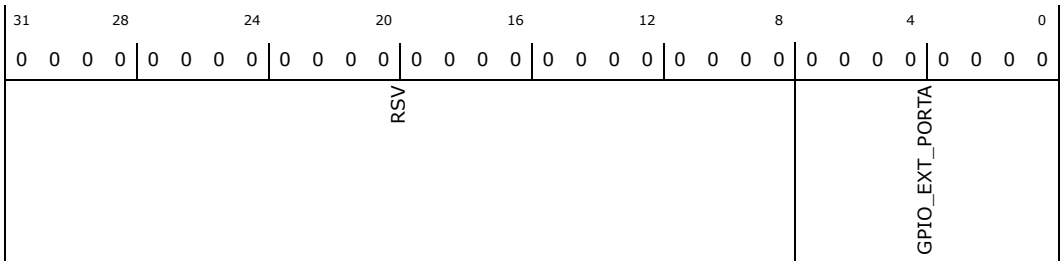
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 50h

BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0b RO	**Reserved (RSV):** Reserved.
7:0	0b RW	**External Port (GPIO_EXT_PORTA):** When the Port is configured as Input, then reading this location reads the values on the external signal. When the data direction is set as Output, reading this location reads the Port data register contents

19.5.2.12 Synchronization Level (GPIO_LS_SYNC)—Offset 60h

Controls if a level-sensitive interrupt type need to be synchronized to the system clock

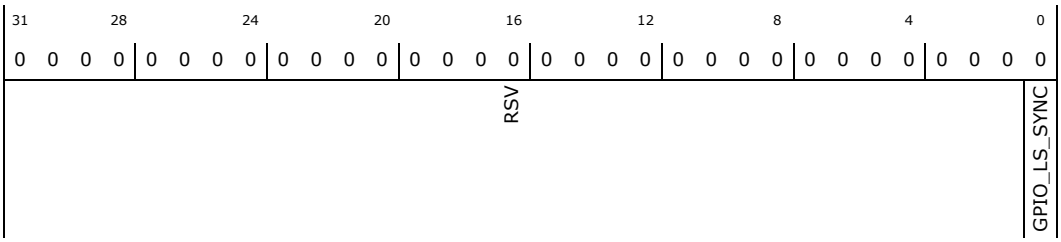
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

Offset: [BAR1] + 60h

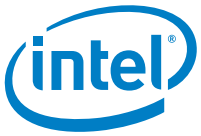
BAR1 Type: PCI Configuration Register (Size: 32 bits)
BAR1 Reference: [B:0, D:21, F:2] + 14h

Default: 00000000h

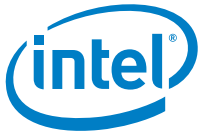


Bit Range	Default & Access	Field Name (ID): Description
31:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RW	**Synchronization Level (GPIO_LS_SYNC):** Writing a 1 to this register results in all level-sensitive interrupts being synchronized to the system clock. 0 Not Synchronized (default) 1 Synchronized

§ §



October 2013
Document Number: 329676-001US



20.0 SPI Interface

The Intel® Quark SoC X1000 implements two SPI controllers that support master mode.

20.1 Signal Descriptions

Please see Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 108. SPI Interface Signals

Signal Name	Direction/ Type	Description
SPI[0/1]_SCK	O CMOS3.3	**SPI Serial Clock**
SPI[[0/1]_SS_B	O CMOS3.3	**SPI Slave Select** SPI Slave Select is active low.
SPI[0/1]_MOSI	O CMOS3.3	**SPI Master Output Slave Input**
SPI[0/1]_MISO	I CMOS3.3	**SPI Slave Output Master Input**

20.2 Features

Compliant with the Motorola* Serial Peripheral Interface:

- supports master mode only
- supports one slave select output
- supports MSB first transfer only
- supports SCLK frequencies up to 25 MHz
- does not support slave mode operation
- does not support SPI boot
- does not support wait mode

20.2.1 SPI Controller

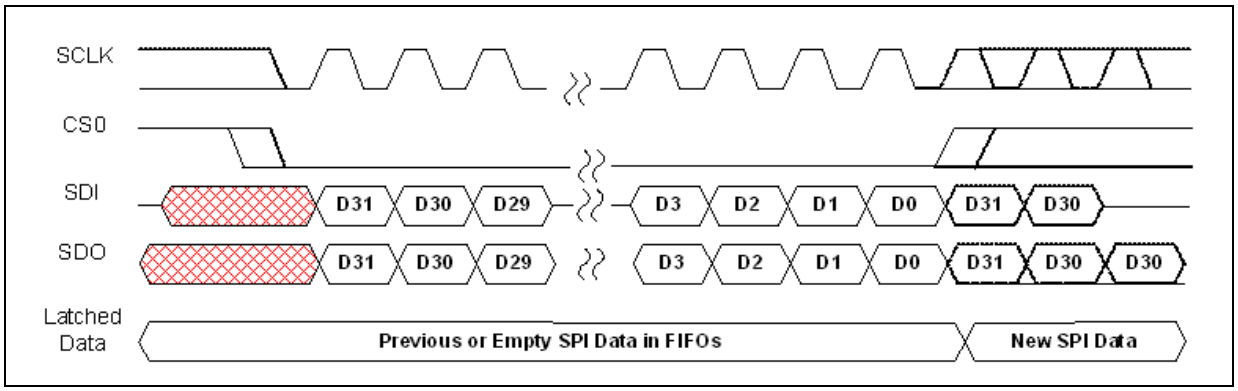
The SPI unit provides one-channel, 3-wire serial input/output interface to directly connect SPI-compatible devices.

Four pins are used to transfer data between the CPU and external device:

- SCLK defines the bit rate at which serial data is driven onto, and sampled from, the bus;
- SS_B or CS defines the boundaries of a basic data "unit", comprised of multiple serial bits.
- MOSI or SDO is the serial data path for outbound data, from system to peripheral
- MISO or SDI is the serial data path for inbound data, from peripheral to system

Serial data is transferred between the system and an external peripheral through FIFOs in the SPI. Transfers are initiated by the host processor and data is transferred over a single transfer. Operation is full duplex. Separate FIFOs and serial data paths allow simultaneous transfers in both directions to/from the external peripheral. Transfer is started when either new data is available in the transmit FIFO or memory is available in the receive FIFO. Transfer is terminated when either new data is not available in the transmit FIFO or memory is not available in the receive FIFO.

Figure 43. Generic SPI Waveform



20.2.1.1 Processor-Initiated Data Transfer

Transmit data (from system to peripheral) is written by the host processor to the SPI Transmit FIFO. The FIFO is seen as one 32-bit location by the processor and a write to the FIFO takes the form of a single 32-bit write transaction to the data register (SSDR). The SPI moves the data from the Transmit FIFO, serializes it, and sends it over the serial wire (SDO) to the peripheral. Receive data from the peripheral (on SDI) is converted to parallel words and stored in the Receive FIFO register file. When passed, a programmable "fullness" threshold triggers an interrupt to the Interrupt Controller (and subsequently, if enabled, to an interrupt input to the CPU). The interrupt service routine responds by identifying the source of the interrupt and then doing a read from the Receive FIFO. The SPI differentiates between the two FIFOs according to whether the transfer is a READ or a WRITE transfer. Read bursts automatically target the Receive FIFO, while write bursts write data to the Transmit FIFO. From a memory map point of view, they are at the same address. FIFOs are 32DWORDS deep or 128 Bytes in aggregate.

20.2.1.2 Data Format

Data in the FIFOs is always stored with one sample per 32-bit word regardless of the format's data word length. Within each 32-bit field, the stored data sample is right justified, with the word's LSB in bit 0, and unused bits packed as zeroes on the left-hand (MSB) side. Logic in the SPI guarantees that data is properly transmitted on SDO according to the selected frame format and MSB first.

20.2.1.3 FIFO Operation

20.2.1.3.1 Processor-Initiated Data Transfers

There are two separate and independent FIFOs for "incoming" (from peripheral) and "outgoing" (to peripheral) serial data. FIFOs are filled or emptied by single transfers or SRAM-like bursts initiated by the system processor. Both FIFO's are the same size and arranged as 32-word positions that are a maximum of 32-bits wide. FIFO word width is programmable from 4 to 32 bits. When FIFO width is programmed to less than a 32-bit width, only the programmed numbers of bits are available in all word positions.

Each FIFO consists of a dual-port register file with control circuitry to make it work as a FIFO, with independent read and write ports. Single FIFO write bursts may be between 1 and 32 words in length, and between 4 and 32 bits per word thus transferring from 4 bits up to 1024 data samples per burst. Also continuous operation is possible by keeping the transmit FIFO loaded with data for much larger data transfers.

FIFO filling and emptying may be performed by the system processor in response to an interrupt from the FIFO logic. Each FIFO has a programmable threshold at which an interrupt is triggered. When the threshold value is exceeded, an interrupt is generated which, if enabled, signals the host processor to empty an "inbound" FIFO or to refill an "outbound" FIFO.

The system can also poll status bits to learn how full a FIFO is.

20.2.1.4 Baud Rate Generation

The SCLK is generated in the SPI unit from the 200 MHz input system clock (sys_clk) according to register configuration.

Two clock synthesis stages are implemented to achieve different SPI baud-rates:

- an internal clock (clk_ssp) is generated from the input reference clock based on the value of SPI_DDS_RATE.DDS_CLK_RATE[23:0].clk_ssp is used in the SPI unit to clock the logic interfacing the SPI link and to generate SCLK. The relationship between sys_clk and ssp_clk frequencies is expressed by
  $f_{clk_ssp} = f_{sys_clk}$ (DDS_CLK_RATE/2^{24})
- clk_ssp is used to generate SCLK. The relationship between ssp_clk and SCLK frequencies is expressed by
  $f_{SCLK} = f_{clk_ssp}$ /(2*(SCR+1))

Table 109 lists a subset of the possible register configurations and relative SCLK frequencies. While the table is not exhaustive of all possible register settings, the following guidelines should be considered:

- To guarantee negligible jitter and duty cycle deviation from 50% it is recommended to not set DDS_CLK_RATE to values greater than h33333 and not listed in the table.
- A duty cycle of 50% and virtually no jitter is guaranteed for any setting where DDS_CLK_RATE is one hot.
- For a given SCLK frequency not listed in Table 109, best jitter and duty cycle values are obtained by setting the lower possible value of DDS_CLK_RATE.
- There is no limitation for SCR values.

Table 109. SPI Clock Frequency Settings (Sheet 1 of 2)

DDS_CLK_RATE[23:0] (hex)	SCLK_INT Frequency	SCR[7:0] (dec)	SCLK Frequency (MHz)	SCLK Frequency (KHz)	Duty Cycle (%)
800000	100.000	0	25.000		50
800000	100.000	2	16.667		50
800000	100.000	4	10.000		50
666666	80.000	0	40.000		60-40
666666	80.000	1	20.000		50
666666	80.000	2	13.333		54-46
666666	80.000	4	8.000		52-48
666666	80.000	9	4.000		50
666666	80.000	19	2.000		50
666666	80.000	49		800.000	50
666666	80.000	99		400.000	50
666666	80.000	199		200.000	50
400000	50.000	0	25.000		50
400000	50.000	3	6.250		50
400000	50.000	4	5.000		50
400000	50.000	9	2.500		50
400000	50.000	24	1.000		50
400000	50.000	49		500.000	50
400000	50.000	99		250.000	50
200000	25.000	0	12.500		50
200000	25.000	4	2.500		50
200000	25.000	9	1.250		50
200000	25.000	19		625.000	50
200000	25.000	24		500.000	50
200000	25.000	49		250.000	50
200000	25.000	124		100.000	50
100000	12.500	0	6.250		50
100000	12.500	49		125.000	50
100000	12.500	124		50.000	50
80000	6.250	0	3.125		50
80000	6.250	124		25.000	50
40000	3.125	0	1.563		50
20000	1.563	0		781.250	50
20000	1.563	77		10.016	50
20000	1.563	154		5.040	50

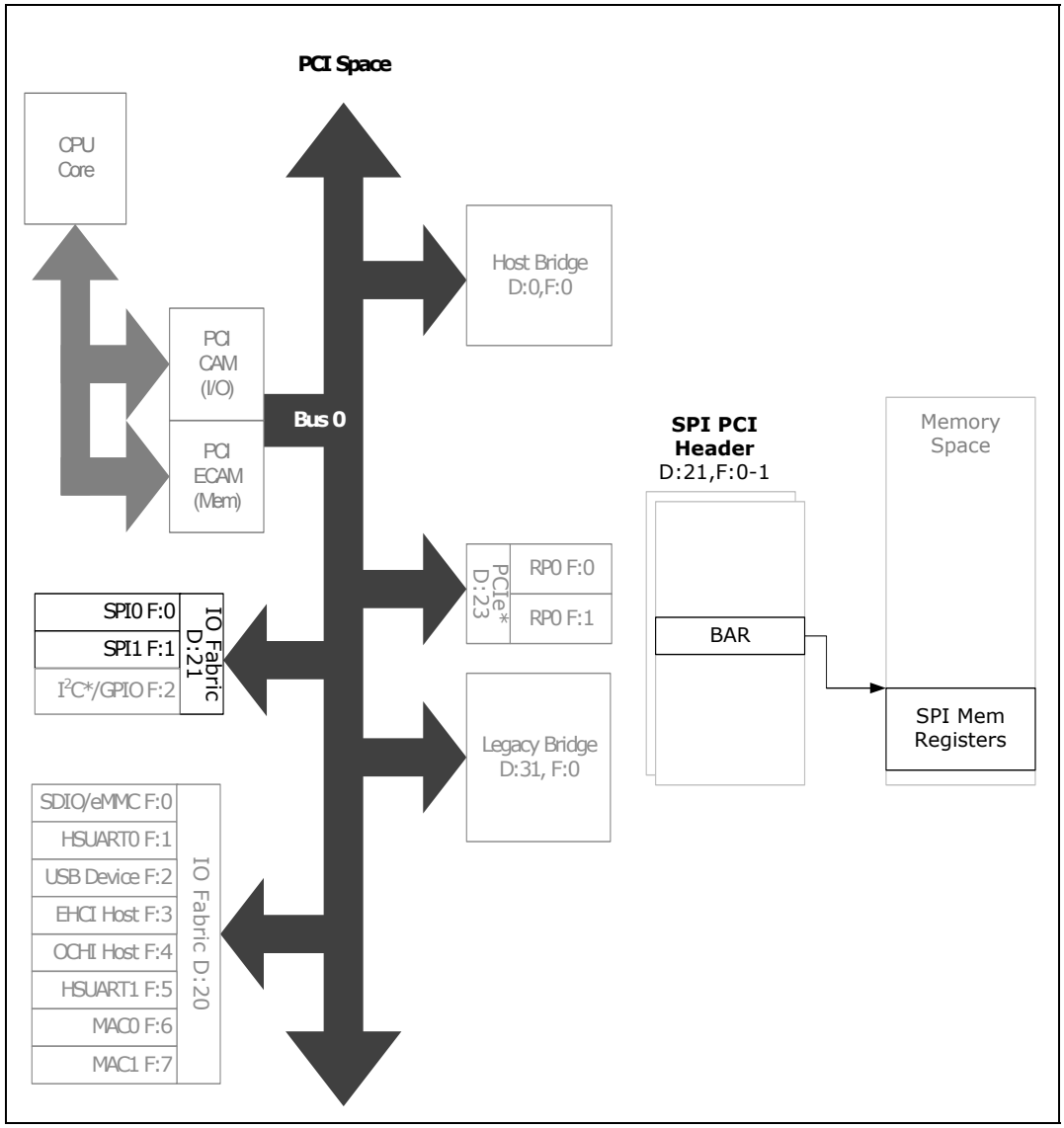
Table 109. SPI Clock Frequency Settings (Sheet 2 of 2)

DDS_CLK_RATE[23:0] (hex)	SCLK_INT Frequency	SCR[7:0] (dec)	SCLK Frequency (MHz)	SCLK Frequency (KHz)	Duty Cycle (%)
10000	0.781	0		390.625	50
8000	0.391	0		195.313	50
8000	0.391	194		1.002	50

20.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 44. SPI Register Map



20.4 PCI Configuration Registers

Registers listed are for Function 0 (SPI Controller 0). Function 1 (SPI Controller 1) contains the same registers. Differences between SPI Controllers Ports will be noted in individual registers.

Table 110. Summary of PCI Configuration Registers—0/21/0

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"Vendor ID (VENDOR_ID)—Offset 0h" on page 778	8086h
2h	3h	"Device ID (DEVICE_ID)—Offset 2h" on page 779	0935h
4h	5h	"Command Register (COMMAND_REGISTER)—Offset 4h" on page 779	0000h
6h	7h	"Status Register (STATUS)—Offset 6h" on page 780	0010h
8h	Bh	"Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h" on page 781	0C800010h
Ch	Ch	"Cache Line Size (CACHE_LINE_SIZE)—Offset Ch" on page 781	00h
Dh	Dh	"Latency Timer (LATENCY_TIMER)—Offset Dh" on page 782	00h
Eh	Eh	"Header Type (HEADER_TYPE)—Offset Eh" on page 782	80h
Fh	Fh	"BIST (BIST)—Offset Fh" on page 782	00h
10h	13h	"Base Address Register (BAR0)—Offset 10h" on page 783	00000000h
28h	2Bh	"Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h" on page 784	00000000h
2Ch	2Dh	"Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch" on page 784	0000h
2Eh	2Fh	"Subsystem ID (SUB_SYS_ID)—Offset 2Eh" on page 784	0000h
30h	33h	"Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h" on page 785	00000000h
34h	37h	"Capabilities Pointer (CAP_POINTER)—Offset 34h" on page 785	00000080h
3Ch	3Ch	"Interrupt Line Register (INTR_LINE)—Offset 3Ch" on page 785	00h
3Dh	3Dh	"Interrupt Pin Register (INTR_PIN)—Offset 3Dh" on page 786	00h
3Eh	3Eh	"MIN_GNT (MIN_GNT)—Offset 3Eh" on page 786	00h
3Fh	3Fh	"MAX_LAT (MAX_LAT)—Offset 3Fh" on page 787	00h
80h	80h	"Capability ID (PM_CAP_ID)—Offset 80h" on page 787	01h
81h	81h	"Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h" on page 787	A0h
82h	83h	"Power Management Capabilities (PMC)—Offset 82h" on page 788	4803h
84h	85h	"Power Management Control/Status Register (PMCSR)—Offset 84h" on page 788	0008h
86h	86h	"PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h" on page 789	00h
87h	87h	"Power Management Data Register (DATA_REGISTER)—Offset 87h" on page 789	00h
A0h	A0h	"Capability ID (MSI_CAP_ID)—Offset A0h" on page 790	05h
A1h	A1h	"Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h" on page 790	00h
A2h	A3h	"Message Control (MESSAGE_CTRL)—Offset A2h" on page 790	0100h
A4h	A7h	"Message Address (MESSAGE_ADDR)—Offset A4h" on page 791	00000000h
A8h	A9h	"Message Data (MESSAGE_DATA)—Offset A8h" on page 791	0000h
ACh	AFh	"Mask Bits for MSI (PER_VEC_MASK)—Offset ACh" on page 792	00000000h
B0h	B3h	"Pending Bits for MSI (PER_VEC_PEND)—Offset B0h" on page 792	00000000h

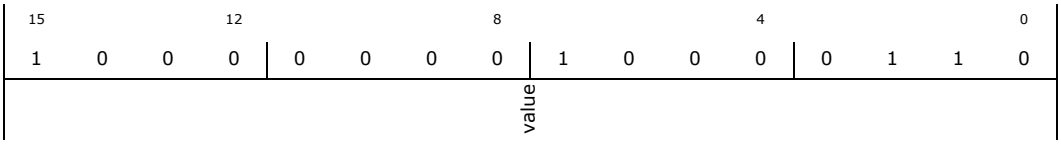
20.4.1 Vendor ID (VENDOR_ID)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

VENDOR_ID: [B:0, D:21, F:0] + 0h

Default: 8086h



Bit Range	Default & Access	Description
15: 0	8086h RO	**Vendor ID (value):** PCI Vendor ID for Intel

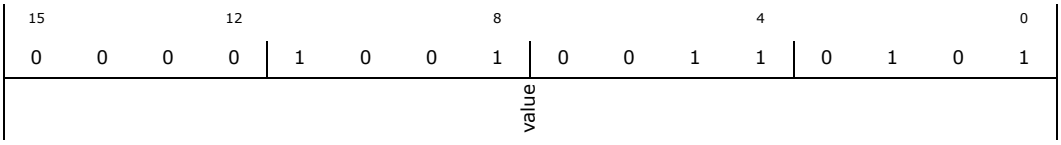
20.4.2 Device ID (DEVICE_ID)—Offset 2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

DEVICE_ID: [B:0, D:21, F:0] + 2h

Default: 0935h



Bit Range	Default & Access	Description
15: 0	0935h RO	**Device ID (value):** PCI Device ID

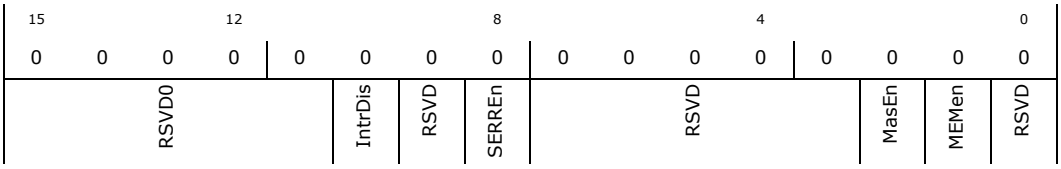
20.4.3 Command Register (COMMAND_REGISTER)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

COMMAND_REGISTER: [B:0, D:21, F:0] + 4h

Default: 0000h



Bit Range	Default & Access	Description
15: 11	0h RO	**RSVD0 (RSVD0):** Reserved
10	0b RW	**Interrupt Disable (IntrDis):** Interrupt disable. Disables generation of interrupt messages in the PCI Express function. 1 =) disabled, 0 =) not disabled

Bit Range	Default & Access	Description
9	0h RO	**Reserved (RSVD):** Reserved.
8	0b RW	**SERR Enable (SERREn):** When set, this bit enables the non-fatal and fatal errors detected by the function to be reported to the root complex.
7: 3	00h RO	**Reserved (RSVD):** Reserved.
2	0b RW	**Bus Master Enable (MasEn):** 0=)disables upstream requests 1=)enables upstream requests.
1	0b RW	**Memory Space Enable (MEMen):** Device support for Memory transactions. 0 =) not supported. 1 =) supported.
0	0h RO	**Reserved (RSVD):** Reserved.

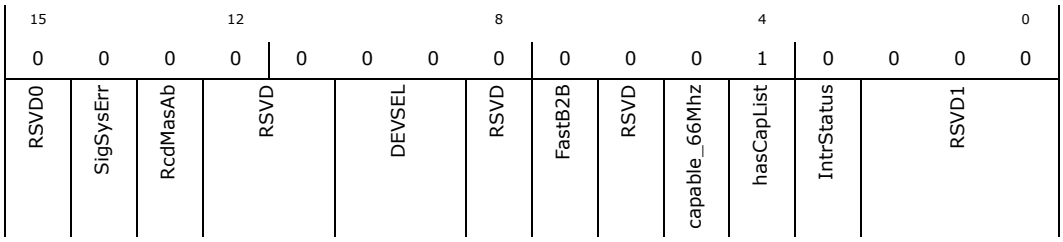
20.4.4 Status Register (STATUS)—Offset 6h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

STATUS: [B:0, D:21, F:0] + 6h

Default: 0010h



Bit Range	Default & Access	Description
15	0h RO	**RSVD0 (RSVD0):** Reserved
14	0b RW	**Signaled System Error (SigSysErr):** Set when a function detects a system error and the SERR Enable bit is set
13	0b RW	**Received master abort (RcdMasAb):** Set when requester receives a completion with Unsupported Request completion status
12: 11	0h RO	**Reserved (RSVD):** Reserved.
10: 9	0b RO	**DEVSEL Timing (DEVSEL):** Deprecated: Hardwired to 0
8	0h RO	**Reserved (RSVD):** Reserved.
7	0b RO	**Fast Back-to-Back Capable (FastB2B):** Deprecated: Hardwired to 0
6	0h RO	**Reserved (RSVD):** Reserved.
5	0b RO	**66MHz-Capable (capable_66Mhz):** Deprecated: Hardwired to 0

Bit Range	Default & Access	Description
4	1h RO	**Capabilities List (hasCapList):** Indicates the presence of one or more capability register sets.
3	0b RO	**Interrupt Status (IntrStatus):** Indicates that the function has a legacy interrupt request outstanding. This bit has no meaning if Message Signaled Interrupts are being used
2: 0	0h RO	**RSVD1 (RSVD1):** Reserved

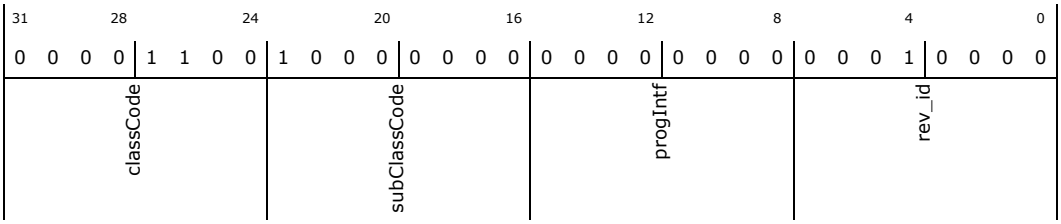
20.4.5 Revision ID and Class Code (REV_ID_CLASS_CODE)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

REV_ID_CLASS_CODE: [B:0, D:21, F:0] + 8h

Default: 0C800010h



Bit Range	Default & Access	Description
31: 24	0Ch RO	**Class Code (classCode):** Broadly classifies the type of function that the device performs.
23: 16	80h RO	**Sub-Class Code (subClassCode):** Identifies more specifically (than the class_code byte) the function of the device.
15: 8	00h RO	**Programming Interface (progIntf):** Used to define the register set variation within a particular sub-class.
7: 0	10h RO	**Revision ID (rev_id):** Assigned by the function manufacturer and identifies the revision number of the function.

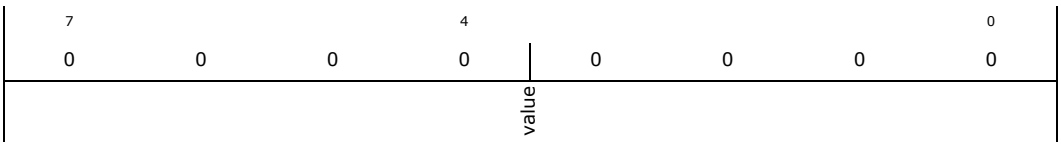
20.4.6 Cache Line Size (CACHE_LINE_SIZE)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

CACHE_LINE_SIZE: [B:0, D:21, F:0] + Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Cache Line Size (value):** Implemented as a R/W register for legacy purposes but has no effect on device functionality.

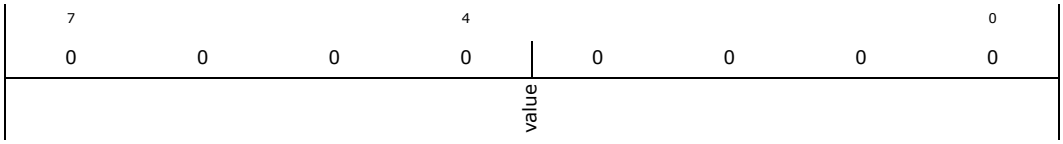
20.4.7 Latency Timer (LATENCY_TIMER)—Offset Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

LATENCY_TIMER: [B:0, D:21, F:0] + Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Latency Timer (value):** Deprecated. Hardwire to 0.

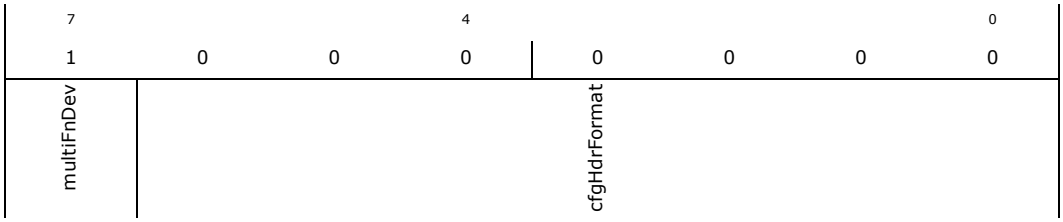
20.4.8 Header Type (HEADER_TYPE)—Offset Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

HEADER_TYPE: [B:0, D:21, F:0] + Eh

Default: 80h



Bit Range	Default & Access	Description
7	1h RO	**Multi-Function Device (multiFnDev):** Hard-wired to 1 to indicate that this is a multi-function device
6: 0	0h RO	**Configuration Header Format (cfgHdrFormat):** Hard-wired to 0 to indicate that this configuration header is a Type 0 header, i.e. it is an endpoint rather than a bridge.

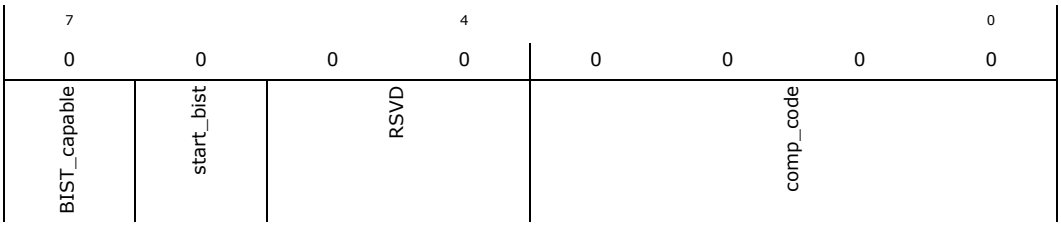
20.4.9 BIST (BIST)—Offset Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

BIST: [B:0, D:21, F:0] + Fh

Default: 00h



Bit Range	Default & Access	Description
7	0h RO	**BIST_capable (BIST_capable):** Hard-wired to 0. (Returns 1 if the function implements a BIST)
6	0h RO	**Start (start_bist):** Set to start the functions BIST if BIST is supported.
5: 4	0h RO	**Reserved (RSVD):** Reserved.
3: 0	0h RO	**Completion Code (comp_code):** Completion code having run BIST if BIST is supported. 0=)success. non-zero=)failure

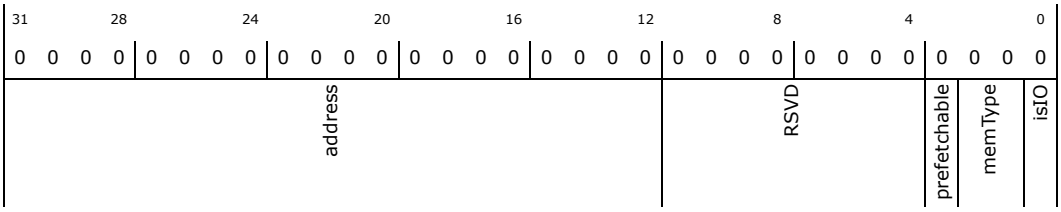
20.4.10 Base Address Register (BAR0)—Offset 10h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BAR0: [B:0, D:21, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31: 12	0h RW	**address (address):** Used to determine the size of memory required by the device and to assign a start address for this required amount of memory.
11: 4	00h RO	**Reserved (RSVD):** Reserved.
3	0b RO	**Prefetchable (prefetchable):** Defines the block of memory as prefetchable or not. A block of memory is prefetchable if it fulfils the following 3 conditions (1) no side effects on reads, (2) the device returns all bytes on reads regardless of the byte enables, and (3) host bridges can merge processor writes into this range without causing errors. Hardwired to 0
2: 1	00b RO	**Type (memType):** Hardwired to 0 to indicate a 32-bit decoder
0	0b RO	**Memory Space Indicator (isIO):** Hardwired to 0 to indicate the register is a memory address decoder

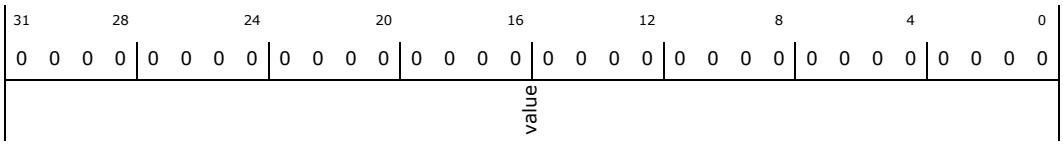
20.4.11 Cardbus CIS Pointer (CARDBUS_CIS_POINTER)—Offset 28h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CARDBUS_CIS_POINTER: [B:0, D:21, F:0] + 28h

Default: 00000000h



Bit Range	Default & Access	Description
31: 0	0h RO	**Cardbus CIS Pointer (value):** Reserved. Hardwire to 0.

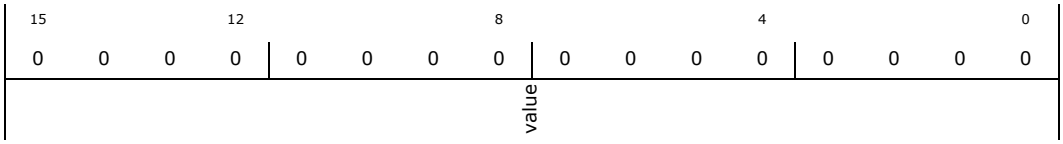
20.4.12 Subsystem Vendor ID (SUB_SYS_VENDOR_ID)—Offset 2Ch

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_VENDOR_ID: [B:0, D:21, F:0] + 2Ch

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem Vendor ID (value):** PCI Subsystem Vendor ID

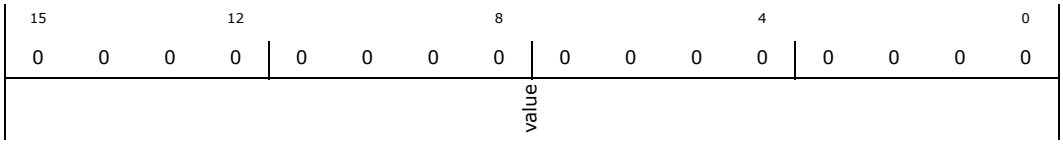
20.4.13 Subsystem ID (SUB_SYS_ID)—Offset 2Eh

Access Method

Type: PCI Configuration Register (Size: 16 bits)

SUB_SYS_ID: [B:0, D:21, F:0] + 2Eh

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RO	**Subsystem ID (value):** PCI Subsystem ID

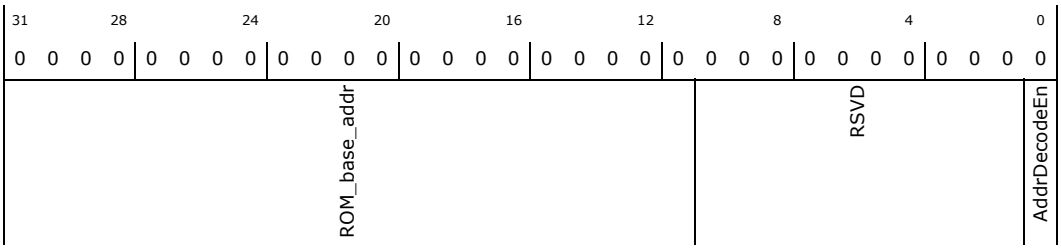
20.4.14 Expansion ROM Base Address (EXP_ROM_BASE_ADR)—Offset 30h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

EXP_ROM_BASE_ADR: [B:0, D:21, F:0] + 30h

Default: 00000000h



Bit Range	Default & Access	Description
31: 11	0h RW	**ROM Start Address (ROM_base_addr):** Used to determine the size of memory required by the ROM and to assign a start address for this required amount of memory.
10: 1	000h RO	**Reserved (RSVD):** Reserved.
0	0h RW	**Address Decode Enable (AddrDecodeEn):** A 1 in this field enables the function's ROM address decoder assuming that the Memory Space bit in the Command Register is also set to 1

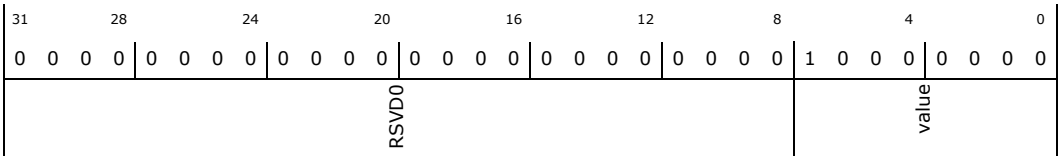
20.4.15 Capabilities Pointer (CAP_POINTER)—Offset 34h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

CAP_POINTER: [B:0, D:21, F:0] + 34h

Default: 00000080h



Bit Range	Default & Access	Description
31: 8	0h RO	**RSVD0 (RSVD0):** Reserved
7: 0	80h RO	**Capabilities Pointer (value):** Pointer to memory location of first entry of linked list of configuration register sets each of which supports a feature. Points to PM (power management) register set at location 0x80

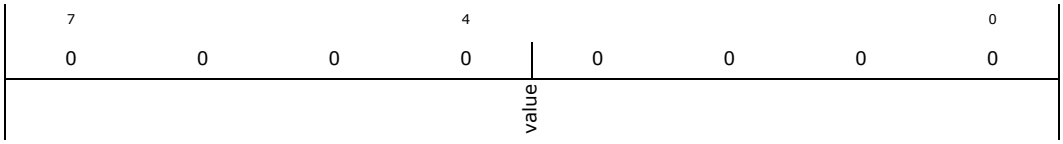
20.4.16 Interrupt Line Register (INTR_LINE)—Offset 3Ch

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_LINE: [B:0, D:21, F:0] + 3Ch

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Interrupt Line Register (value):** The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information.

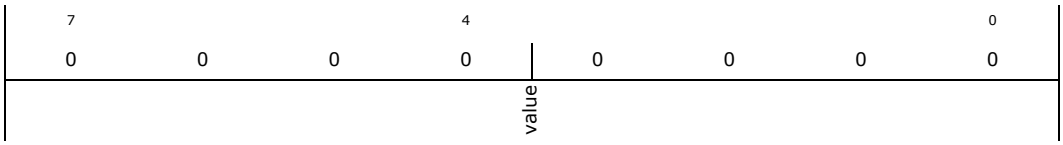
20.4.17 Interrupt Pin Register (INTR_PIN)—Offset 3Dh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

INTR_PIN: [B:0, D:21, F:0] + 3Dh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Interrupt Pin Register (value):** The Interrupt Pin register tells which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. Devices (or device functions) that do not use an interrupt pin must put a 0 in this register. The values 05h through FFh are reserved. For this system function 0 is connected to INTA, 1 to INTB, 2 to INTC 3 to INTD, 4 to INTA, 5 to INTB etc.

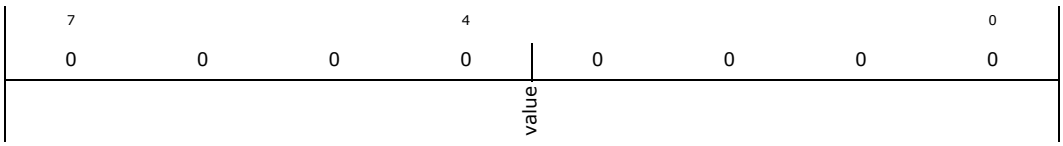
20.4.18 MIN_GNT (MIN_GNT)—Offset 3Eh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MIN_GNT: [B:0, D:21, F:0] + 3Eh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MIN_GNT (value):** Hardwired to 0

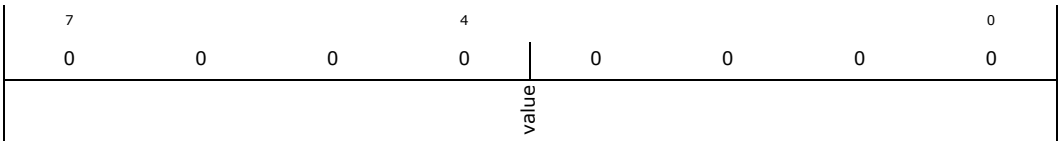
20.4.19 MAX_LAT (MAX_LAT)—Offset 3Fh

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MAX_LAT: [B:0, D:21, F:0] + 3Fh

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**MAX_LAT (value):** Hardwired to 0

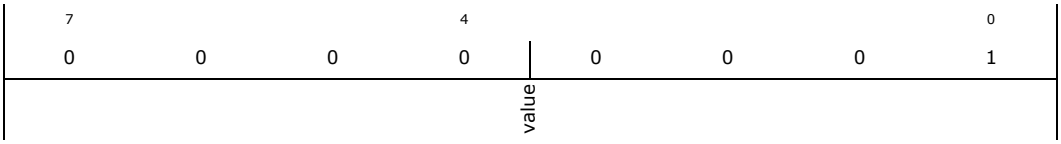
20.4.20 Capability ID (PM_CAP_ID)—Offset 80h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_CAP_ID: [B:0, D:21, F:0] + 80h

Default: 01h



Bit Range	Default & Access	Description
7: 0	01h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

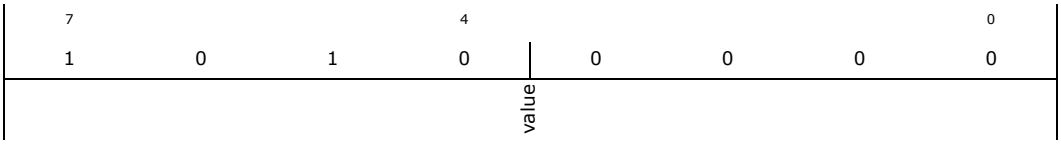
20.4.21 Next Capability Pointer (PM_NXT_CAP_PTR)—Offset 81h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PM_NXT_CAP_PTR: [B:0, D:21, F:0] + 81h

Default: A0h



Bit Range	Default & Access	Description
7: 0	a0h RO	**Next Capability Pointer (value):** Pointer to the next register set of feature specific configuration registers. Hardwired to 0xA0 to point to the MSI Capability Structure

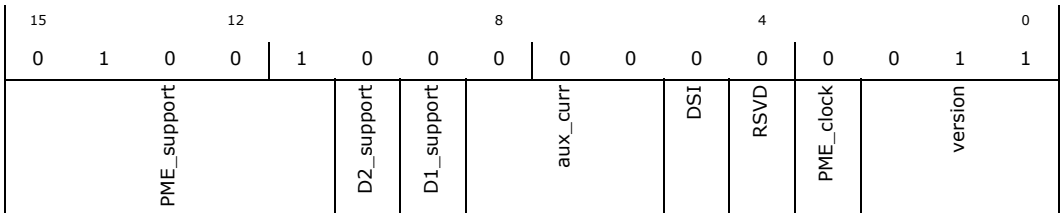
20.4.22 Power Management Capabilities (PMC)—Offset 82h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMC: [B:0, D:21, F:0] + 82h

Default: 4803h



Bit Range	Default & Access	Description
15: 11	09h RO	**PME Support (PME_support):** PME_Support field Indicates the PM states within which the function is capable of sending a PME (Power Management Event) message. 0 in a bit =) PME is not supported in the corresponding PM state, where bit indexes 11,12,13,14,15 correspond to PM states D0, D1, D2, D3hot, D3cold respectively.
10	0h RO	**D2 Support (D2_support):** Hardwired to 0 as the D2 state is not supported
9	0h RO	**D1 Support (D1_support):** Hardwired to 0 as the D1 state is not supported
8: 6	0h RO	**Aux Current (aux_curr):** Hardwired to 0 as the D3hot state is not supported
5	0h RO	**Device Specific Initialisation (DSI):** Hardwired to 0 to indicate that the device does not require a device specific initialisation sequence following transition to the D0 uninitialised state
4	0h RO	**Reserved (RSVD):** Reserved.
3	0h RO	**PME Clock (PME_clock):** Deprecated. Hardwired to 0
2: 0	011b RO	**Version (version):** This function complies with revision 1.2 of the PCI Power Management Interface Specification

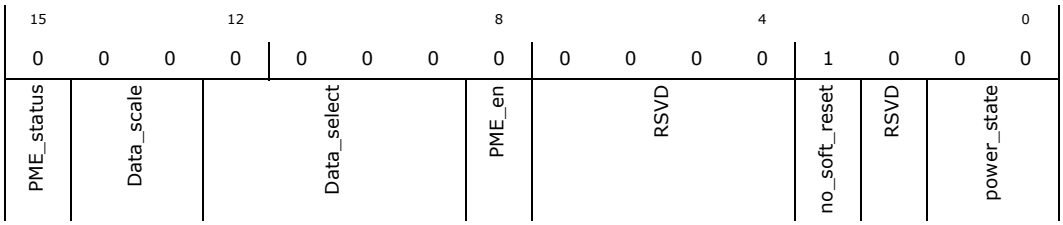
20.4.23 Power Management Control/Status Register (PMCSR)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

PMCSR: [B:0, D:21, F:0] + 84h

Default: 0008h



Bit Range	Default & Access	Description
15	0h RW	**PME Status (PME_status):** Set if function has experienced a PME (even if PME_en (bit 8 of PMCSR register) is not set).
14: 13	0h RO	**Data Scale (Data_scale):** Hardwired to 0 as the data register is not supported
12: 9	0h RO	**Data Select (Data_select):** Hardwired to 0 as the data register is not supported
8	0b RW	**PME Enable (PME_en):** Enable device function to send PME messages when an event occurs. 1=)enabled. 0=)disabled
7: 4	0h RO	**Reserved (RSVD):** Reserved.
3	1b RO	**No Soft Reset (no_soft_reset):** Devices do perform an internal reset when transitioning from D3hot to D0
2	0h RO	**Reserved (RSVD):** Reserved.
1: 0	00b RW	**Power State (power_state):** Allows software to read current PM state or transition device to a new PM state, where 2'b00 = D0, 2'b01=D1, 2'b10=D2, 2'b11=D3hot

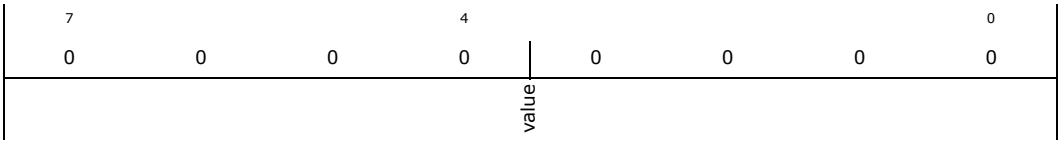
20.4.24 PM CSR PCI-to-PCI Bridge Support Extension (PMCSR_BSE)—Offset 86h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

PMCSR_BSE: [B:0, D:21, F:0] + 86h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**PM CSR PCI-to-PCI Bridge Support Extension (value):** Not Supported. Hardwired to 0.

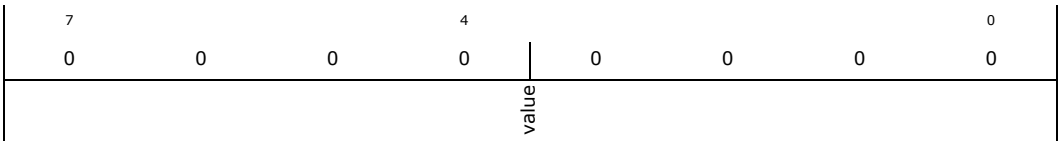
20.4.25 Power Management Data Register (DATA_REGISTER)—Offset 87h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

DATA_REGISTER: [B:0, D:21, F:0] + 87h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RO	**Power Management Data Register (value):** Not Supported. Hardwired to 0

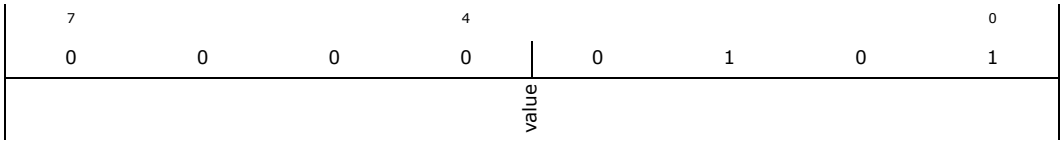
20.4.26 Capability ID (MSI_CAP_ID)—Offset A0h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_CAP_ID: [B:0, D:21, F:0] + A0h

Default: 05h



Bit Range	Default & Access	Description
7: 0	05h RO	**Capability ID (value):** Identifies the feature associated with this register set. Hardwired value as per PCI SIG assigned capability ID

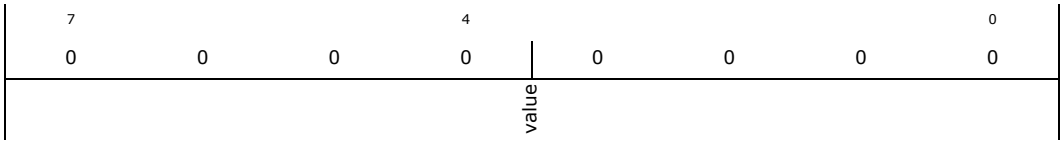
20.4.27 Next Capability Pointer (MSI_NXT_CAP_PTR)—Offset A1h

Access Method

Type: PCI Configuration Register (Size: 8 bits)

MSI_NXT_CAP_PTR: [B:0, D:21, F:0] + A1h

Default: 00h



Bit Range	Default & Access	Description
7: 0	00h RO	**Next Capability Pointer (value):** Hardwired to 0 as this is the last capability structure in the chain

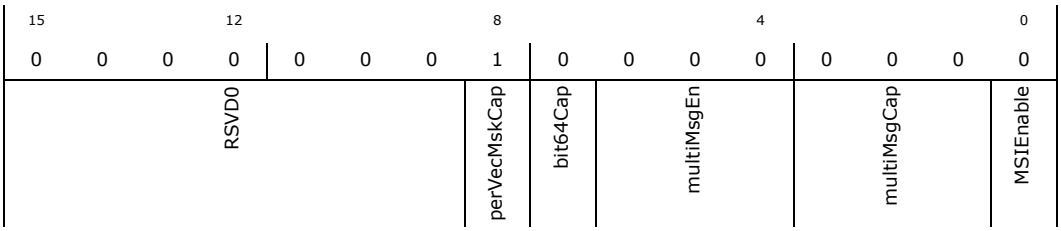
20.4.28 Message Control (MESSAGE_CTRL)—Offset A2h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_CTRL: [B:0, D:21, F:0] + A2h

Default: 0100h



Bit Range	Default & Access	Description
15: 9	0h RO	**RSVD0 (RSVD0):** Reserved
8	1h RO	**Per Vector Masking Capable (perVecMskCap):** Hardwired to 1 to indicate the function supports PVM
7	0h RO	**64 bit Address Capabale (bit64Cap):** This bit is hardwired to 0 to indicate that the function is not capable of sending a 64-bit message address.
6: 4	0h RW	**Multi-Message Enable (multiMsgEn):** As only one vector is supported per function, software should only write a value of 0x0 to this field
3: 1	0h RO	**Multiple Message Enable (multiMsgCap):** This field is hardwired to 0x0 to indicate that the function is requesting a single vector
0	0h RW	**MSI Enable (MSIEnable):** Set to enable MSI to request service. If set then it's prohibited to use the INTx pin. System configuration software sets this bit to enable MSI.

20.4.29 Message Address (MESSAGE_ADDR)—Offset A4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

MESSAGE_ADDR: [B:0, D:21, F:0] + A4h

Default: 00000000h

Bit Range	Default & Access	Description
31: 2	0h RW	**Message Address (address):** If the Message Enable bit (bit 0 of the Message Control register) is set, the contents of this register specify the DWORD-aligned address (AD[31:2]) for the MSI memory write transaction. AD[1:0] are driven to zero during the address phase. This field is read/write
1: 0	0h RO	**RSVD0 (RSVD0):** Reserved

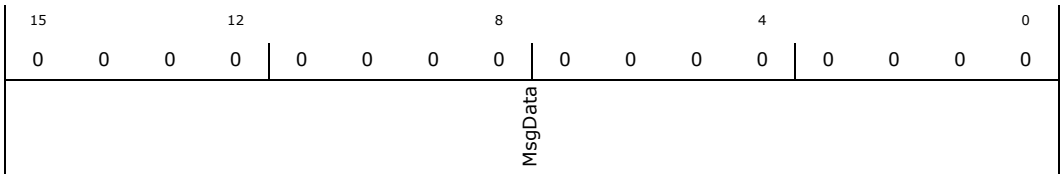
20.4.30 Message Data (MESSAGE_DATA)—Offset A8h

Access Method

Type: PCI Configuration Register (Size: 16 bits)

MESSAGE_DATA: [B:0, D:21, F:0] + A8h

Default: 0000h



Bit Range	Default & Access	Description
15: 0	0h RW	**Data Field (MsgData):** System-specified message data. If the Message Enable bit (bit 0 of the Message Control register) is set, the message data is driven onto the lower word (AD[15:0]) of the memory write transactions data phase. AD[31:16] are driven to zero during the memory write transactions data phase. C/BE[3::0]# are asserted during the data phase of the memory write transaction. None of the message bits will be changed by hardware

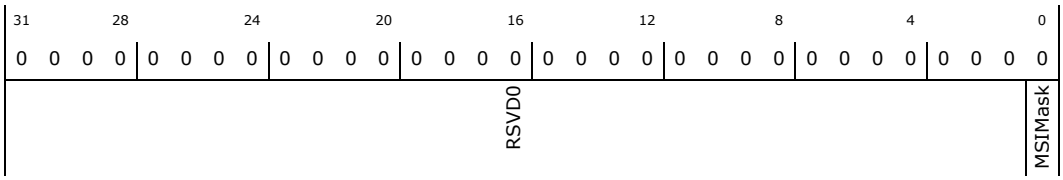
20.4.31 Mask Bits for MSI (PER_VEC_MASK)—Offset ACh

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_MASK: [B:0, D:21, F:0] + ACh

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RW	**Vector 0 Mask (MSIMask):** Mask Bit for Vector 0. If this bit is set, the function will not send MSI messages

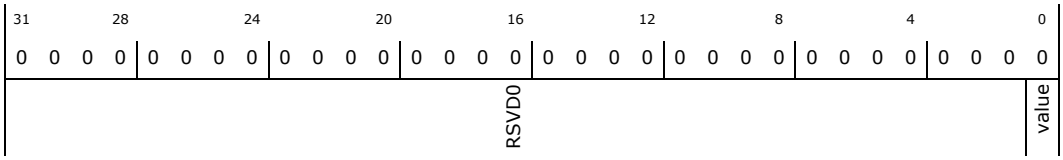
20.4.32 Pending Bits for MSI (PER_VEC_PEND)—Offset B0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PER_VEC_PEND: [B:0, D:21, F:0] + B0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 1	0h RO	**RSVD0 (RSVD0):** Reserved
0	0h RO	**Vector 0 Pending (value):** Pending Bit for Vector 0.

20.5 Memory Mapped Registers

Table 111. Summary of Memory Mapped I/O Registers—BAR0

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"SPI Control Register 0 (SSCR0)—Offset 0h" on page 793	00000000h
4h	7h	"SPI Control Register 1 (SSCR1)—Offset 4h" on page 794	00000000h
8h	Bh	"SPI Status Register (SSSR)—Offset 8h" on page 796	0003E004h
10h	13h	"SPI Data Register (SSDR)—Offset 10h" on page 797	00000000h
28h	2Bh	"DDS Clock Rate Register (DDS_RATE)—Offset 28h" on page 798	00028F5Ch

20.5.1 SPI Control Register 0 (SSCR0)—Offset 0h

The SPI control register 0 controls various SPI functions.

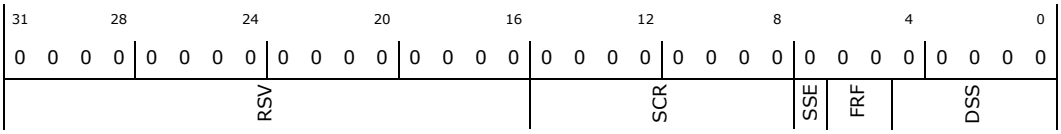
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SSCR0: [BAR0] + 0h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31:16	0b RO	**RSV:** Reserved.
15:8	0b RW	**Serial Clock Rate (SCR):** This bitfield is used to select the baud, or bit rate, of the SPI. A total of 256 different bit rates can be selected to further divide the frequency of the clock obtained by dividing the input system clock according to the DDS_RATE register, see DDS_RATE Register description. The resultant clock is driven on the SCLK pin and is used by the SPIs transmit logic to drive data on the MOSI pin, and to latch data on the MISO pin. The SCLK frequency is given by DDS_FREQ /(2 x (SCR + 1)) where SCR is a decimal integer in the range 0 to 255 and DDS_FREQ is determined by the value programmed in the DDS_RATE Register.

Bit Range	Default & Access	Description
7	0b RW	**Synchronous Serial Port Enable (SSE):** The SPI enable bit is used to enable and disable all SPI operations. When the SPI is disabled, all of its clocks are powered down to minimize power consumption. Note that the SSE is the only control bit within the SPI which is reset to a known state. It is cleared to zero to ensure the SPI is disabled following a reset. When the SSE bit is cleared during active operation, the SPI is disabled immediately, causing the current frame being transmitted to be terminated. Clearing SSE resets the SPIs FIFOs and the SPI status bits. However, the SPIs control registers and the Receive FIFO Overrun (ROR) status bit are not reset. Note: After reset or after clearing the SSE, the user must ensure the SSCR1 and SSSR registers are properly reconfigured or reset before re-enabling the SPI with the SSE; other control bits in SSCR0 may be written at the same time as the SSE. 0 : SPI operation disabled 1 : SPI operation enabled
6:5	0b RW	**Frame Format (FRF):** 00 : Motorola* Serial Peripheral Interface (SPI) 01 : Reserved, undefined operation 10 : Reserved, undefined operation 11 : Reserved, undefined operation
4:0	0b RW	**Data Size Select (DSS):** The 5-bit data size select field is used to select the size of the data transmitted and received by the SPI. Data can be 4 to 32 bits in length. When data is programmed to be less than 16 bits, received data is automatically right justified and the upper bits in the receive FIFO are zero-filled by receive logic. Transmit data should not be left justified by the user before being placed in the transmit FIFO; transmit logic in the SPI will automatically left justify the data sample according to the value of DSS before the sample is transmitted on MOSI. Although it is possible to program data sizes of 1, 2, and 3 bits, these sizes are reserved and produce unpredictable results in the SPI. 00000 : Reserved, undefined operation 00001 : Reserved, undefined operation 00010 : Reserved, undefined operation 00011 : 4-bit data 00100 : 5-bit data 00101 : 6-bit data 00110 : 7-bit data 00111 : 8-bit data 01000 : 9-bit data 01001 : 10-bit data 01010 : 11-bit data 01011 : 12-bit data 01100 : 13-bit data 01101 : 14-bit data 01110 : 15-bit data 01111 : 16-bit data 10000 : 17-bit data 10001 : 18-bit data 10010 : 19-bit data 10011 : 20-bit data 10100 : 21-bit data 10101 : 22-bit data 10110 : 23-bit data 10111 : 24-bit data 11000 : 25-bit data 11001 : 26-bit data 11010 : 27-bit data 11011 : 28-bit data 11100 : 29-bit data 11101 : 30-bit data 11110 : 31-bit data 11111 : 32-bit data

20.5.2 SPI Control Register 1 (SSCR1)—Offset 4h

The SPI Control Register 1 controls various SPI functions.

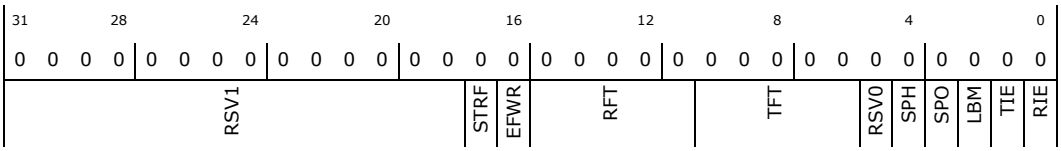
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SSCR1: [BAR0] + 4h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31:18	00h RO	**RSV1:** Reserved.
17	0b RW	**Select FIFO for Enable FIFO Write/Read (STRF):** This bit selects whether the Transmit or Receive FIFO is enabled for writes and reads whenever the EFWR is programmed to one, which puts the SPI in a special functional mode. 0 : Transmit FIFO is selected for both writes and reads through the SSDR. 1 : Receive FIFO is selected for both writes and reads through the SSDR
16	0b RW	**Enable FIFO Write/Read Function (EFWR):** This bit enables a special functional mode for the SPI. 0 : the SPI operates in normal mode. 1 : the SPI enters a mode in which whenever the CPU reads or writes to the SPI Data register it actually reads and writes exclusively to either the Transmit FIFO or the Receive FIFO depending on the programmed state of the Select FIFO for EFWR (STRF) bit. In this special mode, data will not be transmitted on the MOSI pin and data input on the MISO pin will not be stored. This mode can be used to test, through software, whether or not the Transmit FIFO or the Receive FIFO operates properly as a first-in-first-out memory stack.
15:11	0b RW	**Receive FIFO Interrupt Threshold (RFT):** The receive FIFO interrupt threshold sets the threshold at or above which the FIFO controller triggers, if enabled, a CPU interrupt request. This level should be set to the desired threshold value minus 1.
10:6	0b RW	**Transmit FIFO Interrupt Threshold (TFT):** The transmit FIFO interrupt threshold sets the threshold at or below which the FIFO controller triggers, if enabled, a CPU interrupt request. This level should be set to the desired threshold value minus 1.
5	0b RO	**RSV0:** Reserved.
4	0b RW	**Serial Clock Phase (SPH):** The serial clock (SCLK) phase bit determines the phase relationship between the SCLK and the Chip Select (CS) pins. 0 : SCLK remains in its inactive/idle state (as determined by the SPO setting) for one full cycle after CS is asserted low at the beginning of a frame. SCLK continues to transition for the rest of the frame and is then held in its inactive state for one-half of an SCLK period before CS is deasserted high at the end of the frame. 1 : SCLK remains in its inactive/ idle state (as determined by the SPO setting) for half a cycle after CS is asserted low at the beginning of a frame. SCLK continues to transition for the rest of the frame and is then held in its inactive state for one full SCLK period before CS is de-asserted high at the end of the frame. The combination of the SPO and SPH settings determines when SCLK is active during the assertion of CS and which SCLK edge is used to transmit and receive data on the MOSI and MISO pins. When SPO and SPH are programmed to the same value (both 0 or 1), transmit data is driven on the falling edge of SCLK and receive data is latched on the rising edge of SCLK. When SPO and SPH are programmed to opposite values (one 0 and the other 1), transmit date is driven on the rising edge of SCLK and receive data is latched on the falling edge of SCLK.

Bit Range	Default & Access	Description
3	0b RW	**Serial Clock Polarity (SPO):** The serial clock (SCLK) polarity bit selects the polarity of the inactive state of the SCLK pin. 0 : the SCLK is held low in the inactive or idle state when the SPI is not transmitting/ receiving data. 1 : the SCLK is held high during the inactive/idle state. The programmed setting of the SPO alone does not determine which SCLK edge is used to transmit or receive data. The SPO setting in combination with SPH determines this.
2	0b RW	**Loop Back Mode (LBM):** The loop back mode bit is used to enable and disable the ability of the SPI transmit and receive logic to communicate. 0 : the SPI operates normally. The transmit and receive data paths are independent and communicate via their respective pins. 1 : the output of the transmit serial shifter is directly connected to the input of the receive serial shifter internally.
1	0b RW	**Transmit FIFO Interrupt Enable (TIE):** The Transmit FIFO Interrupt Enable bit is used to mask or enable the transmit FIFO service request interrupt. 0 : the interrupt is masked and the state of TFS within the SPI Status Register is ignored by the interrupt controller. 1 : the interrupt is enabled, and whenever TFS is set to one, the interrupt request is made to the interrupt controller. Note that programming TIE=0 does not affect the current state of TFS or the transmit FIFO logics ability to set and clear TFS, it only blocks the generation of the interrupt request.
0	0b RW	**Receive FIFO Interrupt Enable (RIE):** The Receive FIFO Interrupt Enable bit is used to mask or enable the Receive FIFO service request interrupt. 0 : the interrupt is masked, and the state of RFS within the SPI Status Register is ignored by the interrupt controller. 1 : the interrupt is enabled, and whenever RFS is set to one, the interrupt request is made to the interrupt controller. Note that programming RIE=0 does not affect the current state of RFS or the receive FIFO logics ability to set and clear RFS, it only blocks the generation of the interrupt request.

20.5.3 SPI Status Register (SSSR)—Offset 8h

The SPI status register contains bits which signal overrun errors as well as transmit and receive FIFO service requests. Each of these hardware-detected events signals an interrupt request to the interrupt controller. The status register also contains flags that indicate when the SPI is actively transmitting characters, when the transmit FIFO is not full, and when the receive FIFO is not empty (no interrupt generated). Bits which cause an interrupt will signal the request as long as the bit is set. Once the bit is cleared, the interrupt is cleared. Read/write bits are called status bits; read-only bits are called flags. Status bits are referred to as sticky (once set by hardware, must be cleared by software). Writing a one to a sticky status bit clears it, writing a zero has no effect. Read-only flags are set and cleared by hardware; writes have no effect. Additionally, some bits which cause interrupts have corresponding mask bits in the control registers.

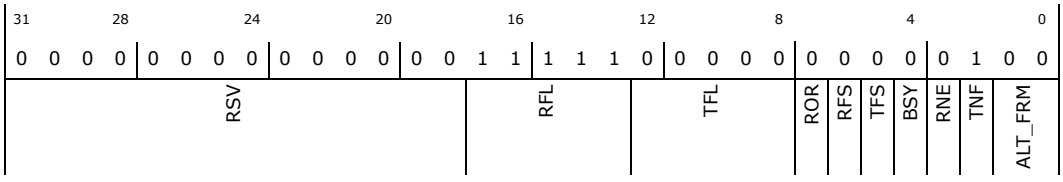
Access Method

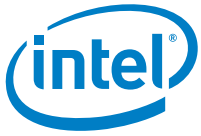
Type: Memory Mapped I/O Register (Size: 32 bits)

SSSR: [BAR0] + 8h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:0] + 10h

Default: 0003E004h





Bit Range	Default & Access	Description
31:18	00h RO	**RSV:** Reserved.
17:13	1Fh RO	**Receive FIFO Level (RFL):** This 5-bit value shows how many valid entries are currently in the Receive FIFO (up to 32).
12:8	0b RO	**Transmit FIFO Level (TFL):** This 5-bit value shows how many valid entries are currently in the Transmit FIFO (up to 32).
7	0b RW/C	**Receiver Overrun Status (ROR):** The receiver overrun status bit is a read/write bit which is set when the receive logic attempts to place data into the receive FIFO after it has been completely filled. Each time a new piece of data is received, the set signal to the ROR bit is asserted, and the newly received data is discarded. This process is repeated for each new piece of data received until at least one empty FIFO entry exists. When the ROR bit is set, an interrupt request is made to the CPU which cannot be locally masked by any SPI register bit. Writing a 1 to this bit resets ROR status and its interrupt request; writing a 0 to this bit does not affect ROR status. Receiver Overrun Status is a non-maskable interrupt
6	0b RO	**Receive FIFO Service Request Flag (RFS):** The receive FIFO service request flag is a read-only bit which is set when the receive FIFO is nearly filled and requires service to prevent an overrun. RFS is set any time the receive FIFO has the same or more entries of valid data than indicated by the Receive FIFO Threshold, and it is cleared when it has fewer entries than the threshold value. When the RFS bit is set, an interrupt request is made unless RIE is cleared. After the CPU reads the FIFO such that it has fewer entries than the RFT value, the RFS flag (and the service request and/or interrupt) is automatically cleared.
5	0b RO	**Transmit FIFO Service Request Flag (TFS):** The Transmit FIFO service request flag is a read-only bit which is set when the transmit FIFO is nearly empty and requires service to prevent an underrun. TFS is set any time the transmit FIFO has the same or fewer entries of valid data than indicated by the Transmit FIFO Threshold, and it is cleared when it has more entries of valid data than the threshold value. When the TFS bit is set, an interrupt request is made unless TIE is cleared. After the CPU fills the FIFO such that it exceeds the threshold, the TFS flag (and the service request and/or interrupt) is automatically cleared.
4	0b RO	**SPI Busy Flag (BSY):** The receive FIFO not empty flag is a read-only bit which is set whenever the receive FIFO contains one or more entries of valid data and is cleared when it no longer contains any valid data. This bit can be polled when using programmed I/O to remove remaining bytes of data from the receive FIFO since CPU interrupt requests are only made when the Receive FIFO Threshold has been met or exceeded. This bit does not request an interrupt.
3	0b RO	**Receive FIFO Not Empty Flag (RNE):** The receive FIFO not empty flag is a read-only bit which is set whenever the receive FIFO contains one or more entries of valid data and is cleared when it no longer contains any valid data. This bit can be polled when using programmed I/O to remove remaining bytes of data from the receive FIFO since interrupt requests are only made when the Receive FIFO Threshold has been met or exceeded. This bit does not request an interrupt.
2	1b RO	**Transmit FIFO Not Full Flag (TNF):** The transmit FIFO not full flag is a read-only bit which is set whenever the transmit FIFO contains one or more entries which do not contain valid data. TNF is cleared when the FIFO is completely full. This bit can be polled when using programmed I/O to fill the transmit FIFO over its threshold level. This bit does not request an interrupt.
1:0	0b RO	**Alternative Frame (ALT_FRM):** This field is not supported and should be treated as reserved.

20.5.4 SPI Data Register (SSDR)—Offset 10h

The SPI Data Register is one 32-bit location that can be accessed by 32-bit data transfers. Transfers can be single word transfer or 2 to 32 -word bursts. The SSDR represents two physical registers: the first is temporary storage for data on its way out through the Transmit FIFO; the other is temporary storage for data coming in through the Receive FIFO. As the register is accessed by the system, FIFO control logic

transfers data automatically between register and FIFO as fast as the system moves it. Data in the FIFO shifts up or down to accommodate the new word (unless it is an attempted write to a full Transmit FIFO). For outbound data transfers (write from system to SPI peripheral), the register may be loaded (written) by the system processor anytime it is below its threshold level. When a data size of less than 32-bits is selected, the user should not left-justify data written to the transmit FIFO. Transmit logic left-justifies the data and ignores any unused bits. Received data less than 32-bits is automatically right justified in the receive buffer.

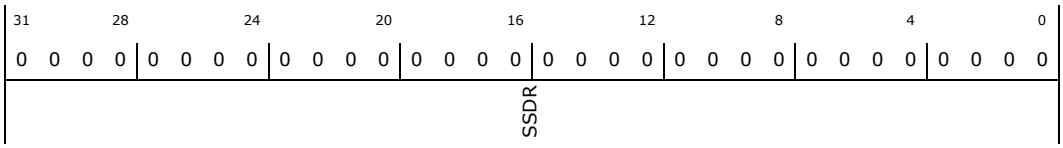
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SSDR: [BAR0] + 10h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:0] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	00h RW	**SPI Data (SSDR):** Data to be written to/read from Transmit/Receive FIFO

20.5.5 DDS Clock Rate Register (DDS_RATE)—Offset 28h

The DDS Clock Rate Register determines the SCLK frequency, along with SCR field of SSCR0 register. This register determines the frequency DDS_FREQ of an internal clock used to clock part of the SPI core logic and to drive the SPI output clock (SCLK). Note that SCLK frequency is not the same as this clock, but is a divided version according to SCR bit in SSCR0 register.

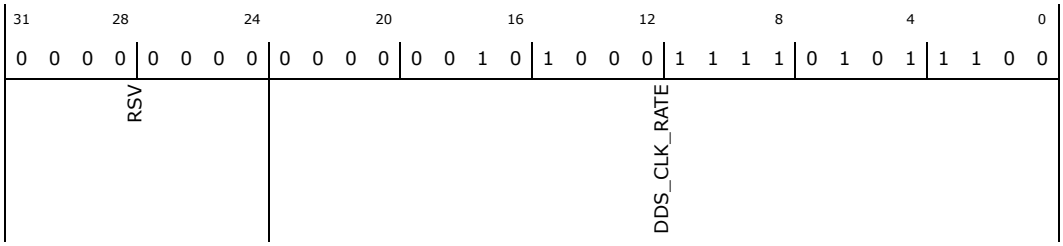
Access Method

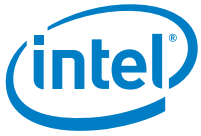
Type: Memory Mapped I/O Register (Size: 32 bits)

DDS_RATE: [BAR0] + 28h

BAR0 Type: PCI Configuration Register (Size: 32 bits)
BAR0 Reference: [B:0, D:21, F:0] + 10h

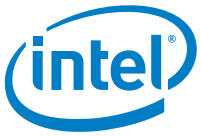
Default: 00028F5Ch



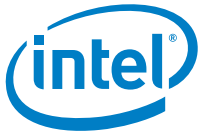


Bit Range	Default & Access	Description
31:24	0b RO	**RSV:** Reserved.
23:0	028F5Ch RW	**DDS Clock Rate (DDS_CLK_RATE):** DDS_CLK_RATE is a 24 bit value that is incrementally added to a 24 bit wrapping internal counter to cause the clock output to toggle. The frequency of the internal generated clock is given by: DDS_FREQ = SYS_FREQ*(DDS_CLK_RATE/2exp24) where SYS_FREQ is 200MHz. NOTE: in order to minimize jitter and duty cycle on the generated clock, this register should be set as described in the SPI section of the design specification document.

§ §



October 2013
Document Number: 329676-001US



21.0 Legacy Bridge

The Legacy Bridge is a collection of hardware blocks that are critical to implement a PC/AT compatible platform. Certain legacy hardware functions are required to support commercially available, shrink-wrap operating systems. In addition, the Legacy Bridge provides interrupt decoding and routing functionality, power management features, and a SPI interface for system firmware.

21.1 Features

The key features of the various blocks are as follows:

- General Purpose Input Output
  - Legacy control interface for SoC GPIOs
  - I/O mapped registers
- 8259 Programmable Interrupt Controller
  - Legacy interrupt support
  - 15 total interrupts through two cascaded controllers
  - I/O mapped registers
- I/O Advanced Programmable Interrupt Controller
  - Legacy-free interrupt support
  - 24 total interrupts
  - Memory mapped registers
- 8254 Programmable Interval Timer
  - Legacy timer support
  - Three timers with fixed uses: System Timer, Refresh Request Signal, and Speaker Tone
  - I/O mapped registers
- HPET - High Performance Event Timers
  - Legacy-free timer support
  - Three timers and one counter
  - Memory mapped registers
- Real-Time Clock (RTC)
  - 242-byte RAM backed by battery (aka CMOS RAM)
  - Can generate wake/interrupt when time matches programmed value
  - I/O and indexed registers
- Watchdog Timer (WDT)
  - Provides ability to trigger a reset in the event of an unresponsive system
  - Resolution from 1μsec to 17 minutes

  - I/O mapped registers
- Serial Peripheral Interface (SPI)
  - Support for one SPI Flash, of up to 16 Mbyte, only. No other SPI peripherals are supported.
  - Stores boot FW and system configuration data
  - Supports SPI clock frequency of 20 MHz.
  - Memory mapped registers
- Power Management Controller (PMC)
  - Provides a software interface to control many of the power management features present in the SoC.

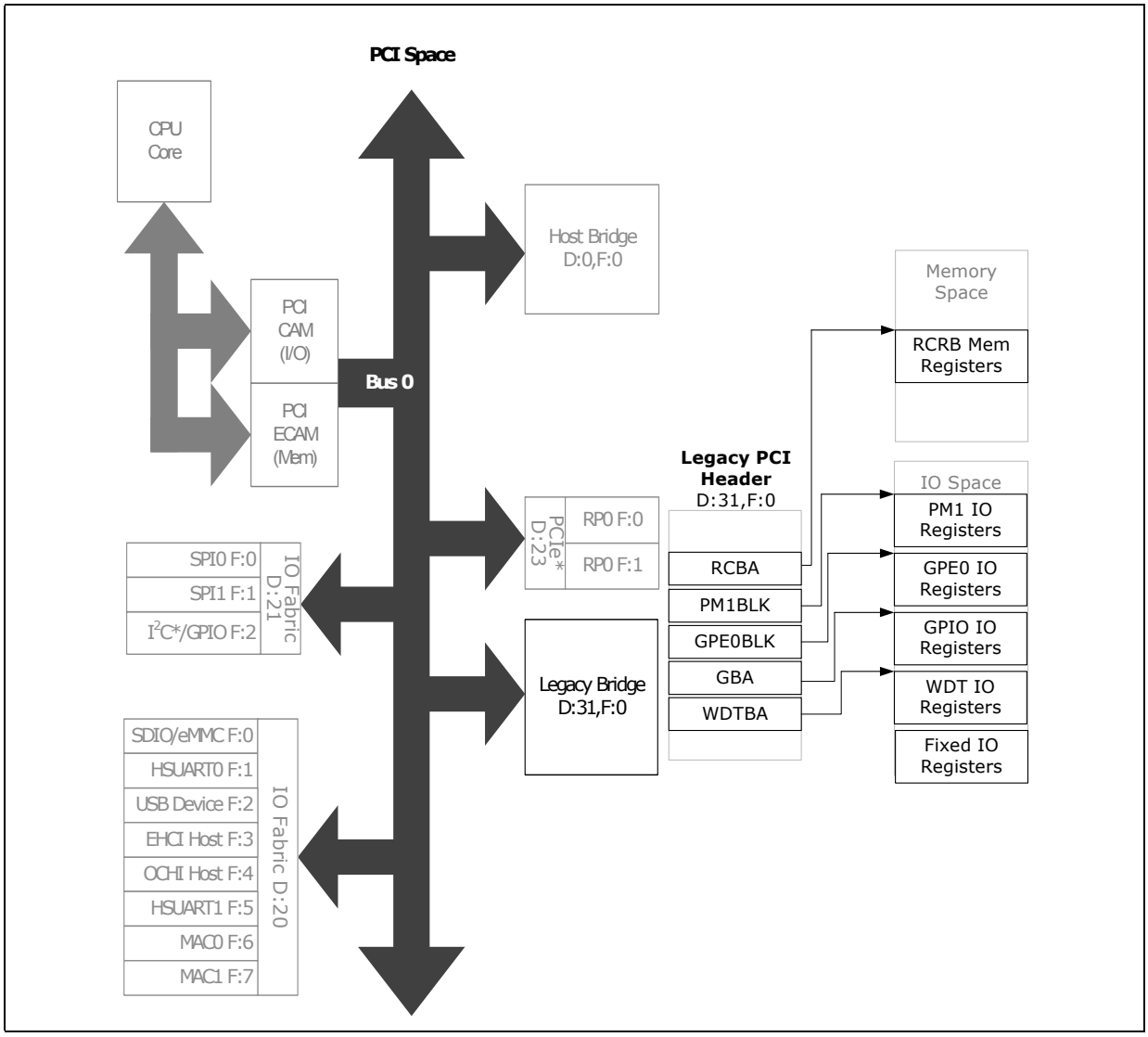
The following features are not supported by the Legacy Bridge:

- LPC Interface
- Serial IRQ (SERIRQ)
- SMBus

21.2 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 45. Legacy Bridge Register Map



21.3 PCI Configuration Registers

Table 112. Summary of PCI Configuration Registers—0/31/0

Offset	Size	Register ID—Description	Default Value
0h	4	"PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h" on page 804	095E8086h
4h	4	"PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h" on page 804	00000003h
8h	4	"PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h" on page 805	06010000h
Ch	4	"PCI Miscellaneous Fields (PCI_MISC)—Offset Ch" on page 805	00000000h

Table 112. Summary of PCI Configuration Registers—0/31/0 (Continued)

Offset	Size	Register ID—Description	Default Value
2Ch	4	"PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch" on page 806	00000000h
44h	4	"GPIO Base Address (GBA)—Offset 44h" on page 807	00000000h
48h	4	"PM1_BLK Base Address (PM1BLK)—Offset 48h" on page 807	00000000h
4Ch	4	"GPE0_BLK Base Address (GPE0BLK)—Offset 4Ch" on page 807	00000000h
58h	4	"ACPI Control (ACTL)—Offset 58h" on page 808	00000003h
60h	4	"PIRQA, PIRQB, PIRQC and PIRQD Routing Control (PABCDRC)—Offset 60h" on page 808	80808080h
64h	4	"PIRQE, PIRQF, PIRQG and PIRQH Routing Control (PEFGHRC)—Offset 64h" on page 810	80808080h
84h	4	"Watch Dog Timer Base Address (WDTBA)—Offset 84h" on page 810	00000000h
D4h	4	"BIOS Decode Enable (BCE)—Offset D4h" on page 811	FF000000h
D8h	4	"BIOS Control (BC)—Offset D8h" on page 812	00000100h
F0h	4	"Root Complex Base Address (RCBA)—Offset F0h" on page 813	00000000h

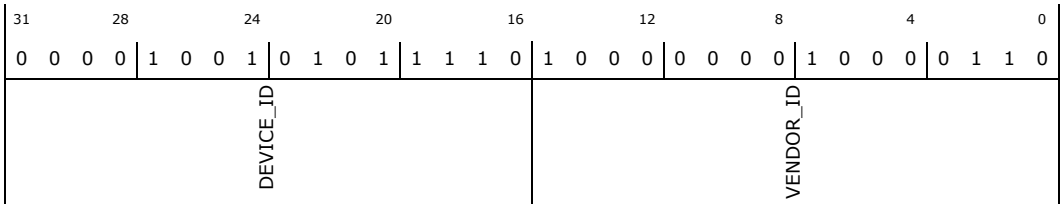
21.3.1 PCI Device ID and Vendor ID Fields (PCI_DEVICE_VENDOR)—Offset 0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_DEVICE_VENDOR: [B:0, D:31, F:0] + 0h

Default: 095E8086h



Bit Range	Default & Access	Description
31:16	095Eh RO	**Device ID (DEVICE_ID):** PCI Device ID
15:0	8086h RO	**Vendor ID (VENDOR_ID):** PCI Vendor ID for Intel

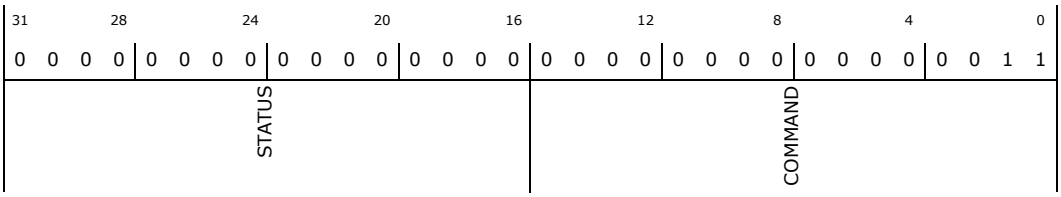
21.3.2 PCI Status and Command Fields (PCI_STATUS_COMMAND)—Offset 4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_STATUS_COMMAND: [B:0, D:31, F:0] + 4h

Default: 00000003h



Bit Range	Default & Access	Description
31:16	0000h RO	**Status (STATUS):** Hardwired to 0.
15:0	0003h RO	**Command (COMMAND):** Hardwired to 0.

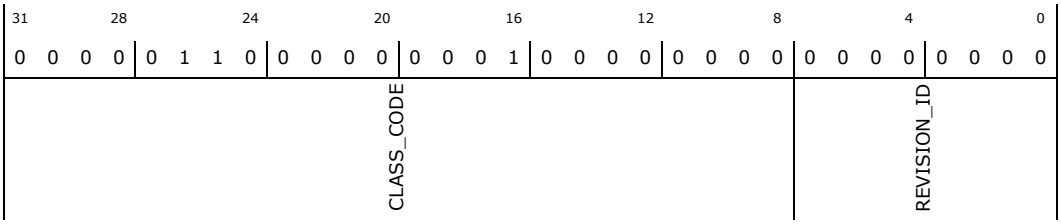
21.3.3 PCI Class Code and Revision ID Fields (PCI_CLASS_REVISION)—Offset 8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_CLASS_REVISION: [B:0, D:31, F:0] + 8h

Default: 06010000h



Bit Range	Default & Access	Description
31:8	060100h RO	**Class Code (CLASS_CODE):** PCI Class Code for Bridge
7:0	00h RO	**Revision ID (REVISION_ID):** PCI Revision ID

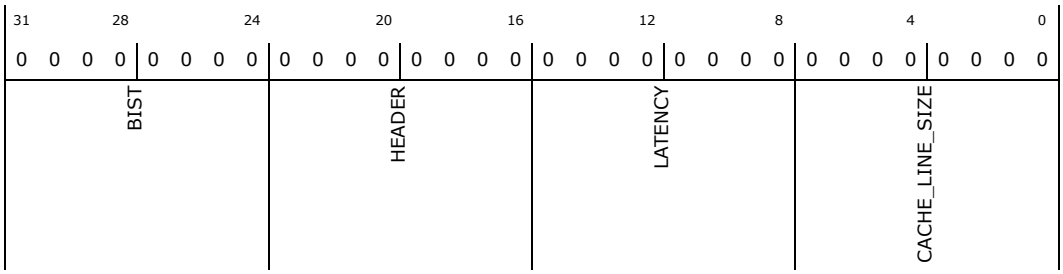
21.3.4 PCI Miscellaneous Fields (PCI_MISC)—Offset Ch

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_MISC: [B:0, D:31, F:0] + Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:24	00h RO	**BIST:** PCI BIST Field
23:16	00h RO	**Header Type (HEADER):** PCI Header Type Field
15:8	00h RO	**Latency Timer (LATENCY):** PCI Latency Timer Field
7:0	00h RO	**Cache Line Size (CACHE_LINE_SIZE):** PCI Cache Line Size Field

21.3.5 PCI Subsystem ID and Subsystem Vendor ID Fields (PCI_SUBSYSTEM)—Offset 2Ch

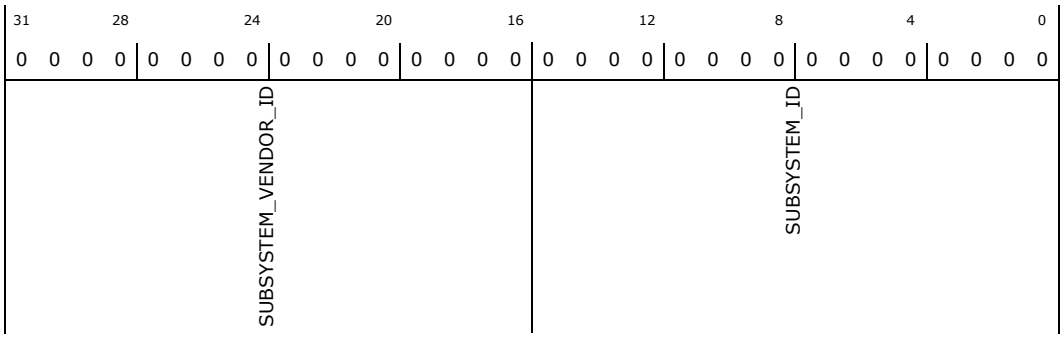
This register is initialized to logic 0 by the assertion of RESET#. This register can be written only once after RESET# de-assertion.

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PCI_SUBSYSTEM: [B:0, D:31, F:0] + 2Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:16	0000h RW/O	**Subsystem Vendor ID (SUBSYSTEM_VENDOR_ID):** This is written by BIOS. No hardware action taken.
15:0	0000h RW/O	**Subsystem ID (SUBSYSTEM_ID):** This is written by BIOS. No hardware action taken.

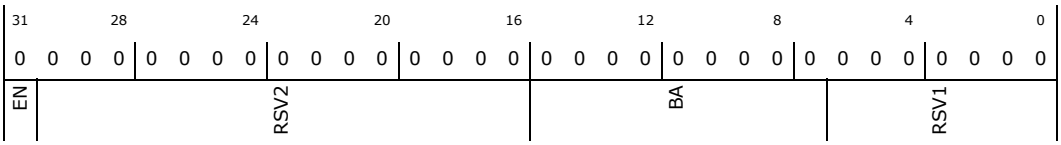
21.3.6 GPIO Base Address (GBA)—Offset 44h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

GBA: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Enable (EN):** When set, decode of the I/O range pointed to by the BA is enabled.
30:16	0b RO	**Reserved (RSV2):** Reserved.
15:7	0b RW	**Base Address (BA):** Provides the 128 bytes of I/O space for GPIO.
6:0	0b RO	**Reserved (RSV1):** Reserved.

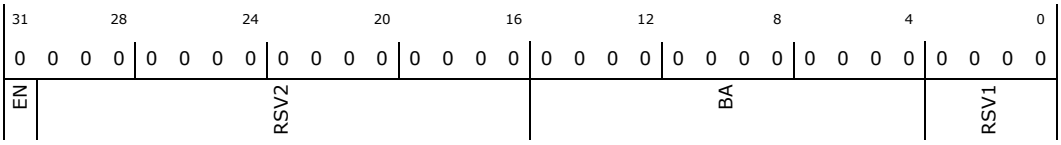
21.3.7 PM1_BLK Base Address (PM1BLK)—Offset 48h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PM1BLK: [B:0, D:31, F:0] + 48h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Enable (EN):** When set, decode of the I/O range pointed to by the BA is enabled.
30:16	0b RO	**Reserved (RSV2):** Reserved.
15:4	0b RW	**Base Address (BA):** Provides the 16 bytes of I/O space for PM1_BLK.
3:0	0b RO	**Reserved (RSV1):** Reserved.

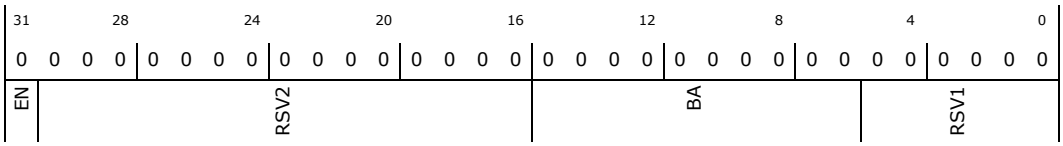
21.3.8 GPE0_BLK Base Address (GPE0BLK)—Offset 4Ch

Access Method

Type: PCI Configuration Register (Size: 32 bits)

GPE0BLK: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Enable (EN):** When set, decode of the I/O range pointed to by the BA is enabled.
30:16	0b RO	**Reserved (RSV2):** Reserved.
15:6	0b RW	**Base Address (BA):** Provides the 64 bytes of I/O space for GPE0_BLK.
5:0	0b RO	**Reserved (RSV1):** Reserved.

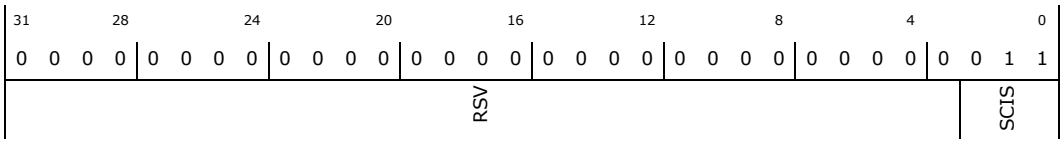
21.3.9 ACPI Control (ACTL)—Offset 58h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

ACTL: [B:0, D:31, F:0] + 58h

Default: 00000003h



Bit Range	Default & Access	Description
31:3	0b RO	**Reserved (RSV):** Reserved.
2:0	011b RW	**SCI IRQ Select (SCIS):** Specifies to which IRQ the SCI is routed. If not using APIC, SCI must be routed to IRQ9-11, and that interrupt is not shared with SERIRQ, but is shared with other interrupts. If using APIC, SCI can be mapped to IRQ20-23, and can be shared with other interrupts. 000b : IRQ9 001b : IRQ10 010b : IRQ11 011b : SCI Disabled 100b : IRQ20 101b : IRQ21 110b : IRQ22 111b : IRQ23 When the interrupt is mapped to APIC interrupts 9, 10 or 11, APIC must be programmed for active-high reception. When the interrupt is mapped to APIC interrupt 20 through 23, PIC must be programmed for active-low reception.

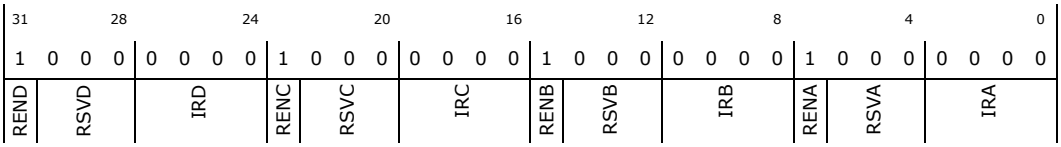
21.3.10 PIRQA, PIRQB, PIRQC and PIRQD Routing Control (PABCDRC)—Offset 60h

Access Method

Type: PCI Configuration Register
(Size: 32 bits)

PABCDRC: [B:0, D:31, F:0] + 60h

Default: 80808080h



Bit Range	Default & Access	Description
31	1b RW	**Interrupt Routing Enable for PIRQD (REND):** When cleared, PIRQD is routed to one of the legacy interrupts specified in bits[27:24]. When set, PIRQD is not routed to the 8259.
30:28	0b RO	**Reserved (RSVD):** Reserved.
27:24	0b RW	**IRQ for PIRQD (IRD):** Indicates how to route PIRQD. 0h - Reserved 1h - Reserved 2h - IRQ2 3h - IRQ3 4h - IRQ4 5h - IRQ5 6h - IRQ6 7h - IRQ7 8h - IRQ8 9h - IRQ9 Ah - IRQ10 Bh - IRQ11 Ch - IRQ12 Dh - Reserved Eh - IRQ14 Fh - IRQ15
23	1b RW	**Interrupt Routing Enable for PIRQC (RENC):** When cleared, PIRQC is routed to one of the legacy interrupts specified in bits[19:16]. When set, PIRQC is not routed to the 8259.
22:20	0b RO	**Reserved (RSVC):** Reserved.
19:16	0b RW	**IRQ for PIRQC (IRC):** Indicates how to route PIRQC.
15	1b RW	**Interrupt Routing Enable for PIRQB (RENB):** When cleared, PIRQB is routed to one of the legacy interrupts specified in bits[11:8]. When set, PIRQB is not routed to the 8259.
14:12	0b RO	**Reserved (RSVB):** Reserved.
11:8	0b RW	**IRQ for PIRQB (IRB):** Indicates how to route PIRQB.
7	1b RW	**Interrupt Routing Enable for PIRQA (RENA):** When cleared, PIRQA is routed to one of the legacy interrupts specified in bits[3:0]. When set, PIRQA is not routed to the 8259.
6:4	0b RO	**Reserved (RSVA):** Reserved.
3:0	0b RW	**IRQ for PIRQA (IRA):** Indicates how to route PIRQA.

21.3.11 PIRQE, PIRQF, PIRQG and PIRQH Routing Control (PEFGHRC)—Offset 64h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

PEFGHRC: [B:0, D:31, F:0] + 64h

Default: 80808080h

Bit Range	Default & Access	Description
31	1b RW	**Interrupt Routing Enable for PIRQH (RENH):** When cleared, PIRQH is routed to one of the legacy interrupts specified in bits[27:24]. When set, PIRQH is not routed to the 8259.
30:28	0b RO	**Reserved (RSVH):** Reserved.
27:24	0b RW	**IRQ for PIRQH (IRH):** Indicates how to route PIRQH.
23	1b RW	**Interrupt Routing Enable for PIRQG (RENG):** When cleared, PIRQG is routed to one of the legacy interrupts specified in bits[19:16]. When set, PIRQG is not routed to the 8259.
22:20	0b RO	**Reserved (RSVG):** Reserved.
19:16	0b RW	**IRQ for PIRQG (IRG):** Indicates how to route PIRQG.
15	1b RW	**Interrupt Routing Enable for PIRQF (RENF):** When cleared, PIRQF is routed to one of the legacy interrupts specified in bits[11:8]. When set, PIRQF is not routed to the 8259.
14:12	0b RO	**Reserved (RSVF):** Reserved.
11:8	0b RW	**IRQ for PIRQF (IRF):** Indicates how to route PIRQF.
7	1b RW	**Interrupt Routing Enable for PIRQE (RENE):** When cleared, PIRQE is routed to one of the legacy interrupts specified in bits[3:0]. When set, PIRQE is not routed to the 8259.
6:4	0b RO	**Reserved (RSVE):** Reserved.
3:0	0b RW	**IRQ for PIRQE (IRE):** Indicates how to route PIRQE.

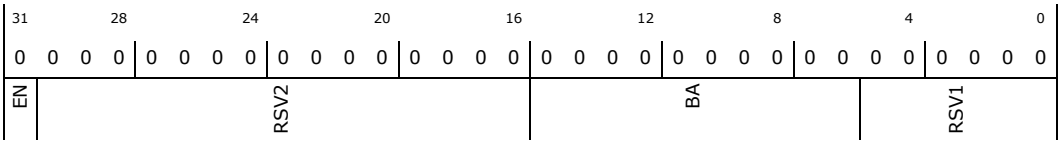
21.3.12 Watch Dog Timer Base Address (WDTBA)—Offset 84h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

WDTBA: [B:0, D:31, F:0] + 84h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Enable (EN):** When set, decode of the I/O range pointed to by the BA is enabled.
30:16	0b RO	**Reserved (RSV2):** Reserved.
15:6	0b RW	**Base Address (BA):** Provides the 64 bytes of I/O space for WDT.
5:0	0b RO	**Reserved (RSV1):** Reserved.

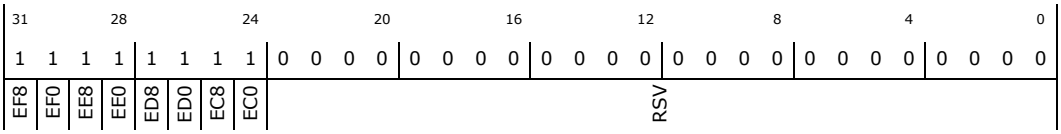
21.3.13 BIOS Decode Enable (BCE)—Offset D4h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BCE: [B:0, D:31, F:0] + D4h

Default: FF000000h



Bit Range	Default & Access	Description
31	1b RO	**F8-FF Enable (EF8):** Enables decoding of BIOS range FFF80000h - FFFFFFFFh and FFB80000h - FFBFFFFFh. 0 = Disable 1 = Enable
30	1b RW	**F0-F0 Enable (EF0):** Enables decoding of BIOS range FFF00000h - FFF7FFFFh and FFB00000h - FFB7FFFFh. 0 = Disable 1 = Enable
29	1b RW	**E8-EF Enable (EE8):** Enables decoding of BIOS range FFE80000h - FFEFFFFFh and FFA80000h - FFAFFFFFh. 0 = Disable 1 = Enable
28	1b RW	**E0-E8 Enable (EE0):** Enables decoding of BIOS range FFE00000h - FFE7FFFFh and FFA00000h - FFA7FFFFh. 0 = Disable 1 = Enable
27	1b RW	**D8-DF Enable (ED8):** Enables decoding of BIOS range FFD80000h - FFDFFFFFh and FF980000h - FF9FFFFFh. 0 = Disable 1 = Enable
26	1b RW	**D0-D8 Enable (ED0):** Enables decoding of BIOS range FFD00000h - FFD7FFFFh and FF900000h - FF97FFFFh. 0 = Disable 1 = Enable

Bit Range	Default & Access	Description
25	1b RW	**C8-CF Enable (EC8):** Enables decoding of BIOS range FFC80000h - FFCFFFFFh and FF880000h - FF8FFFFFh. 0 = Disable 1 = Enable
24	1b RW	**C0-C8 Enable (EC0):** Enables decoding of BIOS range FFC00000h - FFC7FFFFh and FF800000h - FF87FFFFh. 0 = Disable 1 = Enable
23:0	0b RO	**Reserved (RSV):** Reserved.

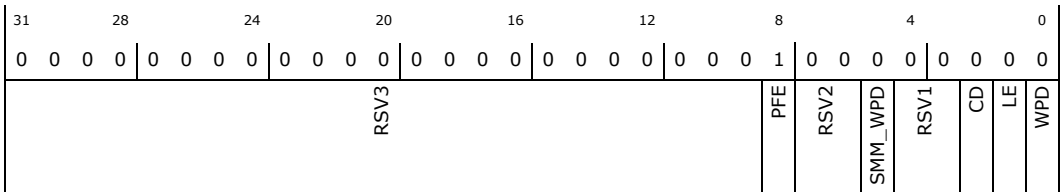
21.3.14 BIOS Control (BC)—Offset D8h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

BC: [B:0, D:31, F:0] + D8h

Default: 00000100h



Bit Range	Default & Access	Description
31:9	0b RO	**Reserved (RSV3):** Reserved.
8	1b RW	**Prefetch Enable (PFE):** When set, BIOS prefetching is enabled. An access to BIOS causes a 64-byte fetch of the line starting at that region. Subsequent accesses within that region result in data being returned from the prefetch buffer. The prefetch buffer is invalidated when this bit is cleared, or a BIOS access occurs to a different line than what is currently in the buffer
7:6	0b RO	**Reserved (RSV2):** Reserved.
5	0b RW	**SMM Write Protect Disable (SMM_WPD):** When LE is clear: Setting this bit has no effect. When LE is set: Setting this bit enables both read and write cycles to the SPI Flash, clearing this bit blocks write cycles to the SPI Flash. This bit is not writeable unless the processor is in SMM mode. This bit must be cleared before the processor exits SMM mode to prevent write cycles to SPI flash when the processor is in a non-SMM mode.
4:3	0b RO	**Reserved (RSV1):** Reserved.
2	0b RW	**Cache Disable (CD):** Enable caching in read buffer for direct memory read.
1	0b RW/P	**Lock Enable (LE):** When cleared, setting the WPD bit will not generate SMIs and the WPD bit is used to enable write cycles to the SPI Flash. When set, enables setting the WPD bit to generate SMIs and the SMM_WPD bit is used to enable write cycles to the SPI Flash. Once set, this bit can only be cleared by a reset.

Bit Range	Default & Access	Description
0	0b RW	**Write Protect Disable (WPD):** When LE is clear: Setting this bit enables both read and write cycles into the SPI Flash, clearing this bit blocks write cycles to the SPI Flash. When LE is set: Setting this bit will generate an SMI, the SMM_WPD bit must then be used to enable write cycles to the SPI Flash. This bit must be cleared before the processor exits SMM mode in order to clear the SMI source.

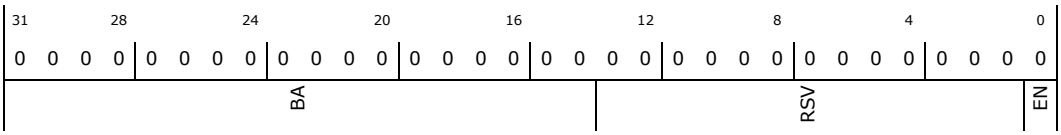
21.3.15 Root Complex Base Address (RCBA)—Offset F0h

Access Method

Type: PCI Configuration Register (Size: 32 bits)

RCBA: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:14	0b RW	**Base Address (BA):** Base Address for the root complex register block decode range. This address is aligned on a 16 KB boundary.
13:1	0b RO	**Reserved (RSV):** Reserved.
0	0b RW	**Enable (EN):** When set, enables the range specified in BA to be claimed as the RCRB.

21.4 Memory Mapped Registers

21.4.1 Root Complex Register Block

This block describes all registers and base functionality that are related to SoC configuration but not a specific interface. It contains the root complex register block. Accesses in this space are limited to 32-bit quantities. Burst accesses are not allowed.

Table 113. Summary of Memory Mapped I/O Registers—RCBA

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"Root Complex Topology Capabilities List (RCTCL)—Offset 0h" on page 814	00010005h
4h	7h	"Element Self Description (ESD)—Offset 4h" on page 814	00000102h
3140h	3141h	"Interrupt Queue Agent 0 (IRQAGENT0)—Offset 3140h" on page 815	0000h
3142h	3143h	"Interrupt Queue Agent 1 (IRQAGENT1)—Offset 3142h" on page 815	0000h
3144h	3145h	"Interrupt Queue Agent 2 (IRQAGENT2)—Offset 3144h" on page 816	0000h
3146h	3147h	"Interrupt Queue Agent 3 (IRQAGENT3)—Offset 3146h" on page 816	0000h
3400h	3403h	"Root Complex Topology Capabilities List (RCTCL)—Offset 0h" on page 814	00000000h

21.4.1.1 Root Complex Topology Capabilities List (RCTCL)—Offset 0h

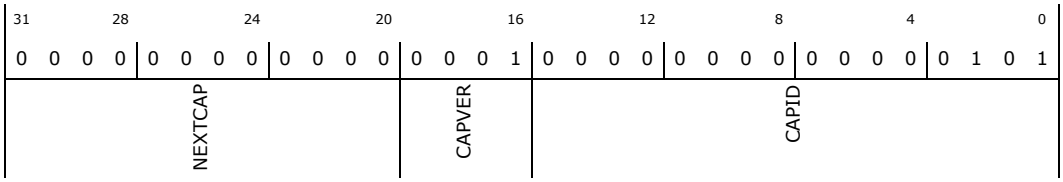
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RCTCL: [RCBA] + 0h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00010005h



Bit Range	Default & Access	Description
31:20	0b RO	**Next Capability (NEXTCAP):** Indicates next item in the list
19:16	1h RO	**Capability Version (CAPVER):** Indicates the version of the capability structure.
15:0	0005h RO	**Capability ID (CAPID):** Indicates this is a PCI Express link capability section of an RCRB

21.4.1.2 Element Self Description (ESD)—Offset 4h

Provides information about the root complex element containing the Link Declaration Capability

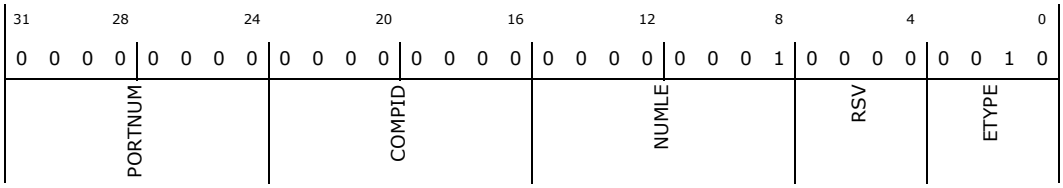
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

ESD: [RCBA] + 4h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000102h



Bit Range	Default & Access	Description
31:24	0b RO	**Port Number (PORTNUM):** A value of 0 to indicate the egress port
23:16	0b RW/O	**Component ID (COMPID):** Indicates the component ID assigned to this element by software. This is written once by BIOS and is locked until a reset.
15:8	01h RO	**Number of Link Entries (NUMLE):** Indicates that one link entry is described by this RCRB.

Bit Range	Default & Access	Description
7:4	0b RO	**Reserved (RSV):** Reserved.
3:0	2h RO	**Element Type (ETYPE):** Indicates that the element type is a root complex internal link

21.4.1.3 Interrupt Queue Agent 0 (IRQAGENT0)—Offset 3140h

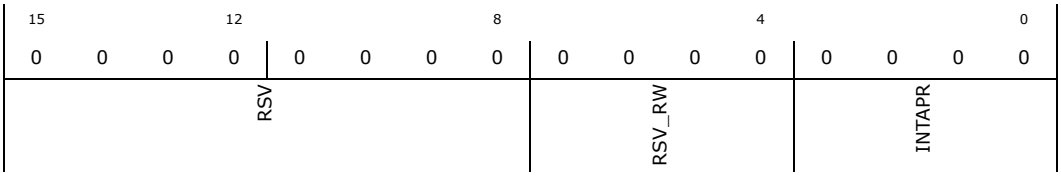
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

IRQAGENT0: [RCBA] + 3140h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0000h



Bit Range	Default & Access	Description
15:8	0b RO	**Reserved (RSV):** Reserved.
7:4	0b RW	**Reserved (RSV_RW):** Reserved.
3:0	0b RW	**Interrupt A Pin Route (INTAPR):** Indicates which PIRQ routing used for INTA#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTA# will be routed to PIRQG if this field is set to 0x6.

21.4.1.4 Interrupt Queue Agent 1 (IRQAGENT1)—Offset 3142h

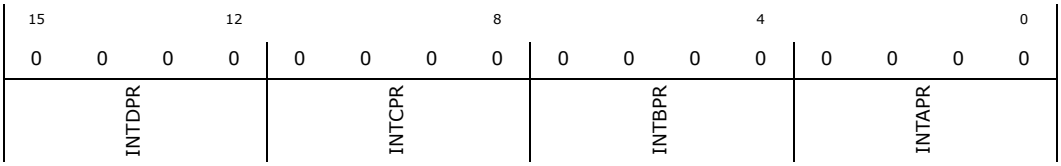
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

IRQAGENT1: [RCBA] + 3142h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0000h



Bit Range	Default & Access	Description
15:12	0b RW	**Interrupt D Pin Route (INTDPR):** Indicates which PIRQ routing used for INTD#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTD# will be routed to PIRQG if this field is set to 0x6.
11:8	0b RW	**Interrupt C Pin Route (INTCPR):** Indicates which PIRQ routing used for INTC#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTC# will be routed to PIRQG if this field is set to 0x6.
7:4	0b RW	**Interrupt B Pin Route (INTBPR):** Indicates which PIRQ routing used for INTB#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTB# will be routed to PIRQG if this field is set to 0x6.
3:0	0b RW	**Interrupt A Pin Route (INTAPR):** Indicates which PIRQ routing used for INTA#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTA# will be routed to PIRQG if this field is set to 0x6.

21.4.1.5 Interrupt Queue Agent 2 (IRQAGENT2)—Offset 3144h

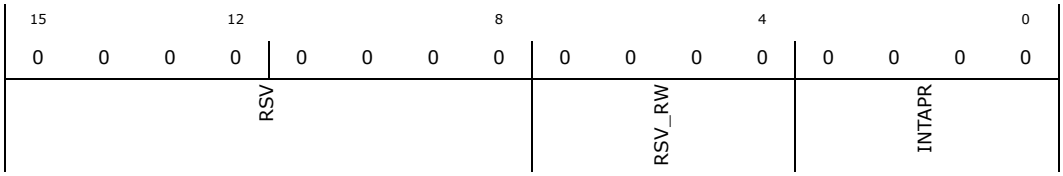
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

IRQAGENT2: [RCBA] + 3144h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0000h



Bit Range	Default & Access	Description
15:8	0b RO	**Reserved (RSV):** Reserved.
7:4	0b RW	**Reserved (RSV_RW):** Reserved.
3:0	0b RW	**Interrupt A Pin Route (INTAPR):** Indicates which PIRQ routing used for INTA#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTA# will be routed to PIRQG if this field is set to 0x6.

21.4.1.6 Interrupt Queue Agent 3 (IRQAGENT3)—Offset 3146h

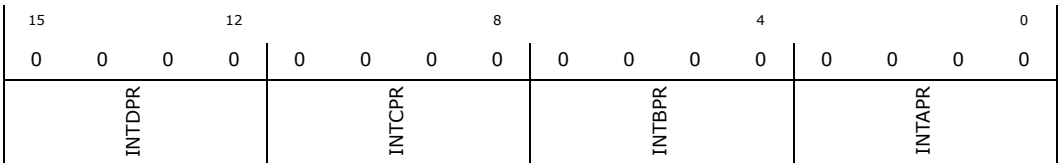
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

IRQAGENT3: [RCBA] + 3146h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0000h



Bit Range	Default & Access	Description
15:12	0b RW	**Interrupt D Pin Route (INTDPR):** Indicates which PIRQ routing used for INTD#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTD# will be routed to PIRQG if this field is set to 0x6.
11:8	0b RW	**Interrupt C Pin Route (INTCPR):** Indicates which PIRQ routing used for INTC#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTC# will be routed to PIRQG if this field is set to 0x6.
7:4	0b RW	**Interrupt B Pin Route (INTBPR):** Indicates which PIRQ routing used for INTB#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTB# will be routed to PIRQG if this field is set to 0x6.
3:0	0b RW	**Interrupt A Pin Route (INTAPR):** Indicates which PIRQ routing used for INTA#. Legal values are [0x0-0x7] corresponding to PIRQ[A-H]. For example, INTA# will be routed to PIRQG if this field is set to 0x6.

21.4.1.7 RTC Configuration (RC)—Offset 3400h

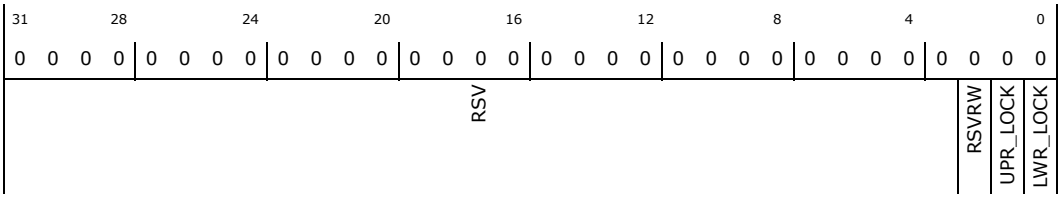
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

RC: [RCBA] + 3400h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:3	0b RO	**Reserved (RSV):** Reserved.
2	0b RW	**Reserved (RSVRW):** Reserved.
1	0b RW/O	**Upper 128 Byte Lock (UPR_LOCK):** When set, bytes 38h-3Fh in the upper 128 byte bank of RTC RAM are locked. Writes will be dropped and reads will not return any guaranteed data.
0	0b RW/O	**Lower 128 Byte Lock (LWR_LOCK):** When set, bytes 38h-3Fh in the lower 128 byte bank of RTC RAM are locked. Writes will be dropped and reads will not return any guaranteed data.

21.5 IO Registers

The Legacy Bridge contains a mix of fixed address I/O Registers and I/O Registers that are mapped by BARs in the Legacy Bridge configuration space. This sections describes the Fixed I/O registers and the Legacy ACPI I/O Register.s All other I/O Registers are described in the relevant sections later in this chapter.

21.5.1 Fixed IO Registers

Table 114. Summary of I/O Registers

Offset Start	Offset End	Register ID—Description	Default Value
61h	61h	"NMI Status and Control Register (NSC)—Offset 61h" on page 818	00h
70h	70h	"NMI Enable and RTC Index Register (NMIE)—Offset 70h" on page 819	80h
B2h	B2h	"Software SMI Control Port (SWSMICTL)—Offset B2h" on page 819	00h
B3h	B3h	"Software SMI Status Port (SWSMISTS)—Offset B3h" on page 820	00h
CF9h	CF9h	"Reset Control Register (RSTC)—Offset CF9h" on page 820	00h

21.5.1.1 NMI Status and Control Register (NSC)—Offset 61h

Access Method

Type: I/O Register
(Size: 8 bits)

NSC: 61h

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
SERR_NMI_STATUS	RSVD	CNTR2_STATUS	CNTR1_TOGGLE_STATUS	RSVD	SERR_NMI_ENABLE	SPKR_ENABLE	CNTR2_ENABLE

Bit Range	Default & Access	Description
7	0b RO	**SERR# NMI Status (SERR_NMI_STATUS):** Set on errors from a PCIe port or internal functions that generate SERR#. SERR# NMI Enable in this register must be cleared in order for this bit to be set. To reset the interrupt, set bit 2 to 1 and then set it to 0.
6	0b RO	**Reserved (RSVD):** Reserved.
5	0b RO	**Timer Counter 2 Status (CNTR2_STATUS):** Reflects the current state of the 8254 counter 2 output. Counter 2 must be programmed for this bit to have a determinate value.
4	0b RO	**Refresh Cycle Toggle Status (CNTR1_TOGGLE_STATUS):** Reflects the current state of 8254 counter 1.
3	0b RO	**Reserved (RSVD):** Reserved.

Bit Range	Default & Access	Description
2	0b RW	**SERR# NMI Enable (SERR_NMI_ENABLE):** When set, SERR# NMIs are disabled. When cleared, SERR# NMIs are enabled.
1	0b RW	**Speaker Data Enable (SPKR_ENABLE):** When this bit is a 0, the SPKR output is a 0. When this bit is a 1, the SPKR output is equivalent to the Counter 2 OUT signal value.
0	0b RW	**Timer Counter 2 Enable (CNTR2_ENABLE):** When cleared, counter 2 counting is disabled. When set, counting is enabled.

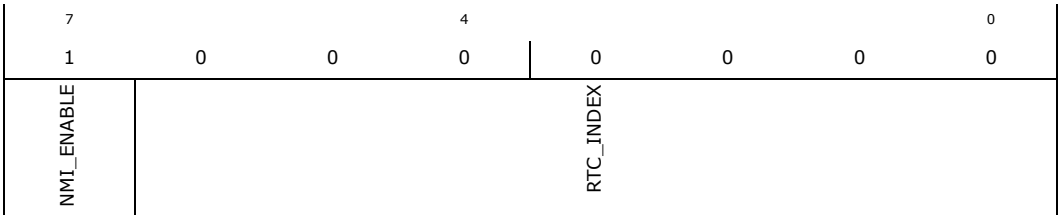
21.5.1.2 NMI Enable and RTC Index Register (NMIE)—Offset 70h

Access Method

Type: I/O Register
(Size: 8 bits)

NMIE: 70h

Default: 80h



Bit Range	Default & Access	Description
7	1b WO	**NMI Enable (NMI_ENABLE):** When set, NMI sources disabled. When cleared, NMI sources enabled.
6:0	0b WO	**Real Time Clock Index (RTC_INDEX):** Selects RTC register or CMOS RAM address to access.

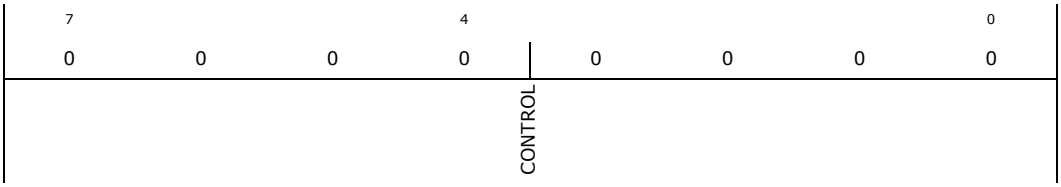
21.5.1.3 Software SMI Control Port (SWSMICTL)—Offset B2h

Access Method

Type: I/O Register
(Size: 8 bits)

SWSMICTL: B2h

Default: 00h



Bit Range	Default & Access	Description
7:0	0b RW	**Software SMI Control Port (CONTROL):** This port is used to pass a command between the OS and the SMI handler. Writes to this port store data, set APM bit of SMI Status register of GPE0 Block, and generate SMI# when APM is set.

21.5.1.4 Software SMI Status Port (SWSMISTS)—Offset B3h

Access Method

Type: I/O Register
(Size: 8 bits)

SWSMISTS: B3h

Default: 00h

Bit Range	Default & Access	Description
7:0	0b RW	**Software SMI Status Port (STATUS):** This port is used to pass data between the OS and the SMI handler. This is a scratchpad register.

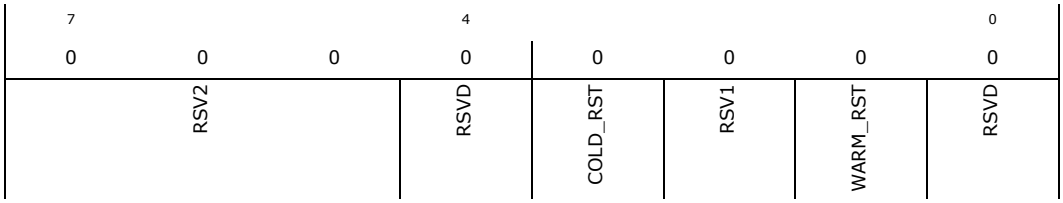
21.5.1.5 Reset Control Register (RSTC)—Offset CF9h

Access Method

Type: I/O Register
(Size: 8 bits)

RSTC: CF9h

Default: 00h



Bit Range	Default & Access	Description
7:5	0b RO	**Reserved (RSV2):** Reserved.
4	0b RO	**Reserved (RSVD):** Reserved.
3	0b RW	**Cold Reset (COLD_RST):** This bit causes SLPMODE, and RSTRDY# to be driven low, while SLPRDY# remains high. In response to this, the platform will perform a full power cycle
2	0b RO	**Reserved (RSV1):** Reserved.
1	0b RW	**Warm Reset (WARM_RST):** This bit causes RSTRDY# to be driven low, with SLPMODE high, while SLPRDY# remains high. In response to this, the platform will pulse RESET# low to reset the CPU and all peripherals
0	0b RO	**Reserved (RSVD):** Reserved.

21.5.2 ACPI GPE0 Block

Table 115. Summary of I/O Registers—GPE0BLK

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"GPE0 Status Register (GPE0STS)—Offset 0h" on page 821	00000000h
4h	7h	"GPE0 Enable Register (GPE0EN)—Offset 4h" on page 822	00000000h
10h	13h	"SMI Enable Register (SMIEN)—Offset 10h" on page 822	00000000h
14h	17h	"SMI Status Register (SMISTS)—Offset 14h" on page 824	00000000h
18h	1Bh	"General Purpose Event Control Register (GPEC)—Offset 18h" on page 825	00000000h
28h	2Bh	"Power Management Configuration Core Well Register (PMCW)—Offset 28h" on page 825	00000000h
2Ch	2Fh	"Power Management Configuration Suspend Well Register (PMSW)—Offset 2Ch" on page 826	00000000h
30h	33h	"Power Management Configuration RTC Well Register (PMRW)—Offset 30h" on page 826	0000000Bh

21.5.2.1 GPE0 Status Register (GPE0STS)—Offset 0h

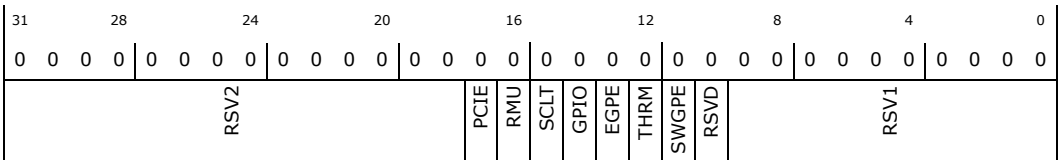
Access Method

Type: I/O Register
(Size: 32 bits)

GPE0STS: [GPE0BLK] + 0h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:18	0b RO	**Reserved (RSV2):** Reserved.
17	0b RW/1C	**PCIE Status (PCIE):** Set when an Assert SCI message from PCIe Controller is received.
16	0b RW/1C	**Remote Management Unit Status (RMU):** Set when an Assert SCI message from the Remote Management Unit is received.
15	0b RW/1C	**Device Status (SCLT):** Set when the SCI signal from Device:20 or Device:21 goes active.
14	0b RW/1C	**GPIO Status (GPIO):** Set when a GPIO configured for GPE goes active.
13	0b RW/1C	**External GPE Status (EGPE):** Set when the GPE_B signal goes active.
12	0b RW/1C	**Thermal Status (THRM):** Set anytime THRM_B is received at the state defined by GPEC.TPOL.
11	0b RW/1C	**Software GPE Status (SWGPE):** Set when GPEC.SWGPE is set.

Bit Range	Default & Access	Description
10	0b RO	**Reserved (RSVD):** Reserved.
9:0	0b RO	**Reserved (RSV1):** Reserved.

21.5.2.2 GPE0 Enable Register (GPE0EN)—Offset 4h

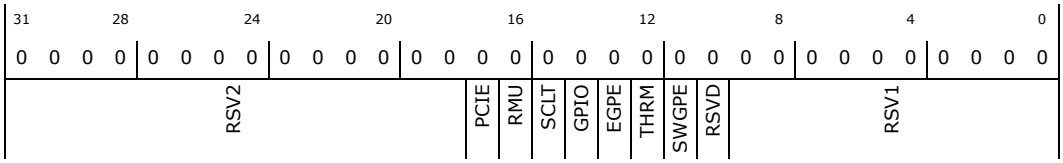
Access Method

Type: I/O Register
(Size: 32 bits)

GPE0EN: [GPE0BLK] + 4h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:18	0b RO	**Reserved (RSV2):** Reserved.
17	0b RW	**PCIe Enable (PCIE):** When set enables GPE0STS.PCIE to generate SCI/SMI.
16	0b RW	**Remote Management Unit Enable (RMU):** When set enables GPE0STS.RMU to generate SCI/SMI.
15	0b RW	**Device Enable (SCLT):** When set enables GPE0STS.SCLT to generate SCI/SMI.
14	0b RW	**GPIO Enable (GPIO):** When set enables GPE0STS.GPIO to generate SCI/SMI.
13	0b RW	**External GPE Enable (EGPE):** When set enables GPE0STS.EGPE to generate SCI/SMI.
12	0b RW	**Thermal Enable (THRM):** When set enables GPE0STS.THRM to generate SCI/SMI.
11	0b RW	**Software GPE Enable (SWGPE):** When set enables GPE0STS.SWGPE to generate SCI/SMI.
10	0b RO	**Reserved (RSVD):** Reserved.
9:0	0b RO	**Reserved (RSV1):** Reserved.

21.5.2.3 SMI Enable Register (SMIEN)—Offset 10h

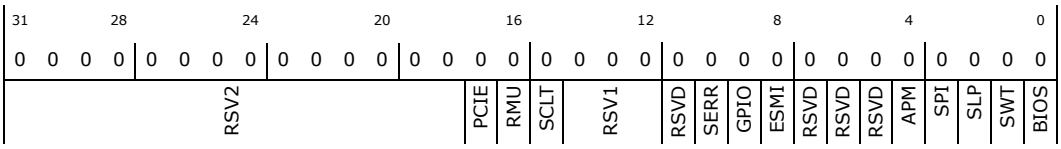
Access Method

Type: I/O Register
(Size: 32 bits)

SMIEN: [GPE0BLK] + 10h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:18	0b RO	**Reserved (RSV2):** Reserved.
17	0b RW	**PCIe Enable (PCIE):** When set enables SMISTS.PCIE to generate SMI.
16	0b RW	**Remote Management Unit Enable (RMU):** When set enables SMISTS.RMU to generate SMI
15	0b RW	**Device Enable (SCLT):** When set enables SMISTS.SCLT to generate SMI.
14:12	0b RO	**Reserved (RSV1):** Reserved.
11	0b RO	**Reserved (RSVD):** Reserved.
10	0b RW	**SERR Enable (SERR):** When set enables SMISTS.SERR to generate SMI#
9	0b RW	**GPIO Enable (GPIO):** When set enables SMISTS.GPIO to generate SMI#
8	0b RW	**External SMI Enable (ESMI):** When set enables SMISTS.ESMI to generate SMI#
7	0b RO	**Reserved (RSVD):** Reserved.
6	0b RO	**Reserved (RSVD):** Reserved.
5	0b RO	**Reserved (RSVD):** Reserved.
4	0b RW	**APM Enable (APM):** When set enables SMISTS.APM to generate SMI#
3	0b RW	**SPI Enable (SPI):** When set enables SMISTS.SPI to generate SMI#
2	0b RW	**Sleep (SLP):** When set enables SMISTS.SLP to generate SMI#
1	0b RW	**Software Timer (SWT):** When set enables SMISTS.SWT to generate SMI#
0	0b RW	**BIOS:** When set enables SMISTS.BIOS to generate SMI#

21.5.2.4 SMI Status Register (SMISTS)—Offset 14h

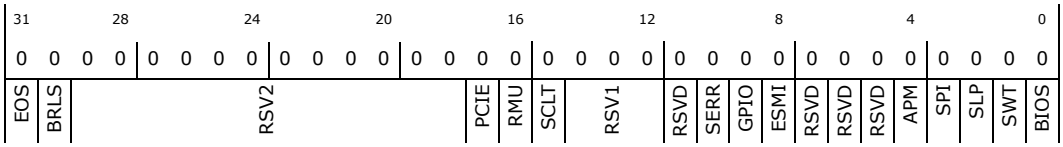
Access Method

Type: I/O Register
(Size: 32 bits)

SMISTS: [GPE0BLK] + 14h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**End of SMI (EOS):** This bit is present only in the SMI Status register and not in SMI Enable register. When set, the Legacy Bridge de-asserts SMI#. Cleared when the Legacy Bridge asserts SMI#.
30	0b WO	**BIOS Release (BRLS):** This bit is present only in the SMI Status register and not in SMI Enable register. Causes SCI to be generated by the Legacy Bridge. Always reads 0.
29:18	0b RO	**Reserved (RSV2):** Reserved.
17	0b RW/1C	**PCIe Status (PCIE):** Set when an Assert SMI message from PCIe Controller is received.
16	0b RW/1C	**Remote Management Unit Status (RMU):** Set when an Assert SMI message from the Remote Management Unit is received.
15	0b RW/1C	**Device Status (SCLT):** Set when the SMI# signal from Device:20 or Device:21 goes active.
14:12	0b RO	**Reserved (RSV1):** Reserved.
11	0b RO	**Reserved (RSVD):** Reserved.
10	0b RW/1C	**SERR Status (SERR):** Set when DO_SERR message is received by the Legacy Bridge.
9	0b RW/1C	**GPIO Status (GPIO):** Set when a GPIO configured for SMI goes active.
8	0b RW/1C	**External GPE Status (ESMI):** Set when the SMI_B input signal goes active.
7	0b RO	**Reserved (RSVD):** Reserved.
6	0b RO	**Reserved (RSVD):** Reserved.
5	0b RO	**Reserved (RSVD):** Reserved.
4	0b RW/1C	**APM Status (APM):** Set when a write to SWSMICTL is performed.
3	0b RW/1C	**SPI Status (SPI):** Set when SPI logic is requesting an SMI
2	0b RW/1C	**Sleep (SLP):** Set when a write occurs to PM1C.SLPEN

Bit Range	Default & Access	Description
1	0b RW/1C	**Software Timer (SWT):** Set when the software SMI has expired.
0	0b RW/1C	**BIOS Status (BIOS):** Set when software sets PM1C.GRLS.

21.5.2.5 General Purpose Event Control Register (GPEC)—Offset 18h

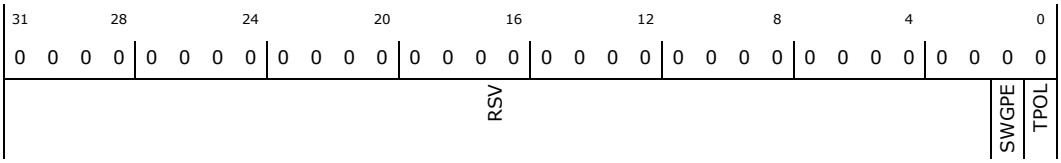
Access Method

Type: I/O Register
(Size: 32 bits)

GPEC: [GPE0BLK] + 18h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1	0b WO	**Software General Purpose Event (SWGPE):** Sets GPE0S.SWGPE when written with 1. This bit always reads back as 0.
0	0b RW	**Thermal Polarity (TPOL):** This bit controls the polarity of THRM_B needed to set GPE0S.THRM. When set, a HIGH value on THRM_B will set GPE0S.THRM. When cleared, a LOW value on THRM_B will set GPE0S.THRM.

21.5.2.6 Power Management Configuration Core Well Register (PMCW)—Offset 28h

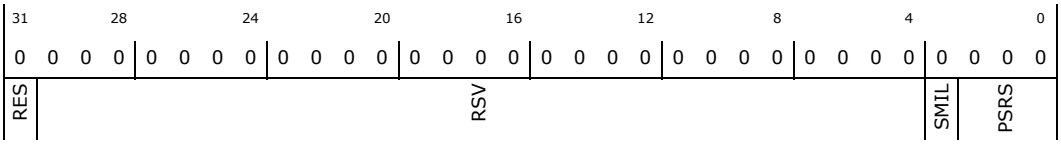
Access Method

Type: I/O Register
(Size: 32 bits)

PMCW: [GPE0BLK] + 28h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW	**Periodic SMI Enable (RES):** When set, an SMIS.SWT will be set by the rate specified by PSRS.
30:4	0b RO	**Reserved (RSV):** Reserved.
3	0b RW/O	**SMI Lock (SMIL):** When set, writes to SMIE have no effect. This bit is only cleared by Core Well reset
2:0	0b RW	**Periodic SMI Rate Selection (PSRS):** Indicates when the timer will time out and cause an SMI#. All values are +/- 30us (RTC Clock). Valid values are: 000b - 1.5ms 001b - 16ms 010b - 32ms 011b - 64ms 100b - 8 sec 101b - 16 sec 110b - 32 sec 111b - 64 sec

21.5.2.7 Power Management Configuration Suspend Well Register (PMSW)—Offset 2Ch

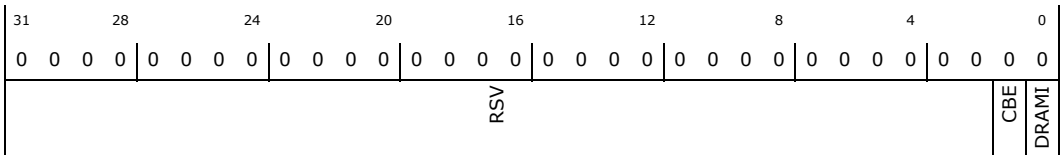
Access Method

Type: I/O Register (Size: 32 bits)

PMSW: [GPE0BLK] + 2Ch

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1	0b RW	**CPU BIST Enable (CBE):** CPU BIST enable INIT functionality not supported.
0	0b RW	**DRAM Initialization Scratch pad (DRAMI):** This bit does not affect hardware functionality. It is provided as a BIOS scratchpad bit that is maintained through warm resets.

21.5.2.8 Power Management Configuration RTC Well Register (PMRW)—Offset 30h

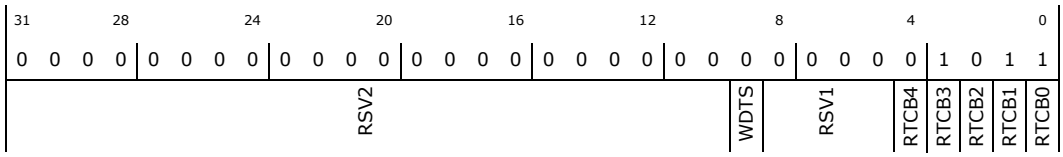
Access Method

Type: I/O Register (Size: 32 bits)

PMRW: [GPE0BLK] + 30h

GPE0BLK Type: PCI Configuration Register (Size: 32 bits)
GPE0BLK Reference: [B:0, D:31, F:0] + 4Ch

Default: 0000000Bh



Bit Range	Default & Access	Description
31:10	0b RO	**Reserved (RSV2):** Reserved.
9	0b RW/1C	**Remote Management Unit Watchdog Trip Status (WDTS):** This bit is set when the Remote Management Unit watchdog timer expires, causing a system shutdown. It is reset by warm and cold resets. It is maintained through the shutdown sequence that is initiated via this trip
8:5	0b RO	**Reserved (RSV1):** Reserved.
4	0b RW	**RTC Bias Resistor 4 (RTCB4):** Adds 192K when de-asserted
3	1b RW	**RTC Bias Resistor 3 (RTCB3):** Adds 96K when de-asserted
2	0b RW	**RTC Bias Resistor 2 (RTCB2):** Adds 48K when de-asserted
1	1b RW	**RTC Bias Resistor 1 (RTCB1):** Adds 24K when de-asserted
0	1b RW	**RTC Bias Resistor 0 (RTCB0):** Adds 12K when de-asserted

21.5.3 ACPI PM1 Block

Table 116. Summary of I/O Registers—PM1BLK

Offset Start	Offset End	Register ID—Description	Default Value
0h	1h	"PM1 Status Register (PM1S)—Offset 0h" on page 827	0000h
2h	3h	"PM1 Enable Register (PM1E)—Offset 2h" on page 828	0000h
4h	7h	"PM1 Control Register (PM1C)—Offset 4h" on page 829	00000000h
8h	Bh	"Power Management 1 Timer Register (PM1T)—Offset 8h" on page 830	00000000h

21.5.3.1 PM1 Status Register (PM1S)—Offset 0h

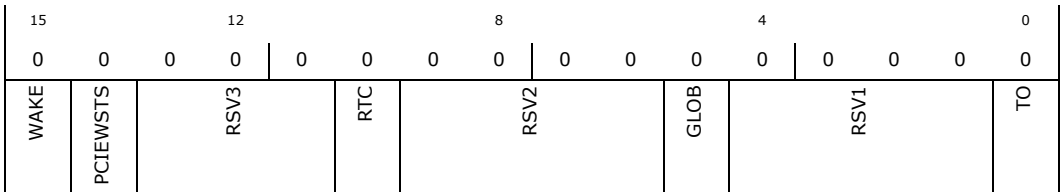
Access Method

Type: I/O Register
(Size: 16 bits)

PM1S: [PM1BLK] + 0h

PM1BLK Type: PCI Configuration Register (Size: 32 bits)
PM1BLK Reference: [B:0, D:31, F:0] + 48h

Default: 0000h



Bit Range	Default & Access	Description
15	0b RW/1C	**Wake Status (WAKE):** Resume Well. This bit is set when the system is in an Sx state and an enable wake event occurs. Upon setting this bit, the Legacy Bridge will transition the system to the S0 state. This bit is not affected by warm resets
14	0b RW/1C	**PCIe Wake Status (PCIEWSTS):** This bit is set by hardware to indicate that the system woke due to a PCI Express wakeup event.
13:11	0b RO	**Reserved (RSV3):** Reserved.
10	0b RW/1C	**RTC Status (RTC):** Resume Well. This bit is set when the RTC asserts IRQ8#, and is not affected by any other enable bit. This bit is not affected by warm resets
9:6	0b RO	**Reserved (RSV2):** Reserved.
5	0b RW/1C	**Global Status (GLOB):** Set when SMIS.BRLS is written to '1'. It always cause an SCI (regardless of PM1C.SCIEN)
4:1	0b RO	**Reserved (RSV1):** Reserved.
0	0b RW/1C	**Timer Overflow Status (TO):** Set anytime bit 22 of PM1T goes low. See PM1E.TO for the effect of this bit being set

21.5.3.2 PM1 Enable Register (PM1E)—Offset 2h

Access Method

Type: I/O Register (Size: 16 bits)

PM1E: [PM1BLK] + 2h

PM1BLK Type: PCI Configuration Register (Size: 32 bits)
PM1BLK Reference: [B:0, D:31, F:0] + 48h

Default: 0000h

Bit Range	Default & Access	Description
15	0b RO	**Reserved (RSV4):** Reserved.
14	0b RW	**PCIe Wake Disable (PWAKED):** This bit disables the inputs to the PCIEWSTS bit for waking the system. Modification of this bit has no impact on the value of the PCIEWSTS bit
13:11	0b RO	**Reserved (RSV3):** Reserved.

Bit Range	Default & Access	Description
10	0b RW	**RTC Enable (RTC):** Resume Well. When set, and PM1S.RTC is set, an SMI#/SCI is generated. This bit is not cleared by any reset other than RTCRST#, CPU/internal thermal Trip, or internal watchdog trip
9:6	0b RO	**Reserved (RSV2):** Reserved.
5	0b RW	**Global Enable (GLOB):** When this bit and PM1S.GLOB are set, SMI#/SCI is generated
4:1	0b RO	**Reserved (RSV1):** Reserved.
0	0b RW	**Timer Overflow Enable (TO):** When set, and PM1S.TO is set, an SMI#/SCI is generated

21.5.3.3 PM1 Control Register (PM1C)—Offset 4h

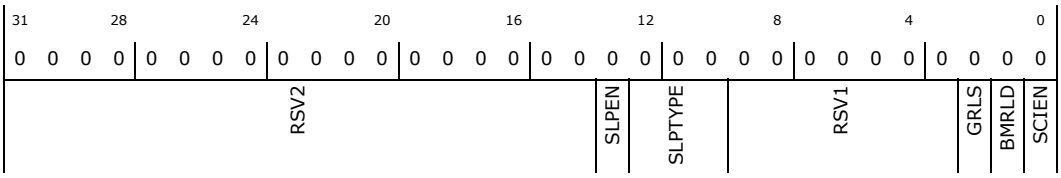
Access Method

Type: I/O Register
(Size: 32 bits)

PM1C: [PM1BLK] + 4h

PM1BLK Type: PCI Configuration Register (Size: 32 bits)
PM1BLK Reference: [B:0, D:31, F:0] + 48h

Default: 00000000h



Bit Range	Default & Access	Description
31:14	0b RO	**Reserved (RSV2):** Reserved.
13	0b WO	**Sleep Enable (SLPEN):** Reads to this bit always return 0. Setting this bit causes the system to sequence into the Sleep state defined by SLPTYP
12:10	0b RW	**Sleep Type (SLPTYPE):** Resume Well. This field defines the type of sleep the system should enter when SLPEN is set. These bits are reset by RTCRST#. 000b - S0 - On 101b - S3 - Suspend to RAM 110b - S4 - Suspend to Disk 111b - S5 - Soft Off All other values are reserved
9:3	0b RO	**Reserved (RSV1):** Reserved.
2	0b WO	**Global Release (GRLS):** Sets SMIS.BIOS when written to 1. This bit always reads as 0
1	0b RW	**Bus Master Reload (BMRLD):** This is treated as a scratchpad bit and has no functionality
0	0b RW	**SCI Enable (SCIEN):** When set, events in GPE0_BLK generate SCI. When cleared, events generate SMI#

21.5.3.4 Power Management 1 Timer Register (PM1T)—Offset 8h

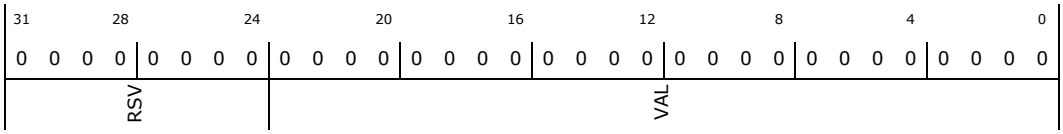
Access Method

Type: I/O Register
(Size: 32 bits)

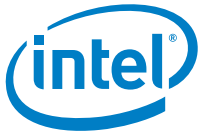
PM1T: [PM1BLK] + 8h

PM1BLK Type: PCI Configuration Register (Size: 32 bits)
PM1BLK Reference: [B:0, D:31, F:0] + 48h

Default: 00000000h



Bit Range	Default & Access	Description
31:24	0b RO	**Reserved (RSV):** Reserved.
23:0	0b RO	**Timer Value (VAL):** Returns the running count of the PM timer. This counter runs off a 3.579545 MHz clock (derived from 14.31818 MHz divided by 4). It is reset on a platform reset, and runs continuously in S0. Any time bit 22 goes from 1 to 0, PM1S.TO is set



21.6 Legacy GPIO

The SoC provides a total of 16 GPIOs. Ten of these GPIOs are available for use during the S0 ACPI state. The remaining 6 GPIOs are available for use during both the S0 and S3 ACPI state. The GPIOs are split between the Legacy Bridge (0/31/0) and the GPIO Controller (0/21/2). The GPIOs within the Legacy Bridge are referred to as Legacy GPIOs and are described in this section.

21.6.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 117. Legacy GPIO Signals

Signal Name	Direction/ Type	Description
GPIO[9:8]	I/O Varies	These Legacy GPIO pins are powered and active in S0 only.
GPIO_SUS[5:0]	I/O Varies	These Legacy GPIO pins are powered and active in S3 and S0.

21.6.2 Features

GPIOs can generate general purpose events (GPEs) on rising and/or falling edges.

The suspend well GPIOs, GPIO_SUS[5:0], can be used to generate wake events when the system is in the ACPI S3 state.

21.6.3 Use

Each GPIO has six registers that control how it is used, or report its status:

- Use Select
- I/O Select
- GPIO Level
- Trigger Positive Edge
- Trigger Negative Edge
- Trigger Status

The Use Select register selects a GPIO pin as a GPIO, or leaves it as its programmed function. This register must be set for all other registers to affect the GPIO.

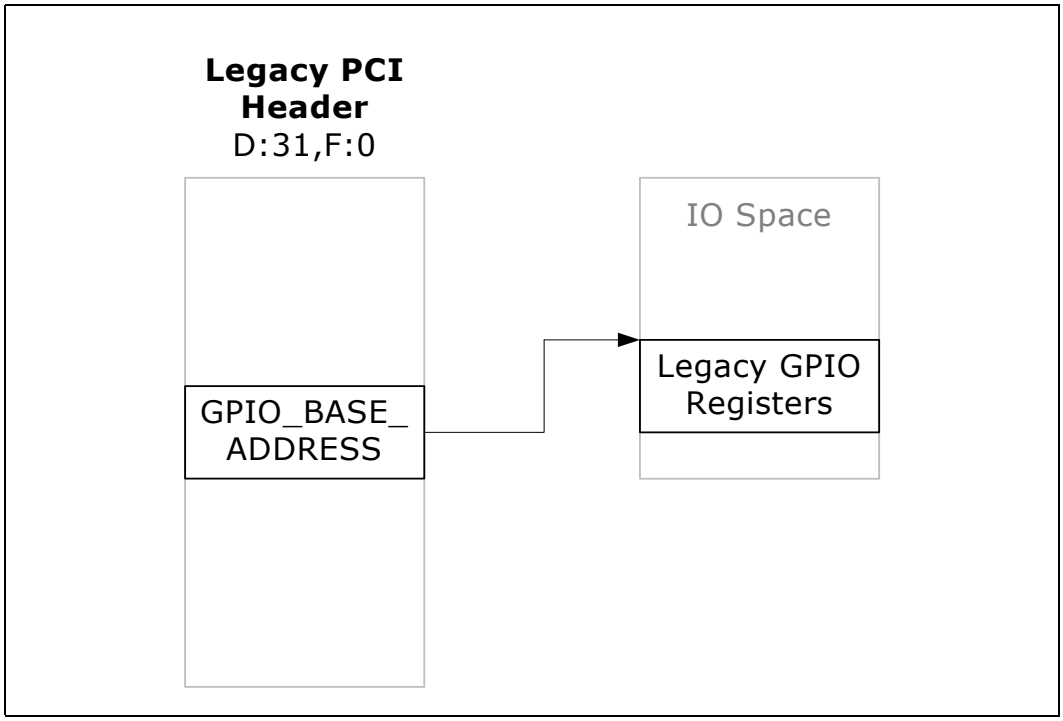
The I/O Select register determines the direction of the GPIO.

The Trigger Positive Edge and Trigger Negative Edge registers enable general purpose events on a rising and falling edge respectively. This only applies to GPIOs set as input.

The Trigger Status register is used by software to determine if the GPIO triggered a GPE. This only applies to GPIOs set as input and with one or both of the Trigger modes enabled.

21.6.4 Register Map

Figure 46. Legacy GPIO Register Map



21.6.5 IO Mapped Registers

Table 118. Summary of I/O Registers—GBA

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"Core Well GPIO Enable (CGEN)—Offset 0h" on page 833	00000003h
4h	7h	"Core Well GPIO Input/Output Select (CGIO)—Offset 4h" on page 833	00000003h
8h	Bh	"Core Well GPIO Level for Input or Output (CGLVL)—Offset 8h" on page 834	00000000h
Ch	Fh	"Core Well GPIO Trigger Positive Edge Enable (CGTPE)—Offset Ch" on page 834	00000000h
10h	13h	"Core Well GPIO Trigger Negative Edge Enable (CGTNE)—Offset 10h" on page 834	00000000h
14h	17h	"Core Well GPIO GPE Enable (CGGPE)—Offset 14h" on page 835	00000000h
18h	1Bh	"Core Well GPIO SMI Enable (CGSMI)—Offset 18h" on page 835	00000000h
1Ch	1Fh	"Core Well GPIO Trigger Status (CGTS)—Offset 1Ch" on page 836	00000000h
20h	23h	"Resume Well GPIO Enable (RGEN)—Offset 20h" on page 836	0000003Fh
24h	27h	"Resume Well GPIO Input/Output Select (RGIO)—Offset 24h" on page 837	0000003Fh
28h	2Bh	"Resume Well GPIO Level for Input or Output (RGLVL)—Offset 28h" on page 837	00000000h
2Ch	2Fh	"Resume Well GPIO Trigger Positive Edge Enable (RGTPE)—Offset 2Ch" on page 837	00000000h

Table 118. Summary of I/O Registers—GBA (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
30h	33h	"Resume Well GPIO Trigger Negative Edge Enable (RGTNE)—Offset 30h" on page 838	00000000h
34h	37h	"Resume Well GPIO GPE Enable (RGGPE)—Offset 34h" on page 838	00000000h
38h	3Bh	"Resume Well GPIO SMI Enable (RGSMI)—Offset 38h" on page 839	00000000h
3Ch	3Fh	"Resume Well GPIO Trigger Status (RGTS)—Offset 3Ch" on page 839	00000000h
40h	43h	"Core Well GPIO NMI Enable (CGNMIEN)—Offset 40h" on page 840	00000000h
44h	47h	"Resume Well GPIO NMI Enable (RGNMIEN)—Offset 44h" on page 840	00000000h

21.6.5.1 Core Well GPIO Enable (CGEN)—Offset 0h

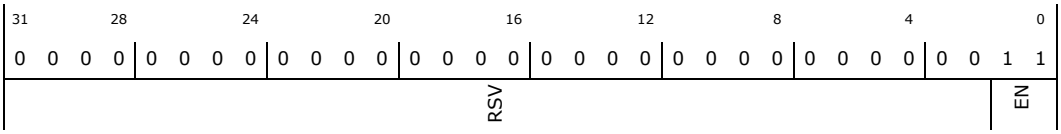
Access Method

Type: I/O Register
(Size: 32 bits)

CGEN: [GBA] + 0h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000003h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	11b RW	**Enable (EN):** When set, enables the pin as a GPIO.

21.6.5.2 Core Well GPIO Input/Output Select (CGIO)—Offset 4h

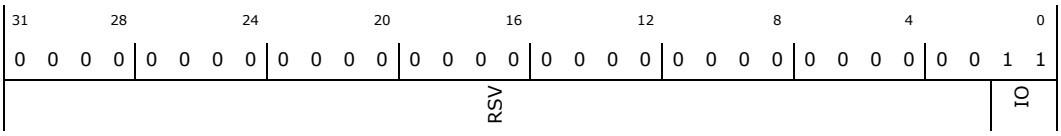
Access Method

Type: I/O Register
(Size: 32 bits)

CGIO: [GBA] + 4h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000003h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	11b RW	**Input/output (IO):** When set, the GPIO signal (if enabled) is programmed as an input. When cleared, the GPIO signal is programmed as an output.

21.6.5.3 Core Well GPIO Level for Input or Output (CGLVL)—Offset 8h

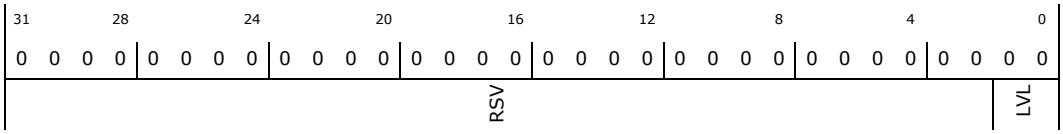
Access Method

Type: I/O Register
(Size: 32 bits)

CGLVL: [GBA] + 8h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Level (LVL):** If the GPIO is programmed to be an output (CGIO.IO[n] cleared), then this bit is used by software to drive a value on the pin. 1 = high, 0 = low. If the GPIO is programmed as an input, then this bit reflects the state of the input signal (1 = high, 0 = low.) and writes will have no effect. The value of this bit has no meaning if the GPIO is disabled (CGEN.EN[n] = 0).

21.6.5.4 Core Well GPIO Trigger Positive Edge Enable (CGTPE)—Offset Ch

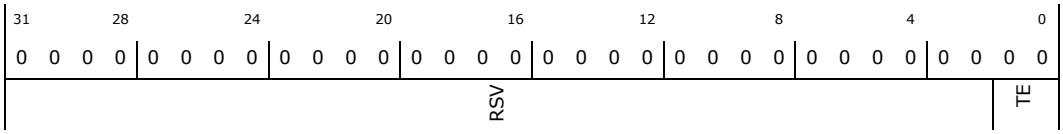
Access Method

Type: I/O Register
(Size: 32 bits)

CGTPE: [GBA] + Ch

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Trigger Enable (TE):** When set, the corresponding GPIO, if enabled as input via CGIO.IO[n], will cause an NMI/SMI/SCI when a 0 to 1 transition occurs. When cleared, the GPIO is not enabled to trigger an NMI/SMI/SCI on a 0 to 1 transition. This bit has no meaning if CGIO.IO[n] is cleared (i.e. programmed for output)

21.6.5.5 Core Well GPIO Trigger Negative Edge Enable (CGTNE)—Offset 10h

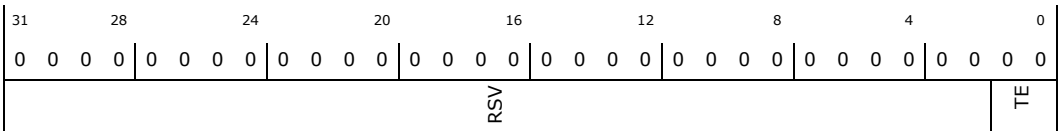
Access Method

Type: I/O Register
(Size: 32 bits)

CGTNE: [GBA] + 10h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Trigger Enable (TE):** When set, the corresponding GPIO, if enabled as input via CGIO.IO[n], will cause an NMI/SMI/SCI when a 1 to 0 transition occurs. When cleared, the GPIO is not enabled to trigger an NMI/SMI/SCI on a 1 to 0 transition. This bit has no meaning if CGIO.IO[n] is cleared (i.e. programmed for output)

21.6.5.6 Core Well GPIO GPE Enable (CGGPE)—Offset 14h

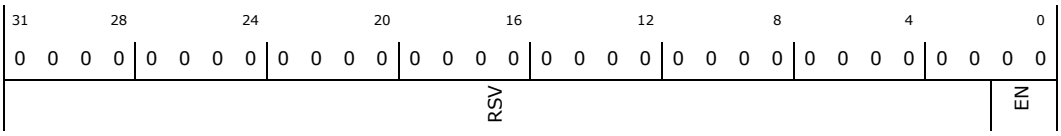
Access Method

Type: I/O Register
(Size: 32 bits)

CGGPE: [GBA] + 14h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an SCI and bit 14 of GPE0 Status register of GPE0 Block will be set.

21.6.5.7 Core Well GPIO SMI Enable (CGSMI)—Offset 18h

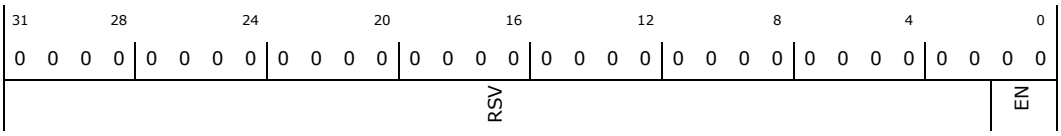
Access Method

Type: I/O Register
(Size: 32 bits)

CGSMI: [GBA] + 18h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an SMI and bit 9 of SMI Status register of GPE0 Block will be set.

21.6.5.8 Core Well GPIO Trigger Status (CGTS)—Offset 1Ch

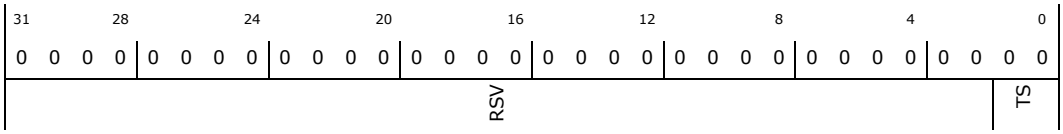
Access Method

Type: I/O Register
(Size: 32 bits)

CGTS: [GBA] + 1Ch

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW/1C	**Trigger Status (TS):** When set, the corresponding GPIO, if enabled as input via CGIO.IO[n], triggered an SMI/SCI/NMI. This will be set if a 0 to 1 transition occurred and CGTPE.TE[n] was set, or a 1 to 0 transition occurred and CGTNE.TE[n] was set. If both CGTPE.TE[n] and CGTNE.TE[n] are set, then this bit will be set on both a 0 to 1 and a 1 to=0 transition. This bit will not be set if the GPIO is configured as an output.

21.6.5.9 Resume Well GPIO Enable (RGEN)—Offset 20h

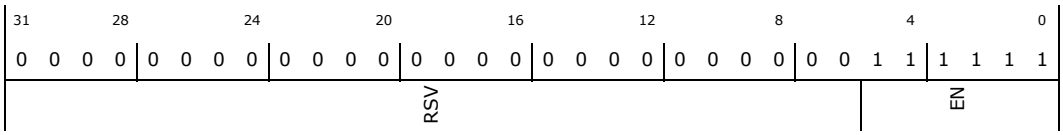
Access Method

Type: I/O Register
(Size: 32 bits)

RGEN: [GBA] + 20h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 0000003Fh



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	3Fh RW	**Enable (EN):** When set, enables the pin as a GPIO.

21.6.5.10 Resume Well GPIO Input/Output Select (RGIO)—Offset 24h

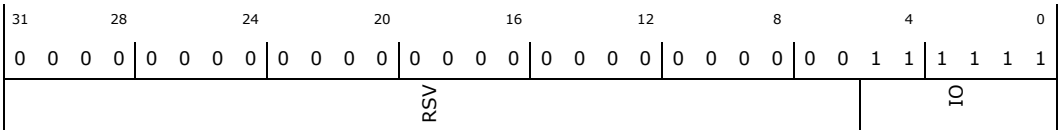
Access Method

Type: I/O Register
(Size: 32 bits)

RGIO: [GBA] + 24h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 0000003Fh



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	3Fh RW	**Input/Output (IO):** When set, the GPIO signal (if enabled) is programmed as an input. When cleared, the GPIO signal is programmed as an output.

21.6.5.11 Resume Well GPIO Level for Input or Output (RGLVL)—Offset 28h

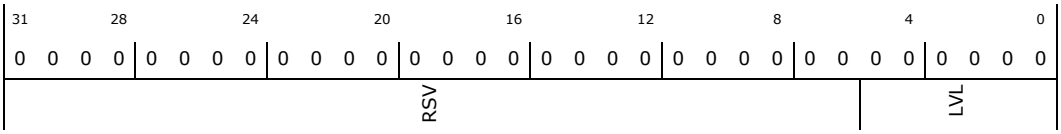
Access Method

Type: I/O Register
(Size: 32 bits)

RGLVL: [GBA] + 28h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Level (LVL):** If the GPIO is programmed to be an output (RGIO.IO[n] cleared), then this bit is used by software to drive a value on the pin. 1 = high, 0 = low. If the GPIO is programmed as an input, then this bit reflects the state of the input signal (1 = high, 0 = low.) and writes will have no effect. The value of this bit has no meaning if the GPIO is disabled (RGEN.EN[n] = 0).

21.6.5.12 Resume Well GPIO Trigger Positive Edge Enable (RGTPE)—Offset 2Ch

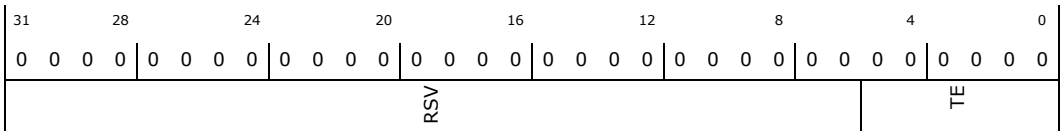
Access Method

Type: I/O Register
(Size: 32 bits)

RGTPE: [GBA] + 2Ch

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Trigger Enable (TE):** When set, the corresponding GPIO, if enabled as input via RGIO.IO[n], will cause an NMI/SMI/SCI when a 0 to 1 transition occurs. When cleared, the GPIO is not enabled to trigger an NMI/SMI/SCI on a 0 to 1 transition. This bit has no meaning if RGIO.IO[n] is cleared (i.e. programmed for output)

21.6.5.13 Resume Well GPIO Trigger Negative Edge Enable (RGTNE)—Offset 30h

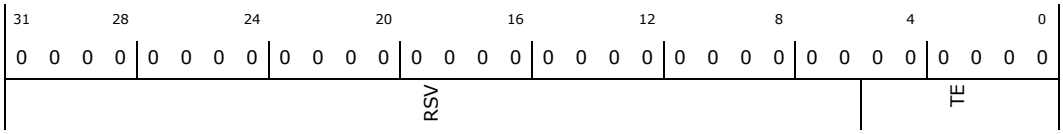
Access Method

Type: I/O Register
(Size: 32 bits)

RGTNE: [GBA] + 30h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Trigger Enable (TE):** When set, the corresponding GPIO, if enabled as input via RGIO.IO[n], will cause an NMI/SMI/SCI when a 1 to 0 transition occurs. When cleared, the GPIO is not enabled to trigger an NMI/SMI/SCI on a 1 to 0 transition. This bit has no meaning if RGIO.IO[n] is cleared (i.e. programmed for output)

21.6.5.14 Resume Well GPIO GPE Enable (RGGPE)—Offset 34h

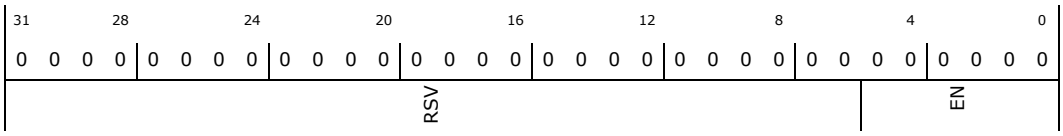
Access Method

Type: I/O Register
(Size: 32 bits)

RGGPE: [GBA] + 34h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an SCI and bit 14 of GPE0 Status register of GPE0 Block will be set.

21.6.5.15 Resume Well GPIO SMI Enable (RGSMI)—Offset 38h

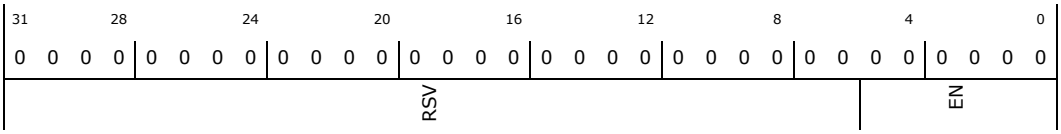
Access Method

Type: I/O Register
(Size: 32 bits)

RGSMI: [GBA] + 38h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an SMI and bit 9 of SMI Status register of GPE0 Block will be set.

21.6.5.16 Resume Well GPIO Trigger Status (RGTS)—Offset 3Ch

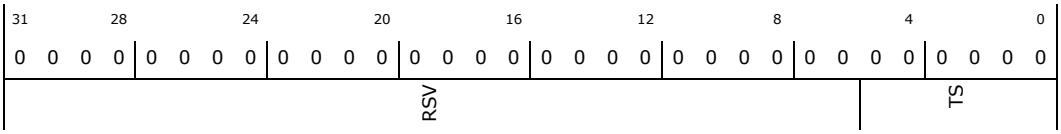
Access Method

Type: I/O Register
(Size: 32 bits)

RGTS: [GBA] + 3Ch

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW/1C	**Trigger Status (TS):** When set, the corresponding GPIO, if enabled as input via RGIO.IO[n], triggered an SMI/SCI/NMI. This will be set if a 0 to 1 transition occurred and RGTPE.TE[n] was set, or a 1 to 0 transition occurred and RGTNE.TE[n] was set. If both RGTPE.TE[n] and RGTNE.TE[n] are set, then this bit will be set on both a 0 to 1 and a 1 to=0 transition. This bit will not be set if the GPIO is configured as an output.

21.6.5.17 Core Well GPIO NMI Enable (CGNMIEN)—Offset 40h

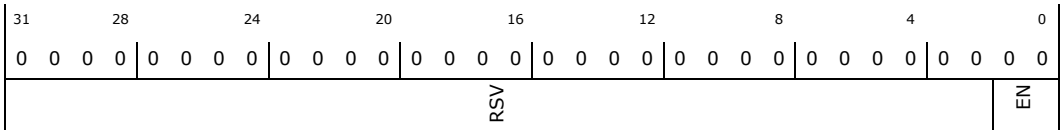
Access Method

Type: I/O Register
(Size: 32 bits)

CGNMIEN: [GBA] + 40h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1:0	0b RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an NMI.

21.6.5.18 Resume Well GPIO NMI Enable (RGNMIEN)—Offset 44h

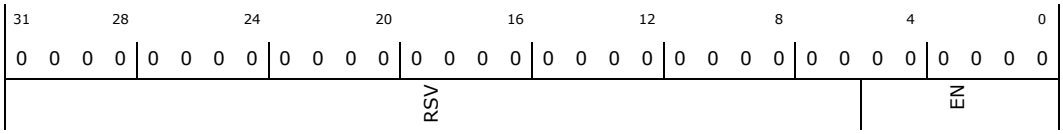
Access Method

Type: I/O Register
(Size: 32 bits)

RGNMIEN: [GBA] + 44h

GBA Type: PCI Configuration Register (Size: 32 bits)
GBA Reference: [B:0, D:31, F:0] + 44h

Default: 00000000h



Bit Range	Default & Access	Description
31:6	0b RO	**Reserved (RSV):** Reserved.
5:0	0h RW	**Enable (EN):** When set, the corresponding GPIO, is enabled to generate an NMI.

21.7 Legacy SPI Controller

The Legacy SPI Controller provides an interface to a SPI Flash device that contains the SoC firmware.

21.7.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 119. Legacy SPI Signals

Signal Name	Direction/ Type	Description
LSPI_MOSI	O	**Legacy SPI Data Output**
LSPI_MISO	I	**Legacy SPI Data Input**
LSPI_SS_B	O	**Legacy Chip Select Signal**
LPSI_SCK	O	**Legacy SPI Clock Output**

21.7.2 Features

The Legacy SPI Controller provides access to system firmware that resides on a SPI Flash device connected to the 4-pin Legacy SPI interface. SPI Flash devices up to 16 MByte in size are supported. A SPI clock frequency of 20 MHz is supported.

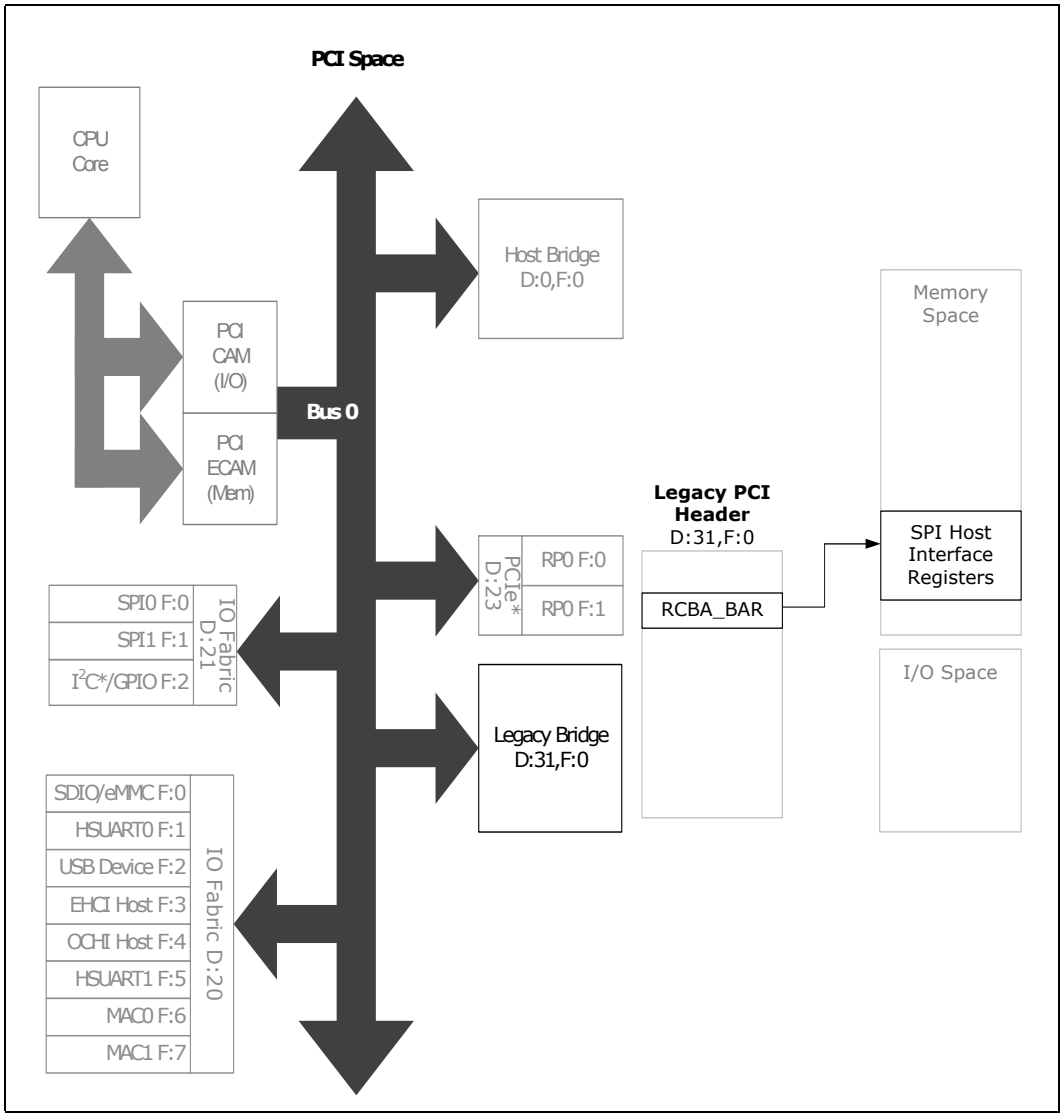
The Legacy SPI Controller supports direct memory reads from the processor. All other operations are controlled via the SPI Host Interface registers that reside in the RCRB Memory Space in the range 3020h to 308Fh.

To protect the integrity of system firmware, the Legacy SPI Controller provides two write protection mechanisms, one scheme based on address ranges and one SMI#-based scheme. If either mechanism indicates an access should not be allowed, then that write access is blocked.

21.7.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 47. Legacy SPI Register Map



21.7.4 Legacy SPI Host Interface Registers

The SPI Host Interface registers are memory-mapped in the RCRB Memory Space in offset range 3020h to 308Fh.

Warning: Address locations that are not listed are considered reserved register locations. Reads to reserved registers may return non-zero values. Writes to reserved locations may cause system failure.

Table 120. Summary of Memory Mapped I/O Registers—RCBA

Offset Start	Offset End	Register ID—Description	Default Value
3020h	3021h	"SPI Status (SPISTS)—Offset 3020h" on page 843	0001h
3022h	3023h	"SPI Control (SPICTL)—Offset 3022h" on page 844	4001h
3024h	3027h	"SPI Address (SPIADDR)—Offset 3024h" on page 845	00000000h
3028h	302Bh	"SPI Data 0 - Lower 32 Bits (SPID0_1)—Offset 3028h" on page 846	00000000h
302Ch	302Fh	"SPI Data 0 - Upper 32 Bits (SPID0_2)—Offset 302Ch" on page 846	00000000h
3030h	3033h	"SPI Data 1 - Lower 32 Bits (SPID1_1)—Offset 3030h" on page 846	00000000h
3034h	3037h	"SPI Data 1 - Upper 32 Bits (SPID1_2)—Offset 3034h" on page 847	00000000h
3038h	303Bh	"SPI Data 2 - Lower 32 Bits (SPID2_1)—Offset 3038h" on page 847	00000000h
303Ch	303Fh	"SPI Data 2 - Upper 32 Bits (SPID2_2)—Offset 303Ch" on page 847	00000000h
3040h	3043h	"SPI Data 3 - Lower 32 Bits (SPID3_1)—Offset 3040h" on page 848	00000000h
3044h	3047h	"SPI Data 3 - Upper 32 Bits (SPID3_2)—Offset 3044h" on page 848	00000000h
3048h	304Bh	"SPI Data 4 - Lower 32 Bits (SPID4_1)—Offset 3048h" on page 848	00000000h
304Ch	304Fh	"SPI Data 4 - Upper 32 Bits (SPID4_2)—Offset 304Ch" on page 849	00000000h
3050h	3053h	"SPI Data 5 - Lower 32 Bits (SPID5_1)—Offset 3050h" on page 849	00000000h
3054h	3057h	"SPI Data 5 - Upper 32 Bits (SPID5_2)—Offset 3054h" on page 850	00000000h
3058h	305Bh	"SPI Data 6 - Lower 32 Bits (SPID6_1)—Offset 3058h" on page 850	00000000h
305Ch	305Fh	"SPI Data 6 - Upper 32 Bits (SPID6_2)—Offset 305Ch" on page 850	00000000h
3060h	3063h	"SPI Data 7 - Lower 32 Bits (SPID7_1)—Offset 3060h" on page 851	00000000h
3064h	3067h	"SPI Data 7 - Upper 32 Bits (SPID7_2)—Offset 3064h" on page 851	00000000h
3070h	3073h	"BIOS Base Address (BBAR)—Offset 3070h" on page 851	00000000h
3074h	3075h	"Prefix Opcode Configuration (PREOP)—Offset 3074h" on page 852	0004h
3076h	3077h	"Opcode Type Configuration (OPTYPE)—Offset 3076h" on page 853	0000h
3078h	307Bh	"Opcode Menu Configuration - Lower 32 Bits (OPMENU_1)—Offset 3078h" on page 853	00000005h
307Ch	307Fh	"Opcode Menu Configuration - Upper 32 Bits (OPMENU_2)—Offset 307Ch" on page 854	00000000h
3080h	3083h	"Protected BIOS Range 0 (PBR0)—Offset 3080h" on page 855	00000000h
3084h	3087h	"Protected BIOS Range 1 (PBR1)—Offset 3084h" on page 855	00000000h
3088h	308Bh	"Protected BIOS Range 2 (PBR2)—Offset 3088h" on page 856	00000000h

21.7.4.1 SPI Status (SPISTS)—Offset 3020h

Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

SPISTS: [RCBA] + 3020h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0001h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
CLD	RSV2											BA	CD	RSV1	CIP

Bit Range	Default & Access	Description
15	0b RW/O	**SPI Configuration Lock-Down (CLD):** When set to 1, the SPI Static Configuration information cannot be overwritten. Once set to 1, this bit can only be cleared by a hardware reset
14:4	0b RO	**Reserved (RSV2):** Reserved.
3	0b RW/1C	**Blocked Access Status (BA):** Hardware sets this bit to 1 when an access is blocked from running on the SPI interface due to one of the protection policies or when any of the programmed cycle registers are written while a programmed access is already in progress. This bit is set for both programmed accesses and direct memory reads that get blocked. This bit remains asserted until cleared by software writing a 1 or hardware reset.
2	0b RW/1C	**Cycle Done Status (CD):** Hardware sets this bit to 1 when the SPI Cycle completes (i.e., SCIP bit is 0) after software sets the GO bit. This bit remains asserted until cleared by software writing a 1 or hardware reset. When this bit is set and the SMI Enable bit in the SPI Control register is set, an internal signal is asserted to the SMI# generation block. Software must make sure this bit is cleared prior to enabling the SPI SMI# assertion for a new programmed access. This bit gets set after the Status Register Polling sequence completes after reset de-asserts. It is cleared before and during that sequence.
1	0b RO	**Reserved (RSV1):** Reserved.
0	01h RO	**Cycle In Progress (CIP):** Hardware sets this bit when software sets the SPI Cycle Go bit in the SPI Control register. This bit remains set until the cycle completes on the SPI interface. Hardware automatically sets and clears this bit so that software can determine when read data is valid and/or when it is safe to begin programming the next command. Software must only program the next command when this bit is 0. This bit reports 1b during the Status Register Polling sequence after reset de-asserts; it is cleared when that sequence completes.

21.7.4.2 SPI Control (SPICTL)—Offset 3022h

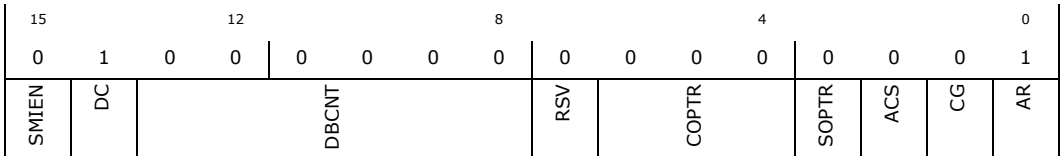
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

SPICTL: [RCBA] + 3022h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 4001h



Bit Range	Default & Access	Description
15	0b RW	**SMI# Enable (SMIEN):** When set to 1, the SPI asserts an SMI# request whenever the Cycle Done Status bit is 1.
14	1b RW	**Data Cycle (DC):** When set to 1, there is data that corresponds to this transaction. When 0, no data is delivered for this cycle, and the DBC and data fields themselves are don't care.
13:8	0b RW	**Data Byte Count (DBCNT):** Data Byte Count: This field specifies the number of bytes to shift in or out during the data portion of the SPI cycle. The valid settings (in decimal) are any value from 0 to 63. The number of bytes transferred is the value of this field plus 1. Note that when this field is 00_0000b, then there is 1 byte to transfer and that 11_1111b means there are 64 bytes to transfer.

Bit Range	Default & Access	Description
7	0b RO	**Reserved (RSV):** Reserved.
6:4	0b RW	**Cycle Opcode Pointer (COPTR):** This field selects one of the programmed opcodes in the Opcode Menu Configuration register to be used as the SPI Command/Opcode. In the case of an Atomic Cycle Sequence, this determines the second command.
3	0b RW	**Sequence Prefix Opcode Pointer (SOPTR):** This field selects one of the two programmed prefix opcodes for use when performing an Atomic Cycle Sequence. A value of 0 points to the opcode in the least significant byte of the Prefix Opcode Configuration register. By making this programmable, the processor supports flash devices that have different opcodes for enabling writes to the data space vs. status register.
2	0b RW	**Atomic Cycle Sequence (ACS):** When set to 1 along with the SCGO assertion, the processor will execute a sequence of commands on the SPI interface without allowing the other SPI master component to arbitrate and interleave cycles. The sequence is composed of: Atomic Sequence Prefix Command (8-bit opcode only) Primary Command specified by software (can include address and data) Polling the Flash Status Register (opcode 05h) until bit 0 becomes 0b. The SPI Cycle in Progress bit remains set and the Cycle Done Status bit in the SPI Status register remains unset until the Busy bit in the Flash Status Register returns 0.
1	0b RW/S	**Cycle Go (CG):** This bit always returns 0 on reads. However, a write to this register with a 1 in this bit starts the SPI cycle defined by the other bits of this register. The SPI Cycle in Progress (SCIP) bit in the SPI Status register gets set by this action. Hardware must ignore writes to this bit while the SPI Cycle In Progress bit is set. Hardware allows other bits in this register to be programmed for the same transaction when writing this bit to 1. This saves an additional memory write.
0	1b RW	**Access Request (AR):** This bit is used by the software to request that the other SPI master stop initiating long transactions on the SPI bus. This bit defaults to a 1 and must be cleared by BIOS after completing the accesses for the boot process.

21.7.4.3 SPI Address (SPIADDR)—Offset 3024h

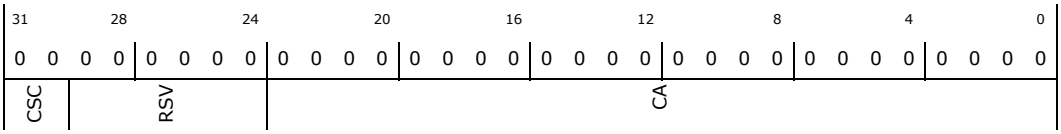
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPIADDR: [RCBA] + 3024h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:30	0b RW	**CSC:** Chip Select Control: These two bits control which SPI Chip Select is used. Default 00 must always select SS0. Direct read mode always uses SS0. 00 : SS0 01 : SS1 10 : SS2 11 : SS3
29:24	0b RO	**Reserved (RSV):** Reserved.
23:0	0b RW	**Cycle Address (CA):** This field is shifted out as the SPI Address (MSB first).

21.7.4.4 SPI Data 0 - Lower 32 Bits (SPID0_1)—Offset 3028h

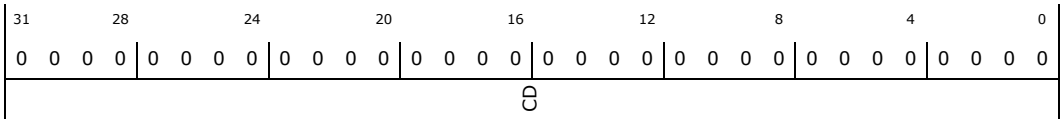
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID0_1: [RCBA] + 3028h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.5 SPI Data 0 - Upper 32 Bits (SPID0_2)—Offset 302Ch

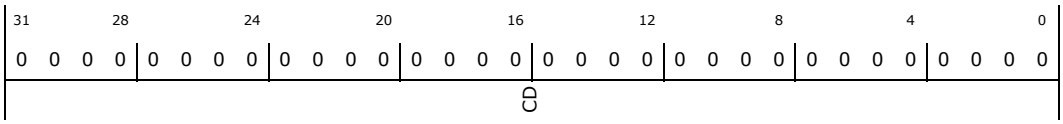
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID0_2: [RCBA] + 302Ch

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.6 SPI Data 1 - Lower 32 Bits (SPID1_1)—Offset 3030h

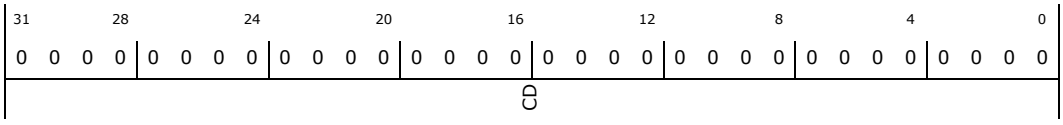
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID1_1: [RCBA] + 3030h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.7 SPI Data 1 - Upper 32 Bits (SPID1_2)—Offset 3034h

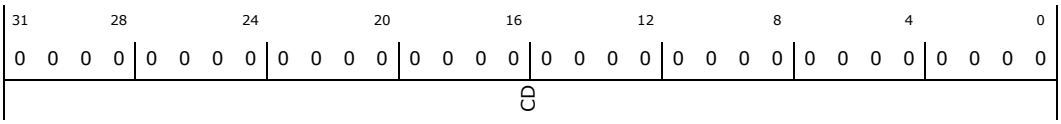
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID1_2: [RCBA] + 3034h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.8 SPI Data 2 - Lower 32 Bits (SPID2_1)—Offset 3038h

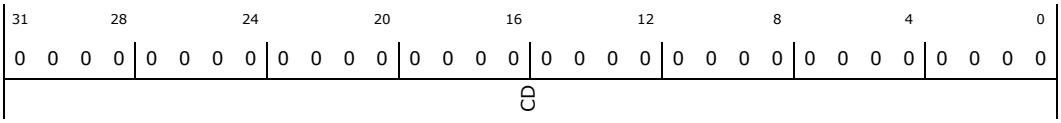
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID2_1: [RCBA] + 3038h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.9 SPI Data 2 - Upper 32 Bits (SPID2_2)—Offset 303Ch

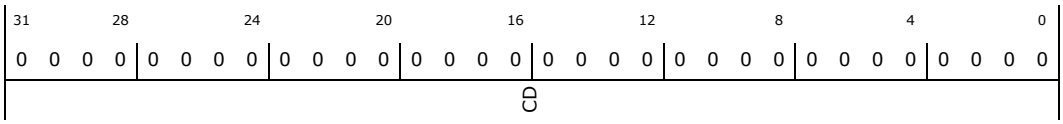
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID2_2: [RCBA] + 303Ch

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.10 SPI Data 3 - Lower 32 Bits (SPID3_1)—Offset 3040h

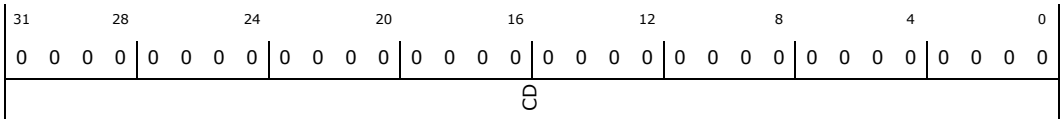
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID3_1: [RCBA] + 3040h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.11 SPI Data 3 - Upper 32 Bits (SPID3_2)—Offset 3044h

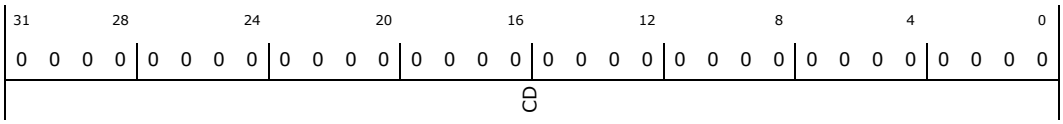
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID3_2: [RCBA] + 3044h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.12 SPI Data 4 - Lower 32 Bits (SPID4_1)—Offset 3048h

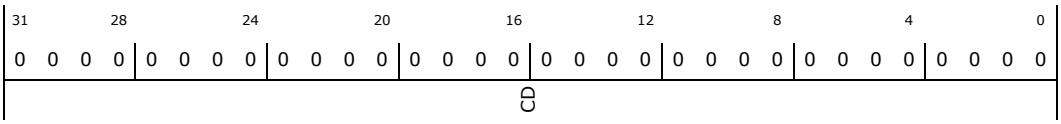
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID4_1: [RCBA] + 3048h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.13 SPI Data 4 - Upper 32 Bits (SPID4_2)—Offset 304Ch

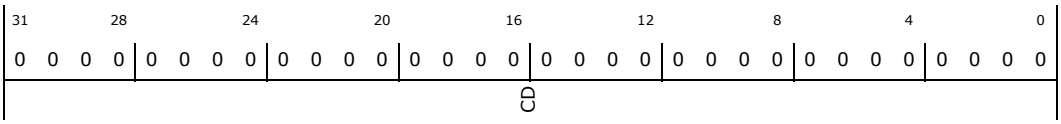
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID4_2: [RCBA] + 304Ch

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.14 SPI Data 5 - Lower 32 Bits (SPID5_1)—Offset 3050h

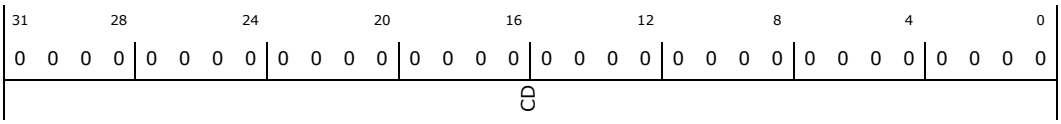
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID5_1: [RCBA] + 3050h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.15 SPI Data 5 - Upper 32 Bits (SPID5_2)—Offset 3054h

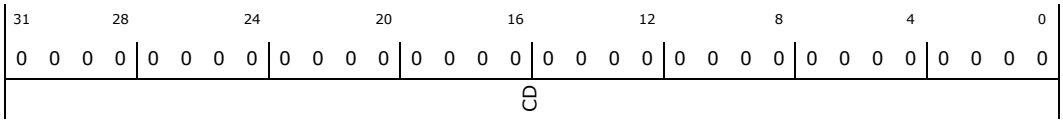
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID5_2: [RCBA] + 3054h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.16 SPI Data 6 - Lower 32 Bits (SPID6_1)—Offset 3058h

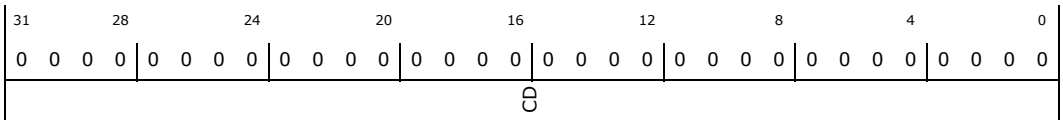
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID6_1: [RCBA] + 3058h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.17 SPI Data 6 - Upper 32 Bits (SPID6_2)—Offset 305Ch

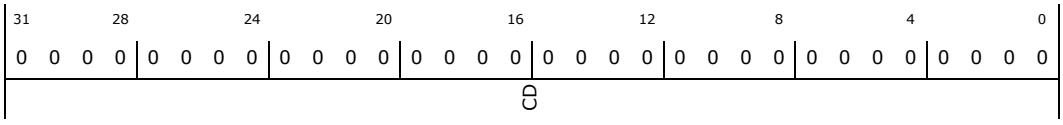
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID6_2: [RCBA] + 305Ch

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.18 SPI Data 7 - Lower 32 Bits (SPID7_1)—Offset 3060h

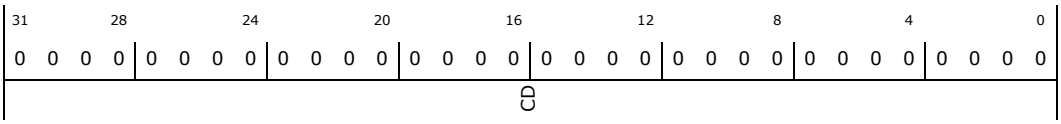
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID7_1: [RCBA] + 3060h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.19 SPI Data 7 - Upper 32 Bits (SPID7_2)—Offset 3064h

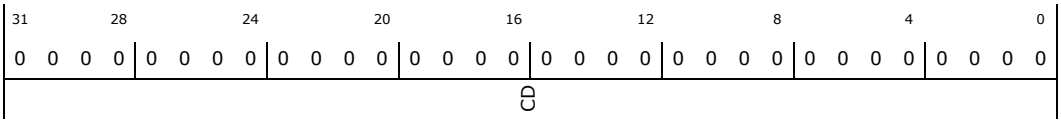
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

SPID7_2: [RCBA] + 3064h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Cycle Data (CD):** This field is shifted out during the data portion of the SPI cycle. This register also shifts in the data during the data portion of the SPI cycle

21.7.4.20 BIOS Base Address (BBAR)—Offset 3070h

This register is not writable when the SPI Configuration Lock-Down bit in the SPI Status register is set.

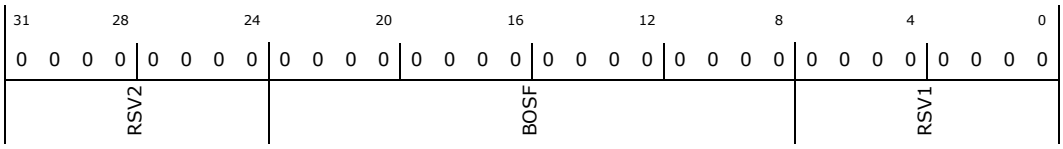
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

BBAR: [RCBA] + 3070h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:24	0b RO	**Reserved (RSV2):** Reserved.
23:8	0b RW/L	**Bottom of System Flash (BOSF):** This field determines the bottom of the System BIOS. The processor will not run Programmed commands nor memory reads whose address field is less than this value. This field corresponds to bits 23:8 of the 3-byte address; bits 7:0 are assumed to be 00h for this vector when comparing to a potential SPI address. Software must always program 1s into the upper, Don't Care bits of this field based on the flash size. Hardware does not know the size of the flash array and relies upon the correct programming by software. The default value of 0000h results in all cycles allowed. Note: The SPI Host Controller prevents any Programmed cycle using the Address Register with an address less than the value in this register. Some flash devices specify that the Read ID command must have an address of 0000h or 0001h. If this command must be supported with these devices, it must be performed with the BBAR - BIOS Base Address programmed to 0h. Some of these devices have actually been observed to ignore the upper address bits of the Read ID command.
7:0	0b RO	**Reserved (RSV1):** Reserved.

21.7.4.21 Prefix Opcode Configuration (PREOP)—Offset 3074h

This register is not writable when the SPI Configuration Lock-Down bit in the SPI Status register is set.

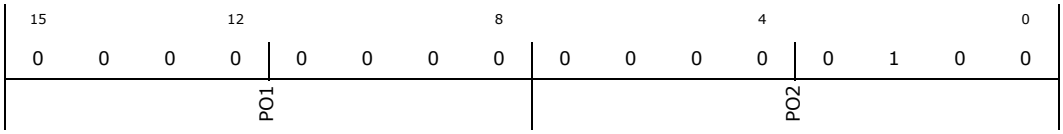
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

PREOP: [RCBA] + 3074h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0004h



Bit Range	Default & Access	Description
15:8	0h RW/L	**Prefix Opcode 1 (PO1):** Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.
7:0	04h RW/L	**Prefix Opcode 2 (PO2):** Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.

21.7.4.22 Opcode Type Configuration (OPTYPE)—Offset 3076h

This register is not writable when the SPI Configuration Lock-Down bit in the SPI Status register is set. Entries in this register correspond to the entries in the Opcode Menu Configuration register. Note that the definition below only provides write protection for opcodes that have addresses associated with them. Therefore, any erase or write opcodes that do not use an address should be avoided (for example, Chip Erase. and Auto-Address Increment Byte Program.).

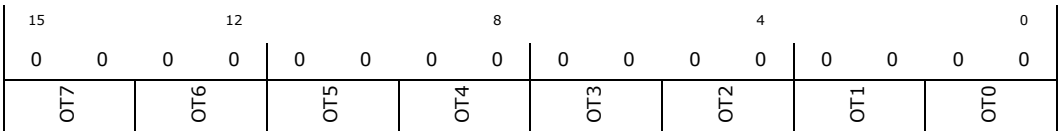
Access Method

Type: Memory Mapped I/O Register (Size: 16 bits)

OPTYPE: [RCBA] + 3076h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 0000h



Bit Range	Default & Access	Description
15:14	0b RW/L	**Opcode Type 7 (OT7):** See the description for bits 1:0
13:12	0b RW/L	**Opcode Type 6 (OT6):** See the description for bits 1:0
11:10	0b RW/L	**Opcode Type 5 (OT5):** See the description for bits 1:0
9:8	0b RW/L	**Opcode Type 4 (OT4):** See the description for bits 1:0
7:6	0b RW/L	**Opcode Type 3 (OT3):** See the description for bits 1:0
5:4	0b RW/L	**Opcode Type 2 (OT2):** See the description for bits 1:0
3:2	0b RW/L	**Opcode Type 1 (OT1):** See the description for bits 1:0
1:0	0b RW/L	**Opcode Type 0 (OT0):** This field specifies information about the corresponding Opcode 0. This information allows the hardware to 1) know whether to use the address field and 2) provide BIOS protection capabilities. The hardware implementation also uses the read vs. write information for modifying the behavior of the SPI interface logic. The encoding of the two bits is: 00 = No Address associated with this Opcode and Read Cycle type 01 = No Address associated with this Opcode and Write Cycle type 10 = Address required; Read cycle type 11 = Address required; Write cycle type

21.7.4.23 Opcode Menu Configuration - Lower 32 Bits (OPMENU_1)—Offset 3078h

This register is not writable when the SPI Configuration Lock-Down bit in the SPI Status register is set. Eight entries are available in this register to give BIOS a sufficient set of commands for communicating with the flash device, while also restricting what malicious software can do. This keeps the hardware flexible enough to operate with a wide variety of SPI devices. It is recommended that BIOS avoid programming Write

Enable opcodes in this menu. Malicious software could then perform writes and erases to the SPI flash without using the atomic cycle mechanism. Write Enable opcodes should only be programmed in the Prefix Opcode Configuration register.

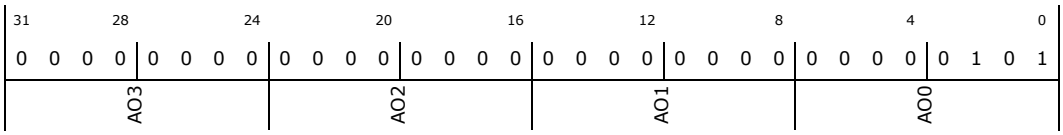
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

OPMENU_1: [RCBA] + 3078h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000005h



Bit Range	Default & Access	Description
31:24	0b RW/L	**Allowable Opcode 3 (AO3):** See the description for bits 7:0
23:16	0b RW/L	**Allowable Opcode 2 (AO2):** See the description for bits 7:0
15:8	0b RW/L	**Allowable Opcode 1 (AO1):** See the description for bits 7:0
7:0	05h RW/L	**Allowable Opcode 0 (AO0):** Software programs an SPI opcode into this field for use when initiating SPI commands through the Control Register.

21.7.4.24 Opcode Menu Configuration - Upper 32 Bits (OPMENU_2)—Offset 307Ch

This register is not writable when the SPI Configuration Lock-Down bit in the SPI Status register is set. Eight entries are available in this register to give BIOS a sufficient set of commands for communicating with the flash device, while also restricting what malicious software can do. This keeps the hardware flexible enough to operate with a wide variety of SPI devices. It is recommended that BIOS avoid programming Write Enable opcodes in this menu. Malicious software could then perform writes and erases to the SPI flash without using the atomic cycle mechanism. Write Enable opcodes should only be programmed in the Prefix Opcode Configuration register.

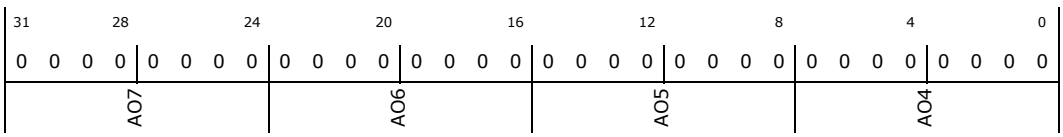
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

OPMENU_2: [RCBA] + 307Ch

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:24	0b RW/L	**Allowable Opcode 7 (AO7):** See the description for bits 7:0
23:16	0b RW/L	**Allowable Opcode 6 (AO6):** See the description for bits 7:0
15:8	0b RW/L	**Allowable Opcode 5 (AO5):** See the description for bits 7:0
7:0	0h RW/L	**Allowable Opcode 4 (AO4):** Software programs an SPI opcode into this field for use when initiating SPI commands through the Control Register.

21.7.4.25 Protected BIOS Range 0 (PBR0)—Offset 3080h

This register cannot be written when the SPI Configuration Lock-Down bit in the SPI Status register is set to 1.

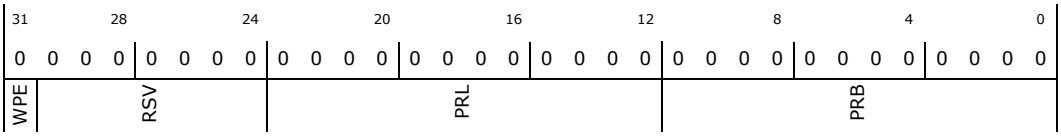
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

PBR0: [RCBA] + 3080h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/L	**Write Protection Enable (WPE):** When set, this bit indicates that the Base and Limit fields in this register are valid and that writes directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:24	0b RO	**Reserved (RSV):** Reserved.
23:12	0b RW/L	**Protected Range Limit (PRL):** This field corresponds to SPI address bits 23:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
11:0	0b RW/L	**Protected Range Base (PRB):** This field corresponds to SPI address bits 23:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

21.7.4.26 Protected BIOS Range 1 (PBR1)—Offset 3084h

This register cannot be written when the SPI Configuration Lock-Down bit in the SPI Status register is set to 1.

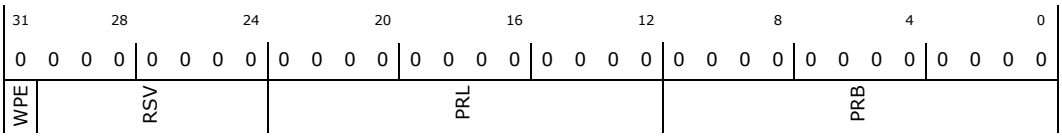
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

PBR1: [RCBA] + 3084h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/L	**Write Protection Enable (WPE):** When set, this bit indicates that the Base and Limit fields in this register are valid and that writes directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:24	0b RO	**Reserved (RSV):** Reserved.
23:12	0b RW/L	**Protected Range Limit (PRL):** This field corresponds to SPI address bits 23:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
11:0	0b RW/L	**Protected Range Base (PRB):** This field corresponds to SPI address bits 23:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

21.7.4.27 Protected BIOS Range 2 (PBR2)—Offset 3088h

This register cannot be written when the SPI Configuration Lock-Down bit in the SPI Status register is set to 1.

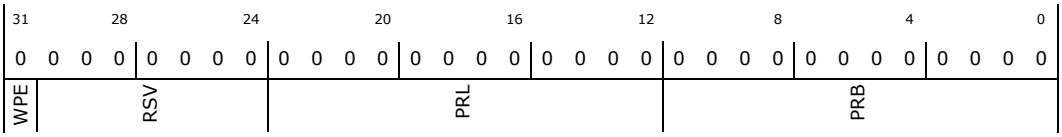
Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

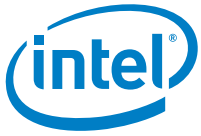
PBR2: [RCBA] + 3088h

RCBA Type: PCI Configuration Register (Size: 32 bits)
RCBA Reference: [B:0, D:31, F:0] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31	0b RW/L	**Write Protection Enable (WPE):** When set, this bit indicates that the Base and Limit fields in this register are valid and that writes directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:24	0b RO	**Reserved (RSV):** Reserved.



Bit Range	Default & Access	Description
23:12	0b RW/L	**Protected Range Limit (PRL):** This field corresponds to SPI address bits 23:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
11:0	0b RW/L	**Protected Range Base (PRB):** This field corresponds to SPI address bits 23:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

21.8 8254 Programmable Interval Timer

The 8254 contains three counters which have fixed uses. All registers are in the core well and clocked by a 14.31818 MHz clock.

21.8.1 Features

21.8.1.1 Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value one counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two, each counter period. The counter negates IRQ0 when the count value reaches 0. It then reloads the initial count value and again decrements the initial count value by two, each counter period. The counter then asserts IRQ0 when the count value reaches 0, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

21.8.1.2 Counter 1, Refresh Request Signal

This counter is programmed for Mode 2 operation and impacts the period of the NSC.RTS (NMI Status and Control Register, bit4, Refresh Cycle Toggle Status). Programming the counter to anything other than Mode 2 results in undefined behavior.

21.8.1.3 Counter 2, Speaker Tone

This counter is typically programmed for Mode 3 operation.

21.8.2 Use

21.8.2.1 Timer Programming

The counter/timers are programmed in the following fashion:

- Write a control word to select a counter
- Write an initial count for that counter.
- Load the least and/or most significant bytes (as required by Control Word bits 5, 4) of the 16-bit counter.
- Repeat with other counters

Only two conventions must be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting will be affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of all three counters. Several commands are available:

- **Control Word Command:** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command:** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command:** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

Table 121 lists the six operating modes for the interval counters.

Table 121. Counter Operating Modes

Mode	Function	Description
0	Out signal on end of count (=0)	Output is 0. When count goes to 0, output goes to 1 and stays at 1 until counter is reprogrammed.
1	Hardware retriggerable one-shot	Output is 0. When count goes to 0, output goes to 1 for one clock time.
2	Rate generator (divide by n counter)	Output is 1. Output goes to 0 for one clock time, then back to 1 and counter is reloaded.
3	Square wave output	Output is 1. Output goes to 0 when counter rolls over, and counter is reloaded. Output goes to 1 when counter rolls over, and counter is reloaded, etc.
4	Software triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.
5	Hardware triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.

21.8.2.2 Reading from the Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters: a simple read operation, counter Latch Command, and the Read-Back Command. Each is explained below.

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

21.8.2.2.1 Simple Read

The first method is to perform a simple read operation. The counter is selected through port 40h (counter 0), 41h (counter 1), or 42h (counter 2).

Note: Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of counter 2, the count can be stopped by writing to the NSC.CNTR2_ENABLE register field.

21.8.2.2.2 Counter Latch Command

The Counter Latch Command, written to port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a two-byte count. The count value is then read from each counter's Count Register as was programmed by the Control Register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch Commands do not affect the programmed mode of the counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

21.8.2.2.3 Read Back Command

The Read Back Command, written to port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back Command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back Commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back Command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

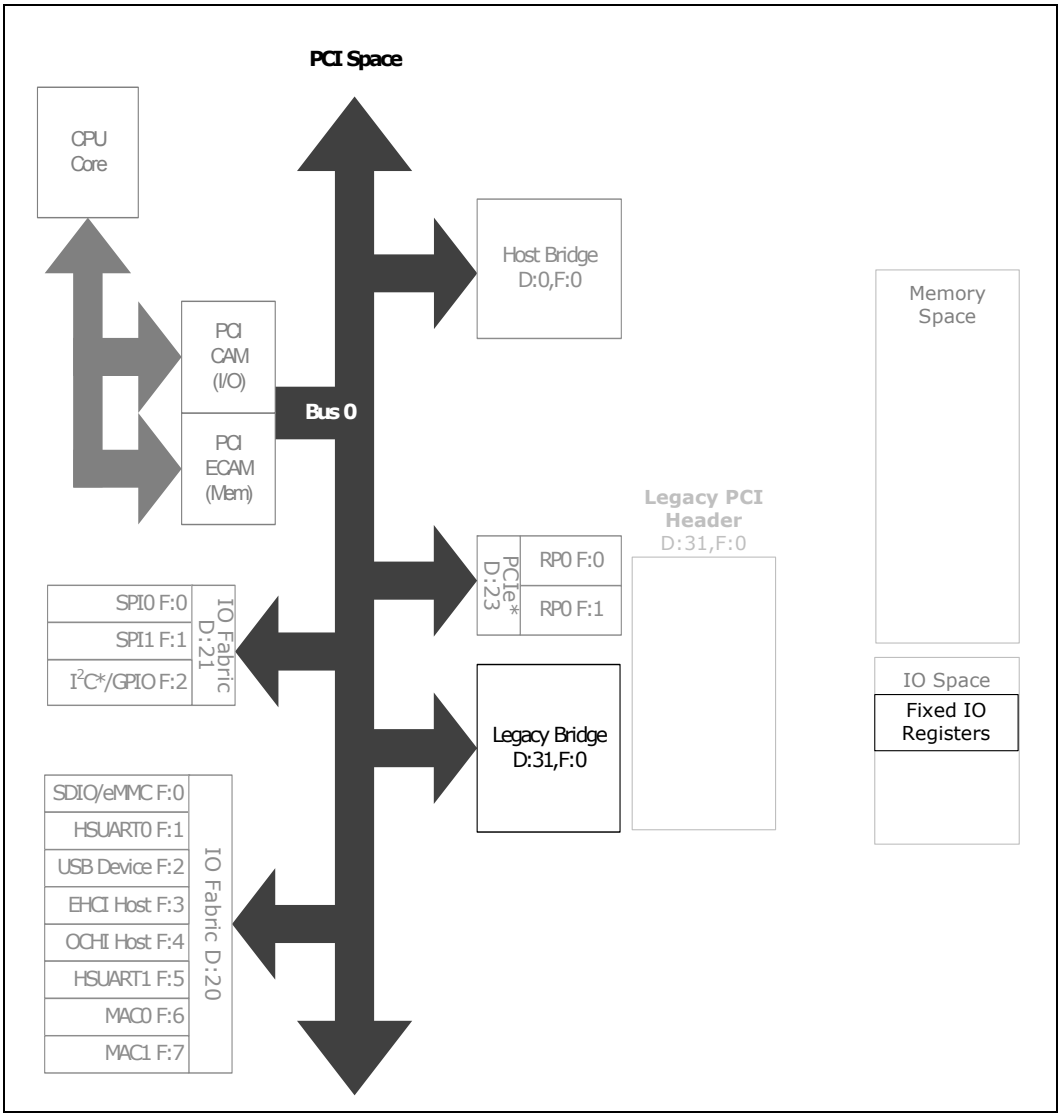
Both count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive separate Read Back Commands. If multiple count and/or status Read Back Commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, return the latched count. Subsequent reads return unlatched count.

21.8.3 Register Map

See Chapter 5.0, “Register Access Methods” for additional information.

Figure 48. 8254 Timers Register Map



21.8.4 Timer I/O Registers

The I/O ports listed in Table 122 have multiple register functions depending on the current programmed state of the 8254. The port numbers referenced in the register descriptions following Table 122 is one possible combination but not the only one.

Table 122. Register Aliases

Port	Alias	Register Name	Default Value	Access
40h	50h	Counter 0 Interval Time Status Byte Format (C0TS)	0xxxxxxxb	RO
		Counter 0 Counter Access Port Register (C0AP)	Undefined	RW
41h	51h	Counter 1 Interval Time Status Byte Format (C1TS)	0xxxxxxxb	RO
		Counter 1 Counter Access Port Register (C1AP)	Undefined	RW

Table 122. Register Aliases

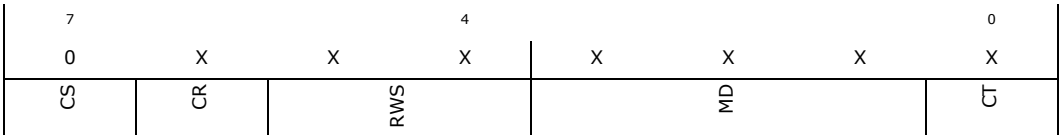
Port	Alias	Register Name	Default Value	Access
42h	52h	Counter 2 Interval Time Status Byte Format (C2TS)	0xxxxxxxb	RO
		Counter 2 Counter Access Port Register (C2AP)	Undefined	RW
43h	-	Timer Control Word Register (TCW)	Undefined	WO
		Read Back Command (RBC)	xxxxxxx0b	WO
		Counter Latch Command (CLC)	xxxx0000b	WO

21.8.4.1 Counter 0 Interval Time Status Byte Format (C0TS)—Offset 40h

Access Method

Type: I/O Register
(Size: 8 bits)

C0TS: 40h



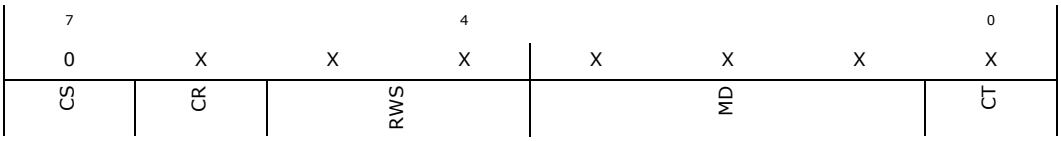
Bit Range	Default & Access	Description
7	0b RO	**Counter State (CS):** When set, OUT of the counter is set. When cleared, OUT of the counter is 0.
6	X RO	**Count Register (CR):** When cleared, indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE) and is available for reading. The time this happens depends on the counter mode.
5: 4	X RO	**Read/Write Selection (RWS):** These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB
3: 1	X RO	**Mode (MD):** Returns the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. Bits Mode Description 000 0 Out signal on end of count (=0) 001 1 Hardware retriggerable one-shot x10 2 Rate generator (divide by n counter) x11 3 Square wave output 100 4 Software triggered strobe 101 5 Hardware triggered strobe
0	X RO	**Countdown Type (CT):** Type: 0 for binary countdown or a 1 for binary coded decimal (BCD) countdown.

21.8.4.2 Counter 1 Interval Time Status Byte Format (C1TS)—Offset 41h

Access Method

Type: I/O Register
(Size: 8 bits)

C1TS: 41h



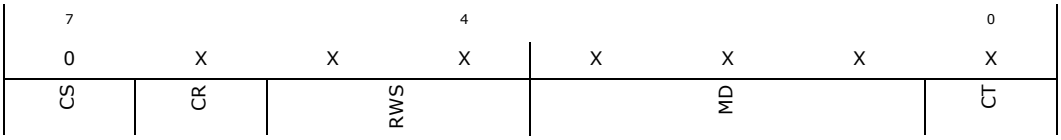
Bit Range	Default & Access	Description
7	0b RO	**Counter State (CS):** When set, OUT of the counter is set. When cleared, OUT of the counter is 0.
6	X RO	**Count Register (CR):** When cleared, indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE) and is available for reading. The time this happens depends on the counter mode.
5: 4	X RO	**Read/Write Selection (RWS):** These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB
3: 1	X RO	**Mode (MD):** Returns the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. Bits Mode Description 000 0 Out signal on end of count (=0) 001 1 Hardware retriggerable one-shot x10 2 Rate generator (divide by n counter) x11 3 Square wave output 100 4 Software triggered strobe 101 5 Hardware triggered strobe
0	X RO	**Countdown Type (CT):** 0 for binary countdown or a 1 for binary coded decimal (BCD) countdown.

21.8.4.3 Counter 2 Interval Time Status Byte Format (C2TS)—Offset 42h

Access Method

Type: I/O Register (Size: 8 bits)

C2TS: 42h



Bit Range	Default & Access	Description
7	0b RO	**Counter State (CS):** When set, OUT of the counter is set. When cleared, OUT of the counter is 0.
6	X RO	**Count Register (CR):** When cleared, indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE) and is available for reading. The time this happens depends on the counter mode.
5: 4	X RO	**Read/Write Select ion (RWS):** These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB

Bit Range	Default & Access	Description
3: 1	X RO	**Mode (MD):** Returns the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. Bits Mode Description 000 0 Out signal on end of count (=0) 001 1 Hardware retriggerable one-shot x10 2 Rate generator (divide by n counter) x11 3 Square wave output 100 4 Software triggered strobe 101 5 Hardware triggered strobe
0	X RO	**Countdown Type (CT):** 0 for binary countdown or a 1 for binary coded decimal (BCD) countdown.

21.8.4.4 Timer Control Word Register (TCW)—Offset 43h

Access Method

Type: I/O Register
(Size: 8 bits)

TCW: 43h

7				4			0
X	X	X	X	X	X	X	X
CS		RWS		CMS			BCS

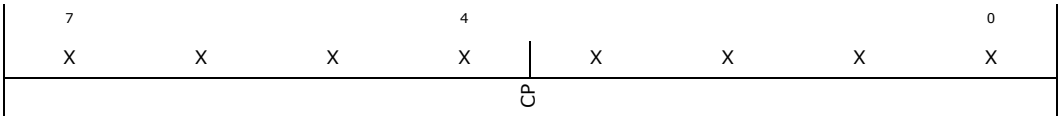
Bit Range	Default & Access	Description
7: 6	X WO	**Counter Select (CS):** The Counter Selection bits select the counter the control word acts upon as shown below. The Read Back Command is selected when bits[7:6] are both 1. 00 Counter 0 select 01 Counter 1 select 10 Counter 2 select 11 Read Back Command
5: 4	X WO	**Read/Write Select (RWS):** The counter programming is done through the counter port (40h for counter 0, 41h for counter 1, and 42h for counter 2) 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB
3: 1	X WO	**Counter Mode Selection (CMS):** Selects one of six modes of operation for the selected counter. 000 = Out signal on end of count (=0) 001 = Hardware retriggerable one-shot x10 = Rate generator (divide by n counter) x11 = Square wave output 100 = Software triggered strobe 101 = Hardware triggered strobe
0	X WO	**Binary/BCD Countdown Select (BCS):** 0 = Binary countdown is used. The largest possible binary count is 216 1 = Binary coded decimal (BCD) count is used. The largest possible BCD count is 104

21.8.4.5 Counter 0 Counter Access Port Register (C0AP)—Offset 50h

Access Method

Type: I/O Register
(Size: 8 bits)

C0AP: 50h



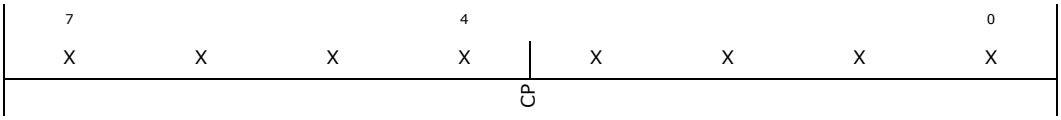
Bit Range	Default & Access	Description
7: 0	X RW	**Counter Port (CP):** Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.

21.8.4.6 Counter 1 Counter Access Port Register (C1AP)—Offset 51h

Access Method

Type: I/O Register
(Size: 8 bits)

C1AP: 51h



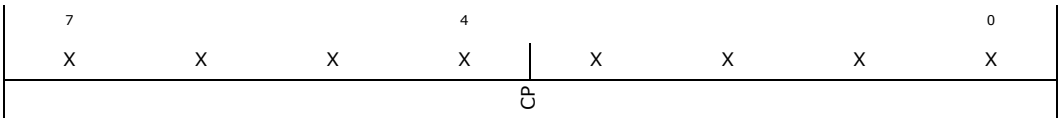
Bit Range	Default & Access	Description
7: 0	X RW	**Counter Port (CP):** Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.

21.8.4.7 Counter 2 Counter Access Port Register (C2AP)—Offset 52h

Access Method

Type: I/O Register
(Size: 8 bits)

C2AP: 52h



Bit Range	Default & Access	Description
7: 0	X RW	**Counter Port (CP):** Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.

21.9 High Precision Event Timer (HPET)

This function provides a set of timers to be used by the operating system for timing events. One timer block is implemented, containing one counter and three timers.

21.9.1 Features

21.9.1.1 Non-Periodic Mode - All Timers

This mode can be thought of as creating a one-shot. When a timer is set up for non-periodic mode, it generates an interrupt when the value in the main counter matches the value in the timer's comparator register. As timers 1 and 2 are 32-bit, they generate another interrupt when the main counter wraps.

T0CV cannot be programmed reliably by a single 64-bit write in a 32-bit environment unless only the periodic rate is being changed. If T0CV must be re-initialized, the following algorithm is performed:

1. Set T0C.TVS
2. Set T0CV[31:0]
3. Set T0C.TVS
4. Set T0CV[63:32]

Every timer is required to support the non-periodic mode of operation.

21.9.1.2 Periodic Mode - Timer 0 Only

In periodic mode, when the main counter value matches the value in T0CV, an interrupt is generated (if enabled). Hardware then increases T0CV by the last value written to T0CV. During run-time, T0CV can be read to find out when the next periodic interrupt will be generated. Software is expected to remember the last value written to T0CV.

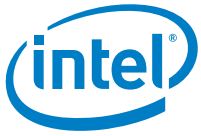
Example: if the value written to T0CV is 00000123h, then:

- An interrupt will be generated when the main counter reaches 00000123h.
- T0CV will then be adjusted to 00000246h.
- Another interrupt will be generated when the main counter reaches 00000246h.
- T0CV will then be adjusted to 00000369h.

When the incremented value is greater than the maximum value possible for T0CV, the value wraps around through 0. For example, if the current value in a 32-bit timer is FFFF0000h and the last value written to this register is 20000, then after the next interrupt the value changes to 00010000h.

If software wants to change the periodic rate, it writes a new value to T0CV. When the timer's comparator matches, the new value is added to derive the next matching point. If software resets the main counter, the value in the comparator's value register must also be reset by setting T0C.TVS. To avoid race conditions, this should be done with the main counter halted. The following usage model is expected:

1. Software clears GCFG.EN to prevent any interrupts.
2. Software clears the main counter by writing a value of 00h to it.
3. Software sets T0C.TVS.
4. Software writes the new value in T0CV.
5. Software sets GCFG.EN to enable interrupts.



21.9.1.3 Interrupts

If each timer has a unique interrupt and the timer has been configured for edge-triggered mode, then there are no specific steps required. If configured to level-triggered mode, then its interrupt must be cleared by software by writing a '1' back to the bit position for the interrupt to be cleared.

Interrupts associated with the various timers have several interrupt mapping options. Software should mask GCFG.LRE when reprogramming HPET interrupt routing to avoid spurious interrupts.

21.9.1.3.1 Mapping Option #1: Legacy Option (GCFG.LRE set)

This forces the following mapping:

Table 123. 8254 Interrupt Mapping

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	The 8254 timer does not cause any interrupts
1	IRQ8	IRQ8	RTC does not cause any interrupts.
2	T2C.IR	T2C.IRC	

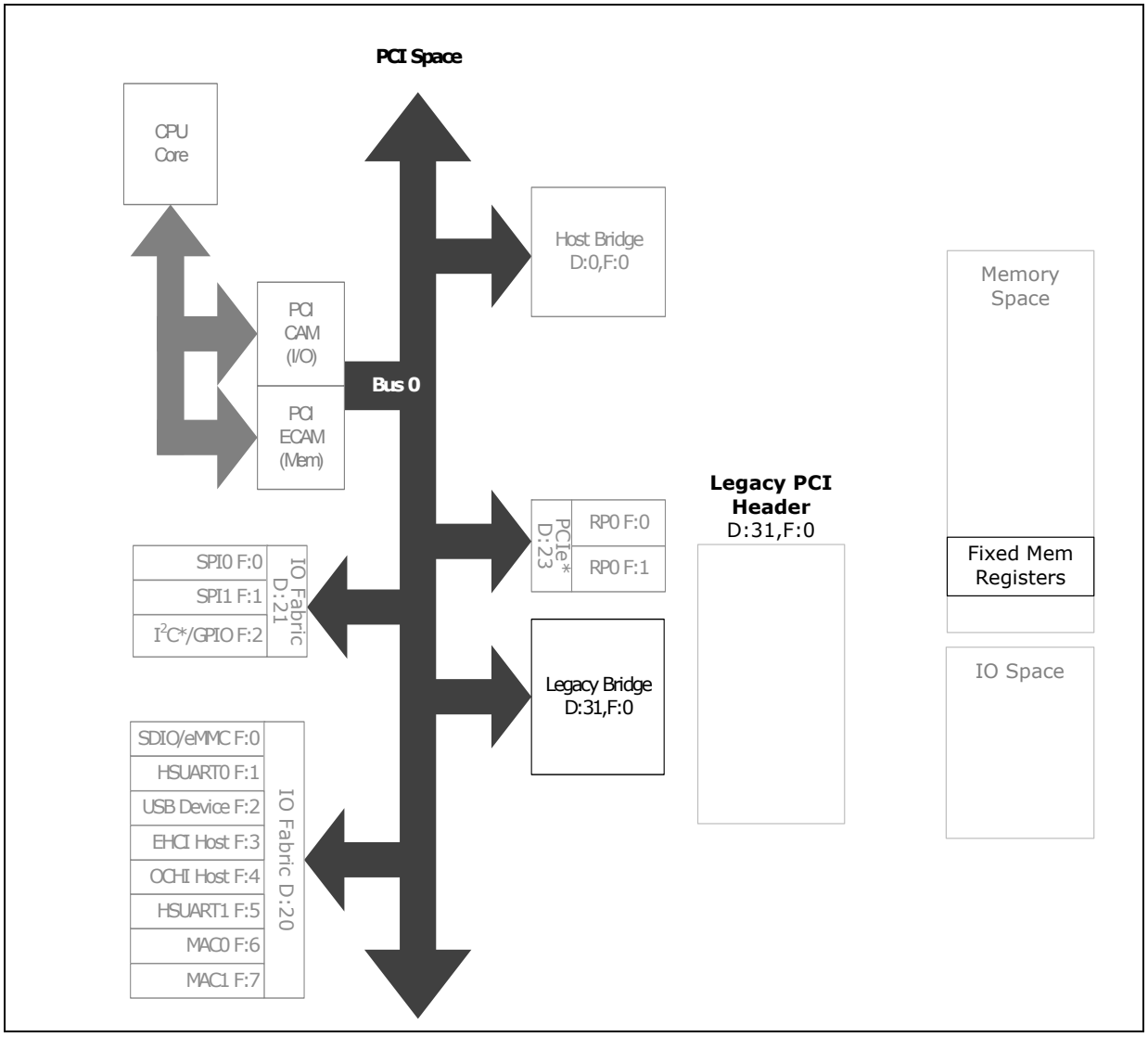
21.9.1.3.2 Mapping Option #2: Standard Option (GCFG.LRE cleared)

Each timer has its own routing control. The interrupts can be routed to various interrupts in the I/O APIC. T[2:0]C.IRC indicates which interrupts are valid options for routing. If a timer is set for edge-triggered mode, the timers should not be shared with any other interrupts.

21.9.2 Register Map

See Chapter 5.0, “Register Access Methods” for additional information.

Figure 49. HPET Register Map



21.9.3 Memory Mapped Registers

The HPET register space is memory mapped to a 1 KB block starting at address FED00000h. All registers are in the core well and reset by RESET#. Accesses that cross register boundaries result in undefined behavior.

Table 124. Summary of Memory Mapped I/O Registers—0xFED00000

Offset Start	Offset End	Register ID—Description	Default Value
0h	3h	"General Capabilities and ID Register - Lower 32 Bits (GCID_1)—Offset 0h" on page 869	8086A201h
4h	7h	"General Capabilities and ID Register - Upper 32 Bits (GCID_2)—Offset 4h" on page 870	0429B17Fh
10h	13h	"General Capabilities and ID Register - Lower 32 Bits (GCID_1)—Offset 0h" on page 869	00000000h

Table 124. Summary of Memory Mapped I/O Registers—0xFED00000 (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
20h	23h	"General Interrupt Status Register (GIS)—Offset 20h" on page 870	00000000h
F0h	F3h	"Main Counter Value Register - Lower 32 Bits (MCV_1)—Offset F0h" on page 871	00000000h
F4h	F7h	"Main Counter Value Register - Upper 32 Bits (MCV_2)—Offset F4h" on page 871	00000000h
100h	103h	"Timer 0 Config and Capabilities Register - Lower 32 Bits (T0C_1)—Offset 100h" on page 872	00000030h
104h	107h	"Timer 0 Config and Capabilities Register - Upper 32 Bits (T0C_2)—Offset 104h" on page 873	00F00000h
108h	10Bh	"Timer 0 Comparator Value Register - Lower 32 Bits (T0CV_1)—Offset 108h" on page 873	FFFFFFFFh
10Ch	10Fh	"Timer 0 Comparator Value Register - Upper 32 Bits (T0CV_2)—Offset 10Ch" on page 873	FFFFFFFFh
120h	123h	"Timer 1 Config and Capabilities Register - Lower 32 Bits (T1C_1)—Offset 120h" on page 874	00000000h
124h	127h	"Timer 1 Config and Capabilities Register - Upper 32 Bits (T1C_2)—Offset 124h" on page 875	00F00000h
128h	12Bh	"Timer 1 Comparator Value Register (T1CV_1)—Offset 128h" on page 875	FFFFFFFFh
140h	143h	"Timer 2 Config and Capabilities Register - Lower 32 Bits (T2C_1)—Offset 140h" on page 875	00000000h
144h	147h	"Timer 2 Config and Capabilities Register - Upper 32 Bits (T2C_2)—Offset 144h" on page 876	00F00800h
148h	14Bh	"Timer 2 Comparator Value Register (T2CV_1)—Offset 148h" on page 877	FFFFFFFFh

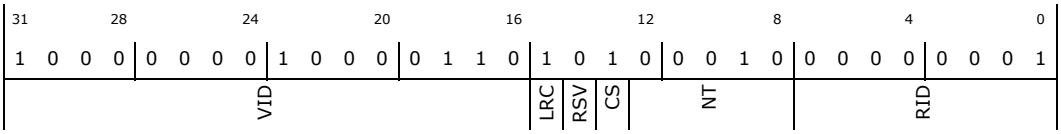
21.9.3.1 General Capabilities and ID Register - Lower 32 Bits (GCID_1)—Offset 0h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

GCID_1: [0xFED00000] + 0h

Default: 8086A201h



Bit Range	Default & Access	Description
31:16	8086h RO	**Vendor ID (VID):** Value of 8086h indicates Intel.
15	1b RO	**Legacy Route Capable (LRC):** Indicates support for Legacy Interrupt Route.
14	0b RO	**Reserved (RSV):** Reserved.
13	1b RO	**Counter Size (CS):** This bit is set to indicate that the main counter is 64 bits wide.
12:8	00010b RO	**Number of Timers (NT):** Indicates that 3 timers are supported.

Bit Range	Default & Access	Description
7:0	01h RO	**Revision ID (RID):** Indicates that revision 1.0 of the specification is implemented.

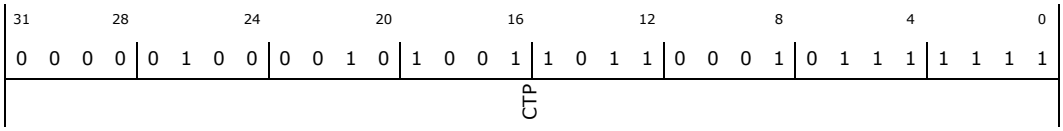
21.9.3.2 General Capabilities and ID Register - Upper 32 Bits (GCID_2)—Offset 4h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

GCID_2: [0xFED00000] + 4h

Default: 0429B17Fh



Bit Range	Default & Access	Description
31:0	0429B17Fh RO	**Counter Tick Period (CTP):** Indicates a period of 69.841279ns, 14.1318 MHz clock.

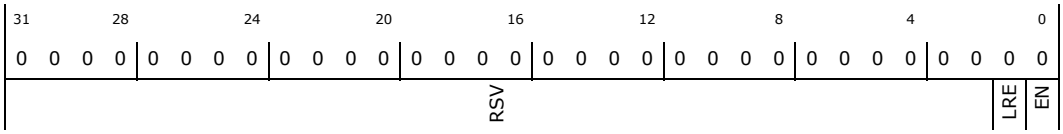
21.9.3.3 General Configuration (GC)—Offset 10h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

GC: [0xFED00000] + 10h

Default: 00000000h



Bit Range	Default & Access	Description
31:2	0b RO	**Reserved (RSV):** Reserved.
1	0b RW	**Legacy Route Enable (LRE):** When set, interrupts will be routed as follows: Timer 0 will be routed to IRQ0 in 8259 and IRQ2 in the I/O APIC Timer 1 will be routed to IRQ8 in 8259 and I/O APIC Timer 2 is routed to IRQ11 in 8259 and Timer 2 will be routed to IOxAPIC as per the routing in T2C.IR When set, the TnC.IR will have no impact for Timers 0 and 1.
0	0b RW	**Overall Enable (EN):** When set, the timers can generate interrupts. When cleared, the main counter will halt and no interrupts will be caused by any timer. For level-triggered interrupts, if an interrupt is pending when this bit is cleared, the GIS.Tx will not be cleared.

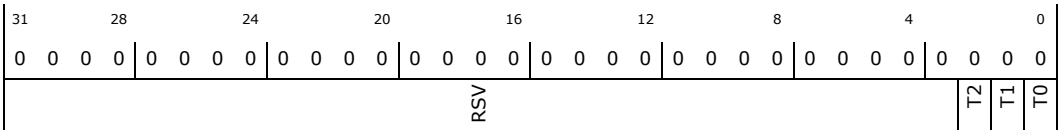
21.9.3.4 General Interrupt Status Register (GIS)—Offset 20h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

GIS: [0xFED00000] + 20h

Default: 00000000h



Bit Range	Default & Access	Description
31:3	0b RO	**Reserved (RSV):** Reserved.
2	0b RW/1C	**Timer 2 Status (T2):** In edge triggered mode, this bit always reads as 0. In level triggered mode, this bit is set when an interrupt is active.
1	0b RW/1C	**Timer 1 Status (T1):** In edge triggered mode, this bit always reads as 0. In level triggered mode, this bit is set when an interrupt is active.
0	0b RW/1C	**Timer 0 Status (T0):** In edge triggered mode, this bit always reads as 0. In level triggered mode, this bit is set when an interrupt is active.

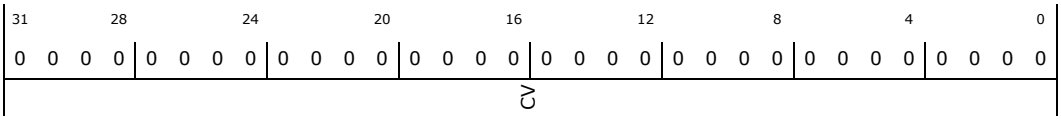
21.9.3.5 Main Counter Value Register - Lower 32 Bits (MCV_1)—Offset F0h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

MCV_1: [0xFED00000] + F0h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Counter Value (CV):** Reads return the current value of the lower 32 bits of the counter. Writes load the new value to the lower 32 bits of the counter.

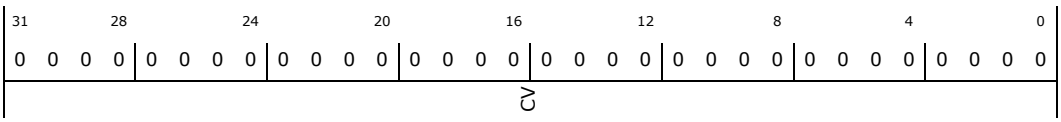
21.9.3.6 Main Counter Value Register - Upper 32 Bits (MCV_2)—Offset F4h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

MCV_2: [0xFED00000] + F4h

Default: 00000000h



Bit Range	Default & Access	Description
31:0	0b RW	**Counter Value (CV):** Reads return the current value of the upper 32 bits of the counter. Writes load the new value to the upper 32 bits of the counter. Timers 1 and Timer 2 return 0.

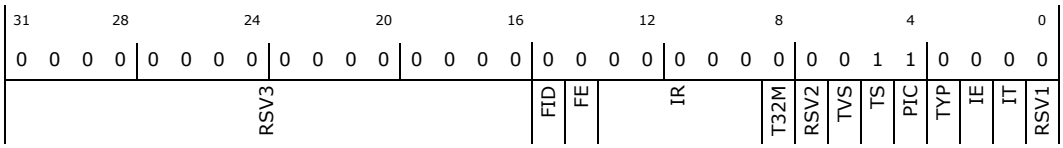
21.9.3.7 Timer 0 Config and Capabilities Register - Lower 32 Bits (T0C_1)—Offset 100h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T0C_1: [0xFED00000] + 100h

Default: 00000030h



Bit Range	Default & Access	Description
31:16	0b RO	**Reserved (RSV3):** Reserved.
15	0b RO	**FSB Interrupt Delivery (FID):** Not Supported
14	0b RO	**FSB Enable (FE):** Not supported, since FID is not supported.
13:9	0b RW	**Interrupt Route (IR):** Indicates the routing for the interrupt to the IOxAPIC. If the value is not supported by this particular timer, the value read back will not match what is written. If GC.LRE is set, then Timers 0 and 1 have a fixed routing, and this field has no effect.
8	0b RW	**Timer 32-bit Mode (T32M):** When set, this bit forces a 64-bit timer to behave as a 32-bit timer.
7	0b RO	**Reserved (RSV2):** Reserved.
6	0b WO	**Timer Value Set (TVS):** This bit will return 0 when read. Writes will only have an effect for Timer 0 if it is set to periodic mode. Writes will have no effect for Timers 1 and 2
5	1b RO	**Timer Size (TS):** 1 = 64-bits, 0 = 32-bits. Set for timer 9. Cleared for timers 1 and 2
4	1b RO	**Periodic Interrupt Capable (PIC):** When set, hardware supports a periodic mode for this timer's interrupt.
3	0b RW	**Timer Type (TYP):** If PIC is set, this bit is read/write, and can be used to enable the timer to generate a periodic interrupt.
2	0b RW	**Interrupt Enable (IE):** When set, enables the timer to cause an interrupt when it times out. When cleared, the timer count and generates status bits, but will not cause an interrupt.
1	0b RW	**Timer Interrupt Type (IT):** When cleared, interrupt is edge triggered. When set, interrupt is level triggered and will be held active until it is cleared by writing 1 to GIS.Tn. If another interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	0b RO	**Reserved (RSV1):** Reserved.

21.9.3.8 Timer 0 Config and Capabilities Register - Upper 32 Bits (T0C_2)—Offset 104h

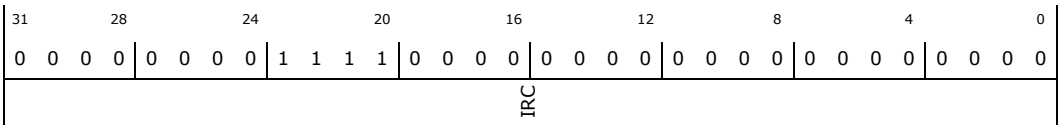
Reads to this register return the current value of the comparator. The default value for each timer is all 1's for the bits that are implemented. Timer 0 is 64-bits wide. Timers 1 and 2 are 32-bits wide.

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T0C_2: [0xFED00000] + 104h

Default: 00F00000h



Bit Range	Default & Access	Description
31:0	00f00000h RO	**Interrupt Route Capability (IRC):** Indicates support for IRQ20, 21, 22, 23

21.9.3.9 Timer 0 Comparator Value Register - Lower 32 Bits (T0CV_1)—Offset 108h

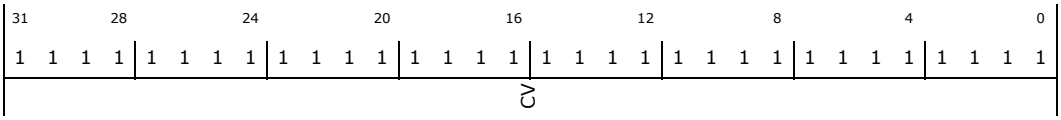
Reads to this register return the current value of the comparator. The default value for each timer is all 1's for the bits that are implemented. Timer 0 is 64-bits wide. Timers 1 and 2 are 32-bits wide.

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T0CV_1: [0xFED00000] + 108h

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31:0	FFFFFFFFh RW	**Comparator Value (CV):** Reads return the current value of the lower 32 bits of the comparator. Writes load the new value to the lower 32 bits of the comparator.

21.9.3.10 Timer 0 Comparator Value Register - Upper 32 Bits (T0CV_2)—Offset 10Ch

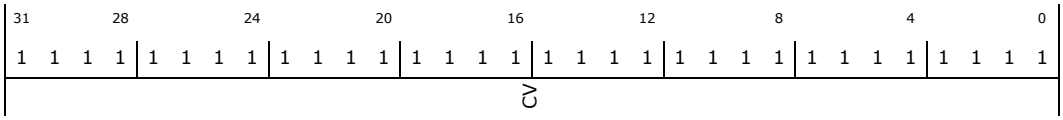
Reads to this register return the current value of the comparator. The default value for each timer is all 1's for the bits that are implemented. Timer 0 is 64-bits wide. Timers 1 and 2 are 32-bits wide.

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T0CV_2: [0xFED00000] + 10Ch

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31:0	FFFFFFFFh RW	**Comparator Value (CV):** Reads return the current value of the upper 32 bits of the comparator. Writes load the new value to the upper 32 bits of the comparator.

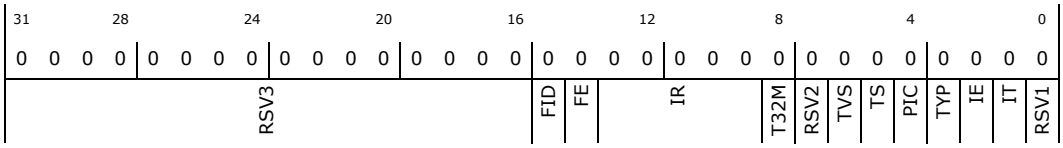
21.9.3.11 Timer 1 Config and Capabilities Register - Lower 32 Bits (T1C_1)—Offset 120h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T1C_1: [0xFED00000] + 120h

Default: 00000000h



Bit Range	Default & Access	Description
31:16	0b RO	**Reserved (RSV3):** Reserved.
15	0b RO	**FSB Interrupt Delivery (FID):** Not Supported.
14	0b RO	**FSB Enable (FE):** Not supported, since FID is not supported.
13:9	0b RW	**Interrupt Route (IR):** Indicates the routing for the interrupt to the IOxAPIC. If the value is not supported by this particular timer, the value read back will not match what is written. If GC.LRE is set, then Timers 0 and 1 have a fixed routing, and this field has no effect.
8	0b RO	**Timer 32-bit Mode (T32M):** Not applicable since Timer 1 is a 32-bit timer.
7	0b RO	**Reserved (RSV2):** Reserved.
6	0b WO	**Timer Value Set (TVS):** This bit will return 0 when read. Writes will only have an effect for Timer 0 if it is set to periodic mode.
5	0b RO	**Timer Size (TS):** 1 = 64-bits, 0 = 32-bits. Set for timer 9. Cleared for timers 1 and 2
4	0b RO	**Periodic Interrupt Capable (PIC):** When set, hardware supports a periodic mode for this timer's interrupt. This bit is set for timer 0, and cleared for timers 1 and 2
3	0b RO	**Timer Type (TYP):** If PIC is set, this bit is read/write, and can be used to enable the timer to generate a periodic interrupt. This bit is RW for timer 0, and RO for timers 1 and 2.
2	0b RW	**Interrupt Enable (IE):** When set, enables the timer to cause an interrupt when it times out. When cleared, the timer count and generates status bits, but will not cause an interrupt.

Bit Range	Default & Access	Description
1	0b RW	**Timer Interrupt Type (IT):** When cleared, interrupt is edge triggered. When set, interrupt is level triggered and will be held active until it is cleared by writing 1 to GIS.Tn. If another interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	0b RO	**Reserved (RSV1):** Reserved.

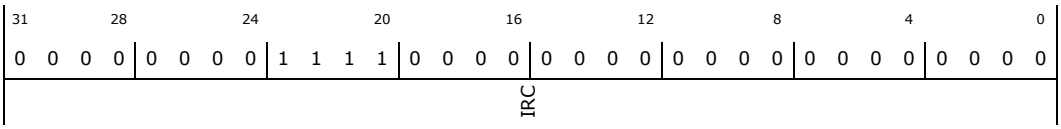
21.9.3.12 Timer 1 Config and Capabilities Register - Upper 32 Bits (T1C_2)—Offset 124h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T1C_2: [0xFED00000] + 124h

Default: 00F00000h



Bit Range	Default & Access	Description
31:0	00f00000h RO	**Interrupt Route Capability (IRC):** Indicates support for IRQ20, 21, 22, 23

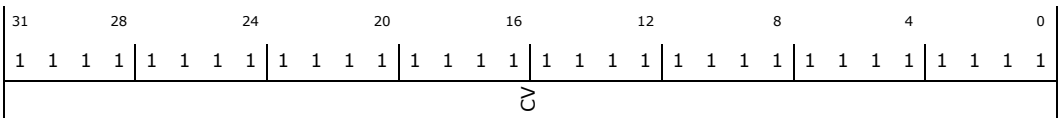
21.9.3.13 Timer 1 Comparator Value Register (T1CV_1)—Offset 128h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T1CV_1: [0xFED00000] + 128h

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31:0	FFFFFFFFh RW	**Comparator Value (CV):** Reads return the current value of the 32 bits of the comparator. Writes load the new value to the 32 bits of the comparator.

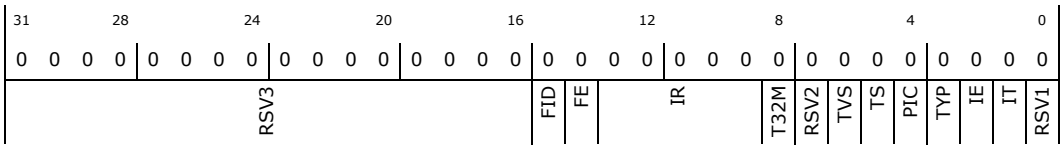
21.9.3.14 Timer 2 Config and Capabilities Register - Lower 32 Bits (T2C_1)—Offset 140h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T2C_1: [0xFED00000] + 140h

Default: 00000000h



Bit Range	Default & Access	Description
31:16	0b RO	**Reserved (RSV3):** Reserved.
15	0b RO	**FSB Interrupt Delivery (FID):** Not Supported.
14	0b RO	**FSB Enable (FE):** Not supported, since FID is not supported.
13:9	0b RW	**Interrupt Route (IR):** Indicates the routing for the interrupt to the IOxAPIC. If the value is not supported by this particular timer, the value read back will not match what is written. If GC.LRE is set, then Timers 0 and 1 have a fixed routing, and this field has no effect.
8	0b RO	**Timer 32-bit Mode (T32M):** Not applicable since Timer 2 is a 32-bit timer.
7	0b RO	**Reserved (RSV2):** Reserved.
6	0b WO	**Timer Value Set (TVS):** This bit will return 0 when read. Writes will only have an effect for Timer 0 if it is set to periodic mode.
5	0b RO	**Timer Size (TS):** 1 = 64-bits, 0 = 32-bits. Set for timer 9. Cleared for timers 1 and 2
4	0b RO	**Periodic Interrupt Capable (PIC):** When set, hardware supports a periodic mode for this timer's interrupt.
3	0b RO	**Timer Type (TYP):** If PIC is set, this bit is read/write, and can be used to enable the timer to generate a periodic interrupt.
2	0b RW	**Interrupt Enable (IE):** When set, enables the timer to cause an interrupt when it times out. When cleared, the timer count and generates status bits, but will not cause an interrupt.
1	0b RW	**Timer Interrupt Type (IT):** When cleared, interrupt is edge triggered. When set, interrupt is level triggered and will be held active until it is cleared by writing 1 to GIS.Tn. If another interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	0b RO	**Reserved (RSV1):** Reserved.

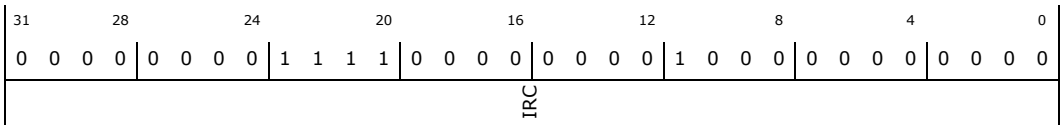
21.9.3.15 Timer 2 Config and Capabilities Register - Upper 32 Bits (T2C_2)—Offset 144h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T2C_2: [0xFED00000] + 144h

Default: 00F00800h



Bit Range	Default & Access	Description
31:0	00f00800h RO	**Interrupt Route Capability (IRC):** Indicates support for IRQ11, 20, 21, 22, 23

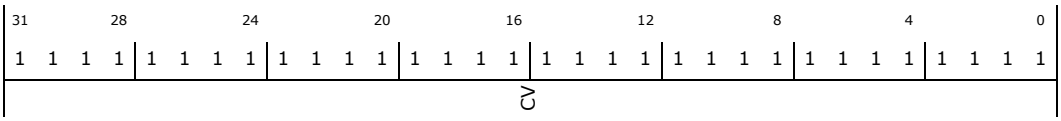
21.9.3.16 Timer 2 Comparator Value Register (T2CV_1)—Offset 148h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

T2CV_1: [0xFED00000] + 148h

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31:0	FFFFFFFFh RW	**Comparator Value (CV):** Reads return the current value of the 32 bits of the comparator. Writes load the new value to the 32 bits of the comparator.

21.9.4 References

IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a.

21.10 Real Time Clock (RTC)

The SoC contains a Motorola MC146818B-compatible real-time clock with 242 bytes of battery-backed RAM. The real-time clock performs two key functions—keeping track of the time of day and storing system data, even when the system is powered down. The RTC operates on an external 32.768 KHz crystal and a 3.3 V battery. An option is also provided to internally generate the 32.768 KHz clock.

The RTC supports two lockable memory ranges. By setting bits in the configuration space, two 8-byte ranges can be locked to read and write accesses. This prevents unauthorized reading of passwords or other system security information.

The RTC supports a date alarm that allows for scheduling a wake up event up to 30 days in advance.

21.10.1 Signal Descriptions

See Chapter 2.0, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 125. RTC Signals

Signal Name	Direction/ Type	Description
RTCX1	I Analog	**Crystal Input 1:** This signal is connected to the 32.768 KHz crystal.
RTCX2	I Analog	**Crystal Input 2:** This signal is connected to the 32.768 KHz crystal.
RTCRST_B	I CMOS3.3	**RTC Reset:** When asserted, this signal resets register bits in the RTC well.
IVCCRTCEXT	I Analog	**External Capacitor Connection**
RTC_EXT_CLK_EN_B	I CMOS3.3	**RTC Internal Clock Select:** Used to select between the oscillator clock from the external 32.768 KHz crystal or an internally generated 32.768 KHz clock. 0 = External 32.768 KHz oscillator 1 = Internal 32.768 KHz Clock (Default)

21.10.2 Features

The Real Time Clock (RTC) module provides a battery backed-up date and time keeping device. Three interrupt features are available: time of day alarm with once a second to once a month range, periodic rates of 122 ms to 500 ms, and end of update cycle notification. Seconds, minutes, hours, days, day of week, month, and year are counted. The hour is represented in twelve or twenty-four hour format, and data can be represented in BCD or binary format. The design is meant to be functionally compatible with the Motorola MS146818B. The time keeping comes from a 32.768 KHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block have very specific functions. The first ten are for time and date information. The next four (0Ah to 0Dh) are registers, which configure and report RTC functions. A host-initiated write takes precedence over a hardware update in the event of a collision.

21.10.2.1 Update Cycles

An update cycle occurs once a second, if the B.SET bit is not asserted and the divide chain is properly configured. During this procedure, the stored time and date are incremented, overflow checked, a matching alarm condition is checked, and the time and date are rewritten to the RAM locations. The update cycle starts at least 488 ms after A.UIP is asserted, and the entire cycle does not take more than 1984 ms to complete. The time and date RAM locations (00h to 09h) are disconnected from the external bus during this time.

21.10.2.2 Interrupts

The real-time clock interrupt is internally routed within the SoC both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt does not leave the SoC, nor is it shared with any other interrupt. However, the High Performance Event Timers can also be mapped to IRQ8#; in this case, the RTC interrupt is blocked.

21.10.2.3 Lockable RAM Ranges

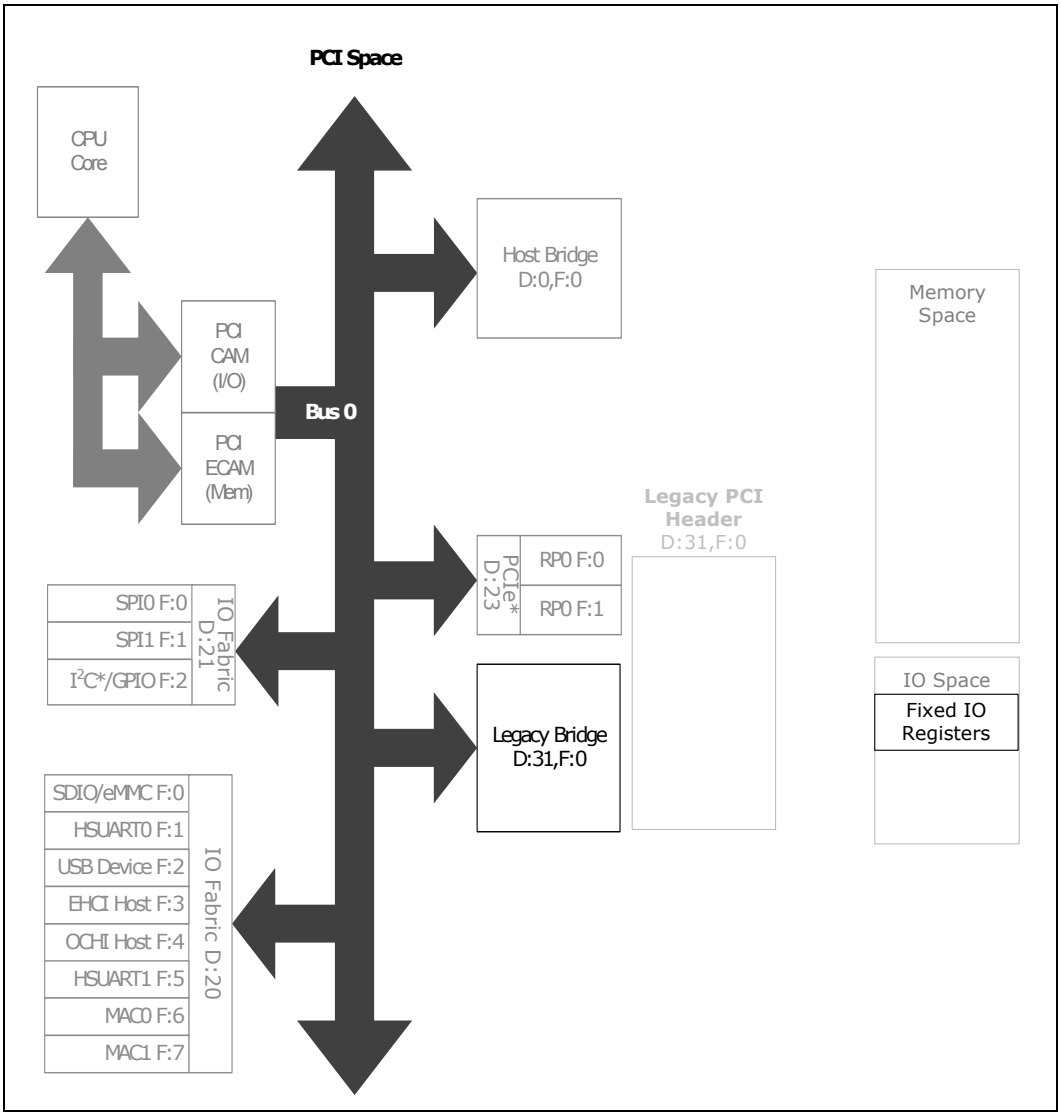
The RTC battery-backed RAM supports two 8-byte ranges that can be locked via the RTC Configuration register. When the locking bits are set, the corresponding range in the RAM is not readable or writable. A write cycle to those locations has no effect. A read cycle to those locations does not return the location’s actual value (resultant value is undefined).

Once a range is locked, the range can be unlocked only by a hard reset, which invokes the BIOS and allows it to relock the RAM range.

21.10.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 50. RTC Register Map



21.10.4 I/O Registers

The RTC internal registers and RAM are organized as two banks of 128 bytes each, called the standard and extended banks.

The first 14 bytes of the standard bank contain the RTC time and date information along with four registers, A - D, that are used for configuration of the RTC. The extended bank contains a full 128 bytes of battery backed SRAM. All data movement between the host CPU and the RTC is done through registers mapped to the standard I/O space.

Note: It is not possible to disable the extended bank.

Note: I/O Locations 70h and 71h are used for data movement to and from the standard bank. Locations 72h and 73h used for data movement to and from the extended bank. All of these I/O locations also have alias I/O locations, as indicated in Table 126. Index addresses above 127 are not valid.

Note: Writes to 74h do not affect the NMI Enable bit of 70h

Table 126. I/O Registers Alias Locations

Register	Original I/O Location	Alias I/O Location
Real-Time Clock (Standard RAM) Index Register	70h	74h
Real-Time Clock (Standard RAM) Target Register	71h	75h
Extended RAM Index Register	72h	76h
Extended RAM Target Register	73h	77h

21.10.5 Indexed Registers

The RTC contains two sets of indexed registers, which are accessed using the two separate Index and Target registers (70/71h or 72/73h).

Table 127. Indexed Registers

Start	End	Name
00h	00h	Seconds
01h	01h	Seconds Alarm
02h	02h	Minutes
03h	03h	Minutes Alarm
04h	04h	Hours
05h	05h	Hours Alarm
06h	06h	Day of Week
07h	07h	Day of Month
08h	08h	Month
09h	09h	Year
0Ah	0Ah	Register A
0Bh	0Bh	Register B
0Ch	0Ch	Register C
0Dh	0Dh	Register D
0Eh	7Fh	114 Bytes of User RAM

21.10.5.1 Offset 0Ah: Register A

This register is in the RTC well, and is used for general configuration of the RTC functions.

Access Method

Type: RTC Indexed Register
(Size: 8 bits)

Default: xxxxxxxxb

7	4	0
x	xxx	xxxx
UIP	DV	RS

Bit Range	Default & Access	Description
7	xb RW	**Update in progress (UIP):** When set, an update is in progress. When cleared, the update cycle will not start for at least 488 µs. The time, calendar, and alarm information in RAM is always available when this bit is cleared.
6: 4	xb RW	**Division Chain Select:** Controls the divider chain for the oscillator; not affected by RSMRST# or any other reset signal. 000b: Invalid 001b: Invalid 010b: Normal Operation 011b: Bypass 5 Stages (Test Mode Only) 100b: Bypass 10 Stages (Test Mode Only) 101b: Bypass 15 Stages (Test Mode Only) 110b: Divider Reset 111b: Divider Reset
3: 0	xb RW	**Rate Select:** Selects one of 13 taps of the 15 stage divider chain. The selected tap can generate a periodic interrupt when B.PIE bit is set. Otherwise this tap sets C.PF. 0000b: Interrupt Never Toggles 0001b: 3.90625 ms 0010b: 7.8125 ms 0011b: 122.070 µs 0100b: 244.141 µs 0101b: 488.281 µs 0110b: 976.5625 µs 0111b: 1.953125 ms 1000b: 3.90625 ms 1001b: 7.8125 ms 1010b: 15.625 ms 1011b: 31.25 ms 1100b: 62.5 ms 1101b: 125 ms 1110b: 250 ms 1111b: 500 ms

21.10.5.2 Offset 0Bh: Register B - General Configuration

This register resides in the resume well.

Access Method

Type: RTC Indexed Register
(Size: 8 bits)

Default: x0x00xxxb

7			4				0
x	0	x	0	0	x	x	x
SET	PIE	AIE	UIE	SQWE	DM	HF	DSE

Bit Range	Default & Access	Description
7	xb RW	**Set Clock (SET):** When cleared, an update cycle occurs once each second. If set, a current update cycle will abort and subsequent update cycles will not occur until SET is returned to zero. When set, SW may initialize time and calendar bytes safely.
6	0b RW	**Periodic Interrupt Enable (PIE):** When set, and C.PF is set, an interrupt is generated.
5	xb RW	**Alarm Interrupt Enable (AIE):** When set, and C.AF is set, an interrupt is generated.
4	0b RW	**Update-ended Interrupt Enable (UIE):** When set and C.UF is set, an interrupt is generated.
3	0b RW	**Square Wave Enable (SQWE):** Not implemented.
2	xb RW	**Data Mode (DM):** When set, represents binary representation. When cleared, denotes BCD.
1	xb RW	**Hour Format (HF):** When set, twenty-four hour mode is selected. When cleared, twelve-hour mode is selected. In twelve hour mode, the seventh bit represents AM (cleared) and PM (set).
0	xb RW	**Daylight Savings Enable (DSE):** Not implemented

21.10.5.3 Offset 0Ch: Register C - Flag Register

All bits in this register are cleared when this register is read.

Access Method

Type: RTC Indexed Register
(Size: 8 bits)

Default: 00x00000b

7			4	0
0	0	x	0	0000
IRQF	PF	AF	UF	RSV1

Bit Range	Default & Access	Description
7	0b RC	**Interrupt Request Flag (IRQF)**: This bit is an AND of the flag with its corresponding interrupt enable in register B, and causes the RTC Interrupt to be asserted.
6	0b RC	**Periodic Interrupt Flag (PF):** Set when the tap as specified by A.RS is one.
5	xb RC	**Alarm Flag (AF):** Set after all Alarm values match the current time.
4	0b RC	**Update-ended Flag (UF):** Set immediately following an update cycle for each second.

Bit Range	Default & Access	Description
3: 0	0b RO	**Reserved (RSV1)**

21.10.5.4 Offset 0Dh: Register D - Flag Register

Access Method

Type: RTC Indexed Register
(Size: 8 bits)

Default: 1xxxxxxxb

7		4	0
1	x	xxxxxx	
VRT	RSV1	DA	

Bit Range	Default & Access	Description
7	0b RC	**Valid RAM and Time Bit (VRT):** This bit should always be written as a 0 for write cycle, however it will return a 1 for read cycles.
6	0b RC	**Reserved (RSV1):** This bit always returns a 0 and should be set to 0 for write cycles.
5: 0	xb RC	**Date Alarm (DA):** These bits store the date of month alarm value. If set to 000000, then a don't care state is assumed. If the date alarm is not enabled, these bits will return zeros to mimic the functionality of the Motorola 146818B. These bits are not affected by any reset assertion.

21.10.6 References

Accessing the Real Time Clock Registers and the NMI Enable Bit:
ftp://download.intel.com/design/intarch/PAPERS/321088.pdf

21.11 Interrupt Decoding & Routing

The Legacy Bridge provides registers that are used to decode and route interrupts received from devices within the SoC.

The interrupt decoder is responsible for receiving interrupt messages from other devices in the SoC and decoding them for consumption by the interrupt router, the 8259 PICs and/or the I/O APIC.

The interrupt router is responsible for mapping each incoming interrupt to the appropriate PIRQx, for consumption by the 8259 PICs and/or I/O APIC.

21.11.1 Features

21.11.1.1 Interrupt Decoder

The interrupt decoder receives interrupt messages from devices in the SoC. These interrupts can be split into two primary groups:

- For consumption by the interrupt router
- For consumption by the 8259 PIC

21.11.1.1.1 For Consumption by the Interrupt Router

When a PCI-mapped device in the SoC asserts or de-asserts an INT[A:D] interrupt, an interrupt message is sent to the decoder. This message is decoded to indicate to the interrupt router which specific interrupt is asserted or de-asserted and which device the INT[A:D] interrupt originated from.

21.11.1.1.2 For Consumption by the 8259 PIC

When a device in the SoC asserts or deasserts a legacy interrupt (IRQ), an interrupt message is sent to the decoder. This message is decoded to indicate to the 8259 PIC which specific interrupt (IRQ[3, 4, 5, 6, 7, 13, 14 or 15]) was asserted or deasserted.

21.11.1.2 Interrupt Router

The interrupt router aggregates the INT[A:D] interrupts for each PCI-mapped device in the SoC, received from the interrupt decoder. It then maps these aggregated interrupts to 8 PCI-based interrupts: PIRQ[A:H]. This mapping is configured using the 4 Interrupt Queue Agent Registers: IRQAGENT0, IRQAGENT1, IRQAGENT2 and IRQAGENT3.

Table 128. IRQAGENT Description

IRQAGENT	Description	Single-Function/ Multi-Function
0	Remote Management Unit	Single-Function (Supports INTA only)
1	PCIe* D:23	Multi-Function (Supports INTA, INTB, INTC & INTD)
2	Reserved	Single-Function (Supports INTA only)
3	IO Fabric D:20 & D21	Multi-Function (Supports INTA, INTB, INTC & INTD)

PCI based interrupts PIRQ[A:H] are then available for consumption by either the 8259 PICs or the IO-APIC, depending on the configuration of the 8 PIRQx Routing Control Registers: PIRQA, PIQRB, PIRQC, PIRQD, PIRQE, PIRQF, PIRQG, PIRQH.

When the PCI based interrupts are consumed by the IO-APIC, a fixed routing scheme is used where interrupts PIRQ[A:H] are routed to IO-APIC interrupts IRQ[16:23].

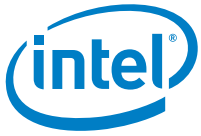
21.11.1.2.1 Routing PCI Based Interrupts to 8259 PIC

The interrupt router can be programmed to allow PIRQA-PIRQH to be routed internally to the 8259 as ISA compatible interrupts IRQ 3–7, 9–12 & 14–15. The assignment is programmable through the 8 PIRQx Routing Control Registers: PIRQA, PIQRB, PIRQC, PIRQD, PIRQE, PIRQF, PIRQG, PIRQH. See Section 21.3 for register details. One or more PIRQs can be routed to the same IRQ input. If ISA Compatible Interrupts are not required, the Route registers can be programmed to disable steering.

The PIRQx# lines are defined as active low, level sensitive. When a PIRQx# is routed to specified IRQ line, software must change the IRQ's corresponding ELCR bit to level sensitive mode. The SoC internally inverts the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an active high device (through SERIRQ). However, active low interrupts can share their interrupt with PCI interrupts.

21.12 8259 Programmable Interrupt Controllers (PIC)

The SoC provides an ISA-compatible programmable interrupt controller (PIC) that incorporates the functionality of two, cascaded 8259 interrupt controllers.



21.12.1 Features

In addition to providing support for ISA compatible interrupts, this interrupt controller can also support PCI based interrupts (PIRQs) by mapping the PCI interrupt onto a compatible ISA interrupt line. Each 8259 controller supports eight interrupts, numbered 0–7. Table 129 shows how the controllers are connected.

Note: The SoC does not implement any external PIRQ# signals. The PIRQs referred to in this section originate from the interrupt routing unit.

Table 129. Interrupt Controller Connections

8259	8259 Input	Connected Pin / Function
Master	0	Internal Timer / Counter 0 output or HPET Timer #0
	1	Reserved
	2	Slave controller INTR output
	3	IRQ3 via PIRQx
	4	IRQ4 via PIRQx
	5	IRQ5 via PIRQx
	6	IRQ6 via PIRQx
	7	IRQ7 via PIRQx
Slave	0	Inverted IRQ8# from internal RTC or HPET Timer #1
	1	IRQ9 via SCI or PIRQx
	2	IRQ10 via SCI or PIRQx
	3	IRQ11 via SCI or PIRQx or HPET Timer #2
	4	IRQ12 via PIRQx
	5	Reserved
	6	IRQ14 via PIRQx
	7	IRQ15 via PIRQx

The SoC cascades the slave controller onto the master controller through master controller interrupt input 2. This means there are only 15 possible interrupts for the SoC PIC.

Interrupts can be programmed individually to be edge or level, except for IRQ0, IRQ2 and IRQ8#.

Note: Active-low interrupt sources (such as a PIRQ#) are inverted inside the SoC. In the following descriptions of the 8259s, the interrupt levels are in reference to the signals at the internal interface of the 8259s, after the required inversions have occurred. Therefore, the term “high” indicates “active,” which means “low” on an originating PIRQ#.

21.12.1.1 Interrupt Handling

21.12.1.1.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 130 defines the IRR, ISR, and IMR.

Table 130. Interrupt Status Registers

Bit	Description
IRR	**Interrupt Request Register.** This bit is set on a low to high transition of the interrupt line in edge mode, and by an active high level in level mode.
ISR	**Interrupt Service Register.** This bit is set, and the corresponding IRR bit cleared, when an interrupt acknowledge cycle is seen, and the vector returned is for that interrupt.
IMR	**Interrupt Mask Register.** This bit determines whether an interrupt is masked. Masked interrupts will not generate INTR.

21.12.1.1.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle that is translated into a Interrupt Acknowledge Cycle by the SoC. The PIC translates this command into two internal INTA# pulses expected by the 8259 controllers. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave sends the interrupt vector to the processor with the acknowledged interrupt code. This code is based upon the ICW2.IVBA bits, combined with the ICW2.IRL bits representing the interrupt within that controller.

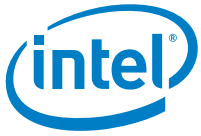
Note: References to ICWx and OCWx registers are relevant to both the master and slave 8259 controllers.

Table 131. Content of Interrupt Vector Byte

Master, Slave Interrupt	Bits [7:3]	Bits [2:0]
IRQ7,15	ICW2.IVBA	111
IRQ6,14		110
IRQ5,13		101
IRQ4,12		100
IRQ3,11		011
IRQ2,10		010
IRQ1,9		001
IRQ0,8		000

21.12.1.1.3 Hardware/Software Interrupt Sequence

1. One or more of the Interrupt Request lines (IRQ) are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor if an asserted interrupt is not masked.
3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle.
4. Upon observing the special cycle, the SoC converts it into the two cycles that the internal 8259 pair can respond to. Each cycle appears as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.
5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine if it must respond with an interrupt vector during the second INTA# pulse.



6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. If no interrupt request is present because the request was too short in duration, the PIC returns vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

21.12.1.2 Initialization Command Words (ICWx)

Before operation can begin, each 8259 must be initialized. This is a four byte sequence to ICW1, ICW2, ICW3 and ICW4. The four initialization command words are referred to by their acronyms: ICW1, ICW2, ICW3, and ICW4.

The base address for each 8259 initialization command word is a fixed location in the I/O memory space: 20h for the master controller, and A0h for the slave controller.

21.12.1.2.1 ICW1

A write to the master or slave controller base address with data bit 4 equal to 1 is interpreted as a write to ICW1. Upon sensing this write, the PIC expects three more byte writes to 21h for the master controller, or A1h for the slave controller, to complete the ICW sequence.

A write to ICW1 starts the initialization sequence during which the following automatically occur:

1. Following initialization, an interrupt request (IRQ) input must make a low-to-high transition to generate an interrupt.
2. The Interrupt Mask Register is cleared.
3. IRQ7 input is assigned priority 7.
4. The slave mode address is set to 7.
5. Special mask mode is cleared and Status Read is set to IRR.

21.12.1.2.2 ICW2

The second write in the sequence (ICW2) is programmed to provide bits [7:3] of the interrupt vector that will be released during an interrupt acknowledge. A different base is selected for each interrupt controller.

21.12.1.2.3 ICW3

The third write in the sequence (ICW3) has a different meaning for each controller.

- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. Within the SoC, IRQ2 is used. Therefore, MICW3.CCC is set to a 1, and the other bits are set to 0s.
- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller if the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and if it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

21.12.1.2.4 ICW4

The final write in the sequence (ICW4) must be programmed for both controllers. At the very least, ICW4.MM must be set to a 1 to indicate that the controllers are operating in an Intel Architecture-based system.

21.12.1.3 Operation Command Words (OCW)

These command words reprogram the Interrupt controller to operate in various interrupt modes.

- OCW1 masks and unmasks interrupt lines.
- OCW2 controls the rotation of interrupt priorities when in rotating priority mode, and controls the EOI function.
- OCW3 sets up ISR/IRR reads, enables/disables the special mask mode (SMM), and enables/disables polled interrupt mode.

21.12.1.4 Modes of Operation

21.12.1.4.1 Fully Nested Mode

In this mode, interrupt requests are ordered in priority from 0 through 7, with 0 being the highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the ISR for the interrupt is set. This ISR bit remains set until either: the processor issues an EOI command immediately before returning from the service routine; or if in AEOI mode, on the trailing edge of the second INTA#. While the ISR bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels generate another interrupt.

Interrupt priorities can be changed in the rotating priority mode.

21.12.1.4.2 Special Fully-Nested Mode

This mode is used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully-nested mode is programmed to the master controller. This mode is similar to the fully-nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave are recognized by the master and initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.
- When exiting the Interrupt Service routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a non-Specific EOI command to the slave and then reading its ISR. If it is 0, a non-specific EOI can also be sent to the master.

21.12.1.4.3 Automatic Rotation Mode (Equal Priority Devices)

In some applications, there are a number of interrupting devices of equal priority. Automatic rotation mode provides for a sequential 8-way rotation. In this mode, a device receives the lowest priority after being serviced. In the worst case, a device requesting an interrupt has to wait until each of seven other devices are serviced at most once.

There are two ways to accomplish automatic rotation using OCW2.REOI; the Rotation on Non-Specific EOI Command (OCW2.REOI=101b) and the rotate in automatic EOI mode which is set by (OCW2.REOI=100b).

21.12.1.4.4 Specific Rotation Mode (Specific Priority)

Software can change interrupt priorities by programming the bottom priority. For example, if IRQ5 is programmed as the bottom priority device, then IRQ6 is the highest priority device. The Set Priority Command is issued in OCW2 to accomplish this, where: OCW2.REOI=11xb, and OCW2.ILS is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes can be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (OCW2.REOI=111b) and OCW2.ILS=IRQ level to receive bottom priority.

21.12.1.4.5 Poll Mode

Poll mode can be used to conserve space in the interrupt vector table. Multiple interrupts that can be serviced by one interrupt service routine do not need separate vectors if the service routine uses the poll command. Poll mode can also be used to expand the number of interrupts. The polling interrupt service routine can call the appropriate service routine, instead of providing the interrupt vectors in the vector table. In this mode, the INTR output is not used and the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting OCW3.PMC. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit if there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read contains a 1 in Bit 7 if there is an interrupt, and the binary code of the highest priority level in Bits 2:0.

21.12.1.4.6 Edge and Level Triggered Mode

In ISA systems this mode is programmed using ICW1.LTIM, which sets level or edge for the entire controller. In the SoC, this bit is disabled and a register for edge and level triggered mode selection, per interrupt input, is included. This is the Edge/Level control Registers ELCR1 and ELCR2.

If an ELCR bit is 0, an interrupt request will be recognized by a low-to-high transition on the corresponding IRQ input. The IRQ input can remain high without generating another interrupt. If an ELCR bit is 1, an interrupt request will be recognized by a high level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

In both the edge and level triggered modes, the IRQ inputs must remain active until after the falling edge of the first internal INTA#. If the IRQ input goes inactive before this time, a default IRQ7 vector is returned.

21.12.1.4.7 End of Interrupt (EOI) Operations

An EOI can occur in one of two fashions: by a command word write issued to the PIC before returning from a service routine, the EOI command; or automatically when the ICW4.AEOI bit is set to 1.

21.12.1.4.8 Normal End of Interrupt

In normal EOI, software writes an EOI command before leaving the interrupt service routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific. When a Non-Specific EOI command is issued, the PIC clears the highest ISR bit of those that are set to 1. Non-Specific EOI is the normal mode of operation of the PIC within the SoC, as the interrupt being serviced currently is the

interrupt entered with the interrupt acknowledge. When the PIC is operated in modes that preserve the fully nested structure, software can determine which ISR bit to clear by issuing a Specific EOI.

An ISR bit that is masked is not cleared by a Non-Specific EOI if the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

21.12.1.4.9 Automatic End of Interrupt Mode

In this mode, the PIC automatically performs a Non-Specific EOI operation at the trailing edge of the last interrupt acknowledge pulse. From a system standpoint, this mode should be used only when a nested multi-level interrupt structure is not required within a single PIC. The AEOI mode can only be used in the master controller and not the slave controller.

Note: Both the master and slave PICs have an AEOI bit: MICW4.AEOI and SICW4.AEOI respectively. Only the MICW4.AEOI bit should be set by software. The SICW4.AEOI bit should not be set by software.

21.12.1.5 Masking Interrupts

21.12.1.5.1 Masking on an Individual Interrupt Request

Each interrupt request can be masked individually by the Interrupt Mask Register (IMR). This register is programmed through OCW1. Each bit in the IMR masks one interrupt channel. Masking IRQ2 on the master controller masks all requests for service from the slave controller.

21.12.1.5.2 Special Mask Mode

Some applications may require an interrupt service routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The special mask mode enables all interrupts not masked by a bit set in the Mask register. Normally, when an interrupt service routine acknowledges an interrupt without issuing an EOI to clear the ISR bit, the interrupt controller inhibits all lower priority requests. In the special mask mode, any interrupts may be selectively enabled by loading the Mask Register with the appropriate pattern.

The special mask mode is set by OCW3.ESMM=1b & OCW3.SMM=1b, and cleared where OCW3.ESMM=1b & OCW3.SMM=0b.

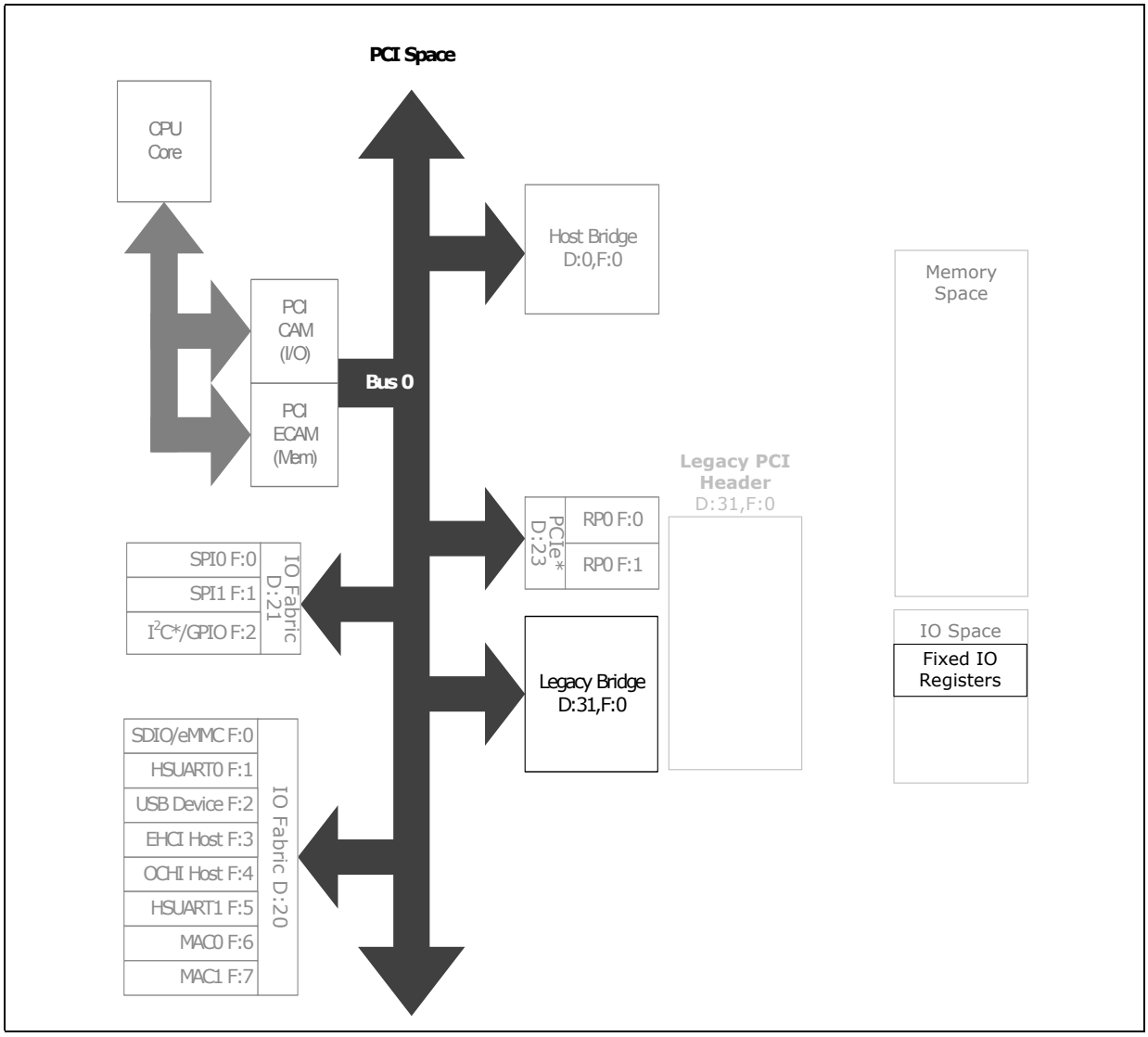
21.12.1.6 Steering of PCI Interrupts

The SoC can be programmed to allow PIRQ[A:H]# to be internally routed to interrupts 3-7, 9-12, 14 or 15, through the PIRQx Route Control registers in Device 31:Function 0. One or more PIRQx# lines can be routed to the same IRQx input. The PIRQx# lines are defined as active low, level sensitive. When a PIRQx# is routed to specified IRQ line, software must change the corresponding ELCR1 or ELCR2 register to level sensitive mode. The SoC will internally invert the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an ISA device

21.12.2 Register Map

See Chapter 5.0, “Register Access Methods” for additional information.

Figure 51. 8259 Register Map



21.12.3 I/O Registers

The interrupt controller registers are located at 20h and 21h for the master controller (IRQ0-7), and at A0h and A1h for the slave controller (IRQ8-13). These registers have multiple functions, depending upon the data written to them. Table 132 describes the different register possibilities for each address.

Note: The register descriptions after Table 132 represent one register possibility.

Table 132. 8259 I/O Registers Alias Locations

Registers	Original I/O Location	Alias I/O Locations
MICW1 MOCW2 MOCW3	20h	24h
		28h
		2Ch
		30h
		34h
		38h
		3Ch
MICW2 MICW3 MICW4 MOCW1	21h	25h
		29h
		2Dh
		31h
		35h
		39h
		3Dh
SICW1 SoCW2 SoCW3	A0h	A4h
		A8h
		ACh
		B0h
		B4h
		B8h
		BCh
SICW2 SICW3 SICW4 SoCW1	A1h	A5h
		A9h
		ADh
		B1h
		B5h
		B9h
		BDh
ELCR1	4D0h	N/A
ELCR2	4D1h	N/A

Table 133. Summary of I/O Registers

Offset Start	Offset End	Register ID—Description	Default Value
20h	20h	"Master Initialization Command Word 1 (MICW1)—Offset 20h" on page 893	81Fh
21h	21h	"Master Initialization Command Word 2 (MICW2)—Offset 21h" on page 894	63h
24h	24h	"Master Operational Control Word 2 (MOCW2)—Offset 24h" on page 894	67h

Table 133. Summary of I/O Registers (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
25h	25h	"Master Initialization Command Word 3 (MICW3)—Offset 25h" on page 895	E7h
28h	28h	"Master Operational Control Word 3 (MOCW3)—Offset 28h" on page 895	19Eh
29h	29h	"Master Initialization Command Word 4 (MICW4)—Offset 29h" on page 896	421h
2Dh	2Dh	"Master Operational Control Word 1 (MOCW1)—Offset 2Dh" on page 897	00h
A0h	A0h	"Slave Initialization Command Word 1 (SICW1)—Offset A0h" on page 897	81Fh
A1h	A1h	"Slave Initialization Command Word 2 (SICW2)—Offset A1h" on page 898	63h
A4h	A4h	"Slave Operational Control Word 2 (SoCW2)—Offset A4h" on page 898	67h
A5h	A5h	"Slave Initialization Command Word 3 (SICW3)—Offset A5h" on page 899	E7h
A8h	A8h	"Slave Operational Control Word 3 (SoCW3)—Offset A8h" on page 899	19Eh
A9h	A9h	"Slave Initialization Command Word 4 (SICW4)—Offset A9h" on page 900	421h
ADh	ADh	"Slave Operational Control Word 1 (SoCW1)—Offset ADh" on page 901	00h
4D0h	4D0h	"Master Edge/Level Control (ELCR1)—Offset 4D0h" on page 901	108h
4D1h	4D1h	"Slave Edge/Level Control (ELCR2)—Offset 4D1h" on page 901	14Ah

21.12.3.1 Master Initialization Command Word 1 (MICW1)—Offset 20h

A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

- The Interrupt Mask register is cleared.
- IRQ7 input is assigned priority 7.
- The slave mode address is set to 7.
- Special Mask Mode is cleared and Status Read is set to IRR.

Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Access Method

Type: I/O Register (Size: 8 bits)

MICW1: 20h

7			4				0
X	X	X	X	X	X	X	X
	MCS85		ICWOCWSEL	LTIM	ADI	SNGL	IC4

Bit Range	Default & Access	Description
7: 5	X WO	**MCS85 (MCS85):** These bits are MCS-85 specific, and not needed. Should be programmed to 000
4	X WO	**ICW/OCW Select (ICWOCWSEL):** This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.
3	X WO	**Edge/Level Bank Select (LTIM):** Disabled. Replaced by ELCR1 and ELCR2.
2	X WO	**ADI (ADI):** Should be programmed to 0.
1	X WO	**Single or Cascade (SNGL):** Must be programmed to a 0 to indicate two controllers operating in cascade mode.
0	X WO	**wICW4 Write Required (IC4):** This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.

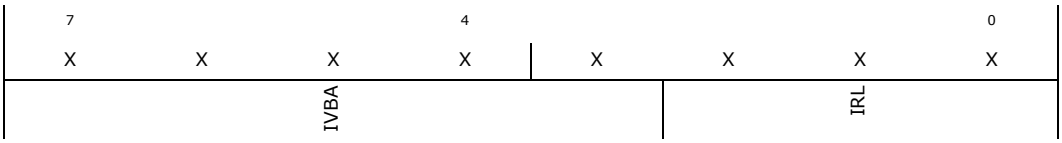
21.12.3.2 Master Initialization Command Word 2 (MICW2)—Offset 21h

Master ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits[7:3] is used by the CPU to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Access Method

Type: I/O Register
(Size: 8 bits)

MICW2: 21h



Bit Range	Default & Access	Description
7: 3	X WO	**Interrupt Vector Base Address (IVBA):** Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.
2: 0	X WO	**Interrupt Request Level (IRL):** When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code: Code Master Interrupt Slave Interrupt 000 IRQ0 IRQ8 001 IRQ1 IRQ9 010 IRQ2 IRQ10 011 IRQ3 IRQ11 100 IRQ4 IRQ12 101 IRQ5 IRQ13 110 IRQ6 IRQ14 111 IRQ7 IRQ15

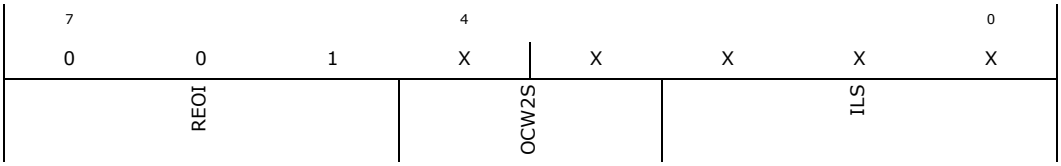
21.12.3.3 Master Operational Control Word 2 (MOCW2)—Offset 24h

Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Access Method

Type: I/O Register
(Size: 8 bits)

MOCW2: 24h



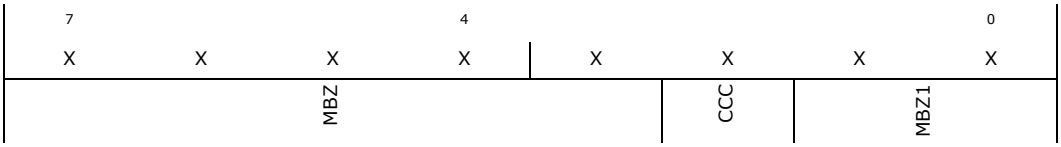
Bit Range	Default & Access	Description
7: 5	001b WO	**Rotate and EOI Codes (REOI):** R, SL, EOI - These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations is listed above under the bit definition. 000 - Rotate in Auto EOI Mode (Clear) 001 - Non-specific EOI command 010 - No Operation 011 - *Specific EOI Command 100 - Rotate in Auto EOI Mode (Set) 101 - Rotate on Non-Specific EOI Command 110 - *Set Priority Command 111 - *Rotate on Specific EOI Command *L0 - L2 Are Used
4: 3	X WO	**OCW2 Select (OCW2S):** When selecting OCW2, bits 4:3 = 00
2: 0	X WO	**Interrupt Level Select (L2, L1, L0) (ILS):** L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined above, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function; programming L2, L1 and L0 to 0 is sufficient in this case. Bits Interrupt Level Bits Interrupt Level 000 IRQ0/8 100 IRQ4/12 001 IRQ1/9 101 IRQ5/13 010 IRQ2/10 110 IRQ6/14 011 IRQ3/11 111 IRQ7/15

21.12.3.4 Master Initialization Command Word 3 (MICW3)—Offset 25h

Access Method

Type: I/O Register
(Size: 8 bits)

MICW3: 25h



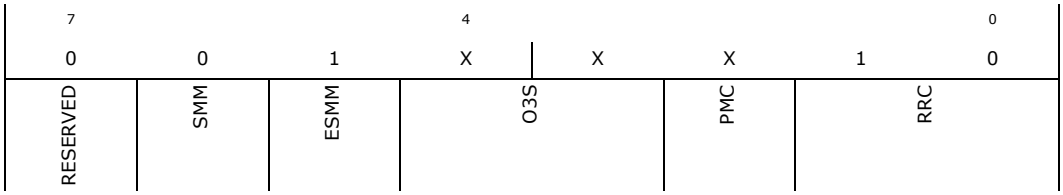
Bit Range	Default & Access	Description
7: 3	X WO	**MBZ (MBZ):** These bits must be programmed to zero.
2	X WO	**Cascaded Controller Connection (CCC):** This bit must always be programmed to a 1 to indicate the slave controller for interrupts 8 15 is cascaded on IRQ2.
1: 0	X WO	**MBZ (MBZ1):** These bits must be programmed to zero.

21.12.3.5 Master Operational Control Word 3 (MOCW3)—Offset 28h

Access Method

Type: I/O Register
(Size: 8 bits)

MOCW3: 28h



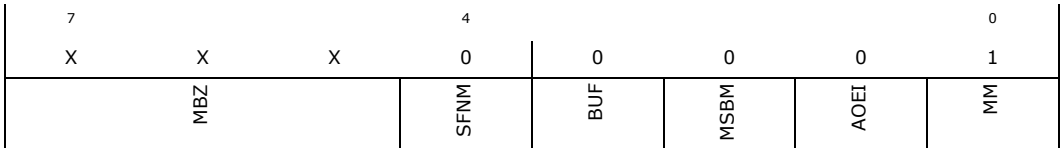
Bit Range	Default & Access	Description
7	0b RO	**RESERVED (RESERVED):** Must be 0.
6	0b WO	**Special Mask Mode (SMM):** If this bit is set, the Special Mask Mode can be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/ disabling of the other channel's mask bits. Bit 6, the ESMM bit, must be set for this bit to have any meaning.
5	1b WO	**Enable Special Mask Mode (ESMM):** When set, the SMM bit is enabled to set or reset the Special Mask Mode. When cleared, the SMM bit becomes a don't care.
4: 3	X WO	**OCW3 Select (O3S):** When selecting OCW3, bits 4:3 = 01
2	X WO	**Poll Mode Command (PMC):** When cleared, poll command is not issued. When set, the next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.
1: 0	10b WO	**Register Read Command (RRC):** These bits provide control for reading the ISR and Interrupt IRR. When bit 1=0, bit 0 will not affect the register read selection. Following ICW initialization, the default OCW3 port address read will be read IRR. To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register can be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. 00 No Action 01 No Action 10 Read IRQ Register 11 Read IS Register

21.12.3.6 Master Initialization Command Word 4 (MICW4)—Offset 29h

Access Method

Type: I/O Register
(Size: 8 bits)

MICW4: 29h



Bit Range	Default & Access	Description
7: 5	X WO	**MBZ (MBZ):** These bits must be programmed to zero.
4	0b WO	**Special Fully Nested Mode (SFNM):** Should normally be disabled by writing a 0 to this bit. If SFNM=1, the special fully nested mode is programmed.
3	0b WO	**Buffered Mode (BUF):** Must be cleared for non-buffered mode. Writing 1 will result in undefined behavior.

Bit Range	Default & Access	Description
2	0b WO	**Master/Slave in Buffered Mode (MSBM):** Not used. Should always be programmed to 0.
1	0b WO	**Automatic End of Interrupt (AOEI):** This bit should normally be programmed to 0. This is the normal end of interrupt. If this bit is 1, the automatic end of interrupt mode is programmed.
0	1b WO	**Microprocessor Mode (MM):** This bit must be written to 1 to indicate that the controller is operating in an Intel Architecture-based system. Writing 0 will result in undefined behavior.

21.12.3.7 Master Operational Control Word 1 (MOCW1)—Offset 2Dh

Access Method

Type: I/O Register
(Size: 8 bits)

MOCW1: 2Dh

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0
IRM							

Bit Range	Default & Access	Description
7: 0	00h RW	**Interrupt Request Mask (IRM):** When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.

21.12.3.8 Slave Initialization Command Word 1 (SICW1)—Offset A0h

.A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

- The Interrupt Mask register is cleared.
- IRQ7 input is assigned priority 7.
- The slave mode address is set to 7.
- Special Mask Mode is cleared and Status Read is set to IRR.

Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Access Method

Type: I/O Register
(Size: 8 bits)

SICW1: A0h

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X
	MCS85		ICWOCWSEL	LTIM	ADI	SNGL	IC4

Bit Range	Default & Access	Description
7: 5	X WO	**MCS85 (MCS85):** These bits are MCS-85 specific, and not needed. Should be programmed to 000
4	X WO	**ICW/OCW select (ICWOCWSEL):** This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.
3	X WO	**Edge/Level Bank Select (LTIM):** Disabled. Replaced by ELCR1 and ELCR2.
2	X WO	**ADI (ADI):** Should be programmed to 0.
1	X WO	**Single or Cascade (SNGL):** Must be programmed to a 0 to indicate two controllers operating in cascade mode.
0	X WO	**wICW4 Write Required (IC4):** This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.

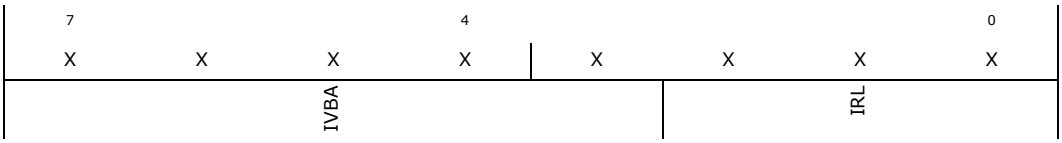
21.12.3.9 Slave Initialization Command Word 2 (SICW2)—Offset A1h

Slave ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits[7:3] is used by the CPU to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Access Method

Type: I/O Register
(Size: 8 bits)

SICW2: A1h



Bit Range	Default & Access	Description
7: 3	X WO	**Interrupt Vector Base Address (IVBA):** Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.
2: 0	X WO	**Interrupt Request Lever (IRL):** When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code: Code Master Interrupt Slave Interrupt 000 IRQ0 IRQ8 001 IRQ1 IRQ9 010 IRQ2 IRQ10 011 IRQ3 IRQ11 100 IRQ4 IRQ12 101 IRQ5 IRQ13 110 IRQ6 IRQ14 111 IRQ7 IRQ15

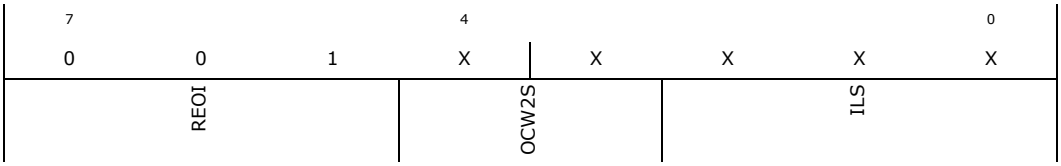
21.12.3.10 Slave Operational Control Word 2 (SoCW2)—Offset A4h

Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Access Method

Type: I/O Register
(Size: 8 bits)

SoC**W2:** A4h



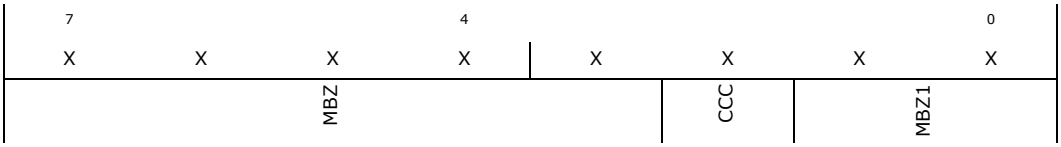
Bit Range	Default & Access	Description
7: 5	001b WO	**Rotate and EOI Codes (REOI):** R, SL, EOI - These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations is listed above under the bit definition. 000 - Rotate in Auto EOI Mode (Clear) 001 - Non-specific EOI command 010 - No Operation 011 - *Specific EOI Command 100 - Rotate in Auto EOI Mode (Set) 101 - Rotate on Non-Specific EOI Command 110 - *Set Priority Command 111 - *Rotate on Specific EOI Command *L0 - L2 Are Used
4: 3	X WO	**OCW2 Select (OCW2S):** When selecting OCW2, bits 4:3 = 00
2: 0	X WO	**Interrupt Level Select (L2, L1, L0) (ILS):** L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined above, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function; programming L2, L1 and L0 to 0 is sufficient in this case. Bits Interrupt Level Bits Interrupt Level 000 IRQ0/8 100 IRQ4/12 001 IRQ1/9 101 IRQ5/13 010 IRQ2/10 110 IRQ6/14 011 IRQ3/11 111 IRQ7/15

21.12.3.11 Slave Initialization Command Word 3 (SICW3)—Offset A5h

Access Method

Type: I/O Register
(Size: 8 bits)

SICW3: A5h



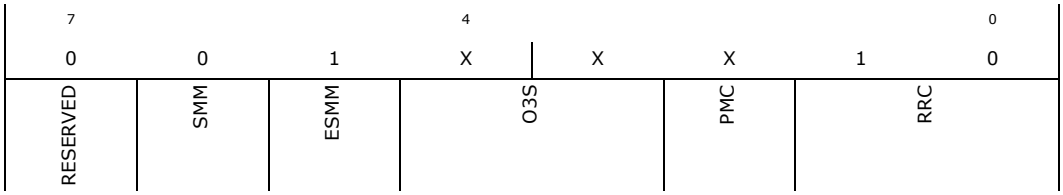
Bit Range	Default & Access	Description
7: 3	X WO	**MBZ (MBZ):** These bits must be programmed to zero.
2	X WO	**Cascaded Controller Connection (CCC):** This bit must always be programmed to a 1 to indicate the slave controller for interrupts 8 15 is cascaded on IRQ2.
1: 0	X WO	**MBZ (MBZ1):** These bits must be programmed to zero.

21.12.3.12 Slave Operational Control Word 3 (SoCW3)—Offset A8h

Access Method

Type: I/O Register
(Size: 8 bits)

SoC**W3:** A8h



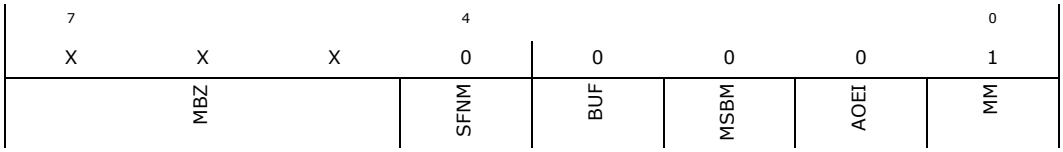
Bit Range	Default & Access	Description
7	0b RO	**RESERVED (RESERVED):** Must be 0.
6	0b WO	**Special Mask Mode (SMM):** If this bit is set, the Special Mask Mode can be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/ disabling of the other channel's mask bits. Bit 6, the ESMM bit, must be set for this bit to have any meaning.
5	1b WO	**Enable Special Mask Mode (ESMM):** When set, the SMM bit is enabled to set or reset the Special Mask Mode. When cleared, the SMM bit becomes a don't care.
4: 3	X WO	**OCW3 Select (O3S):** When selecting OCW3, bits 4:3 = 01
2	X WO	**Poll Mode Command (PMC):** When cleared, poll command is not issued. When set, the next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.
1: 0	10b WO	**Register Read Command (RRC):** These bits provide control for reading the ISR and Interrupt IRR. When bit 1=0, bit 0 will not affect the register read selection. Following ICW initialization, the default OCW3 port address read will be read IRR. To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register can be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. 00 No Action 01 No Action 10 Read IRQ Register 11 Read IS Register

21.12.3.13 Slave Initialization Command Word 4 (SICW4)—Offset A9h

Access Method

Type: I/O Register
(Size: 8 bits)

SICW4: A9h



Bit Range	Default & Access	Description
7: 5	X WO	**MBZ (MBZ):** These bits must be programmed to zero.
4	0b WO	**Special Fully Nested Mode (SFNM):** Should normally be disabled by writing a 0 to this bit. If SFNM=1, the special fully nested mode is programmed.
3	0b WO	**Buffered Mode (BUF):** Must be cleared for non-buffered mode. Writing 1 will result in undefined behavior.

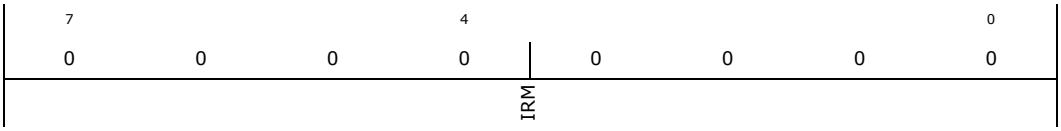
Bit Range	Default & Access	Description
2	0b WO	**Master/Slave Buffered Mode (MSBM):** Not used. Should always be programmed to 0.
1	0b WO	**Automatic End of Interrupt (AOEI):** This bit should normally be programmed to 0. This is the normal end of interrupt. If this bit is 1, the automatic end of interrupt mode is programmed.
0	1b WO	**Microprocessor Mode (MM):** This bit must be written to 1 to indicate that the controller is operating in an Intel Architecture-based system. Writing 0 will result in undefined behavior.

21.12.3.14 Slave Operational Control Word 1 (SoCW1)—Offset ADh

Access Method

Type: I/O Register
(Size: 8 bits)

SoC**W1:** ADh



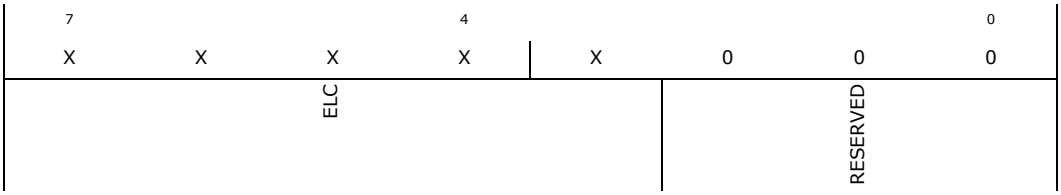
Bit Range	Default & Access	Description
7: 0	00h RW	**Interrupt Request Mask (IRM):** When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.

21.12.3.15 Master Edge/Level Control (ELCR1)—Offset 4D0h

Access Method

Type: I/O Register
(Size: 8 bits)

ELCR1: 4D0h



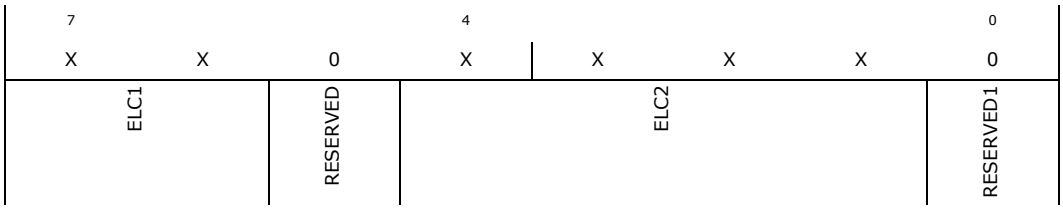
Bit Range	Default & Access	Description
7: 3	X RW	**Edge Level Control (ECL[7:3]) (ELC):**: In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level.
2: 0	0b RO	**RESERVED (RESERVED):** Reserved.

21.12.3.16 Slave Edge/Level Control (ELCR2)—Offset 4D1h

Access Method

Type: I/O Register
(Size: 8 bits)

ELCR2: 4D1h



Bit Range	Default & Access	Description
7: 6	X RW	**Edge Level Control (ECL[15:14]) (ELC1):** In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level. Bit 7 applies to IRQ15, and bit 6 to IRQ14.
5	0b RO	**RESERVED (RESERVED):** Reserved.
4: 1	X RW	**Edge Level Control (ECL[12:9]) (ELC2):**: In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level. Bit 4 applies to IRQ12, bit 3 to IRQ11, bit 2 to IRQ10, and bit 1 to IRQ9.
0	0b RO	**RESERVED (RESERVED):** Reserved.

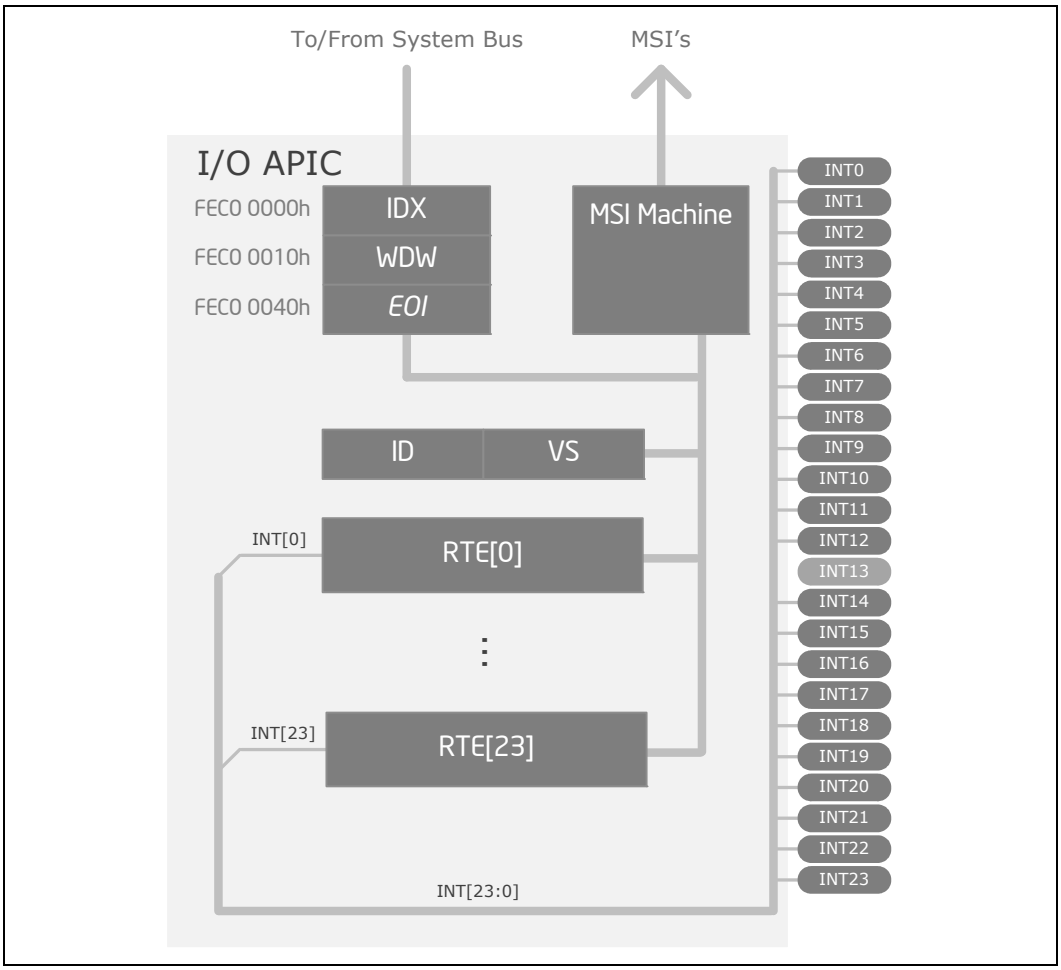
21.13 I/O APIC

The I/O Advanced Programmable Interrupt Controller (APIC) is used to support line interrupts more flexibly than the 8259 PIC. Line interrupts are routed to it from multiple sources, including legacy devices, via the interrupt decoder or they are routed to it from the interrupt router in the Legacy Bridge. These line-based interrupts are then used to generate interrupt messages targeting the local APIC in the processor.

21.13.1 Features

- 24 interrupt lines
  - — IRQ0-23
- Edge or level trigger mode per interrupt
- Active low or high polarity per interrupt
- Works with local APIC in processor via MSIs
- MSIs can target specific processor core
- Established APIC programming model

Figure 52. Detailed I/O APIC Block Diagram

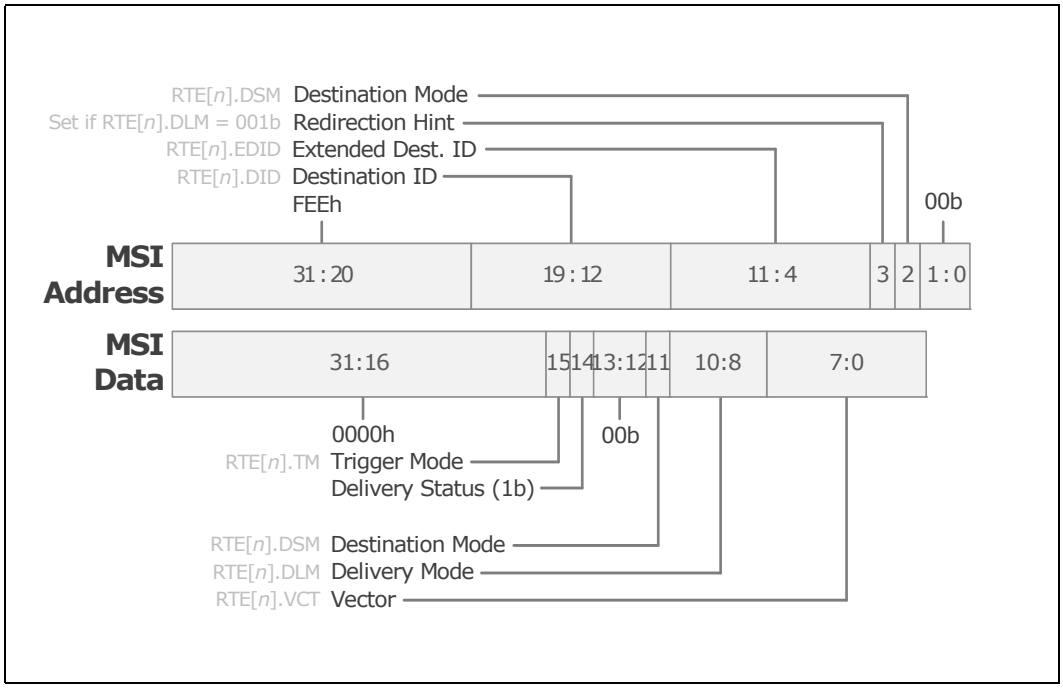


Note: INT13 is unavailable and is effectively tied low within the I/O APIC, INT14 & INT15 are unused in the

SoC and are tied low.

MSIs generated by the I/O APIC are sent as 32-bit memory writes to the Local APIC. The address and data of the write transaction are used as follows.

Figure 53. MSI Address and Data



Destination ID (DID) and Extended Destination ID (EDID) are used to target a specific processor core's local APIC.

21.13.2 Use

The I/O APIC contains indirectly accessed I/O APIC registers and normal memory mapped registers. There are three memory mapped registers:

- Index Register (IDX)
- Window Register (WDW)
- End Of Interrupt Register (EOI)

The Index register selects an indirect I/O APIC register (ID/VS/RTE[*n*]) to appear in the Window register.

The Window register is used to read or write the indirect register selected by the Index register.

The EOI register is written to by the Local APIC in the processor. The I/O APIC compares the lower eight bits written to the EOI register to the Vector set for each interrupt (RTE.VCT). All interrupts that match this vector will have their RTE.RIRR register cleared. All other EOI register bits are ignored.

21.13.3 Unsupported Modes

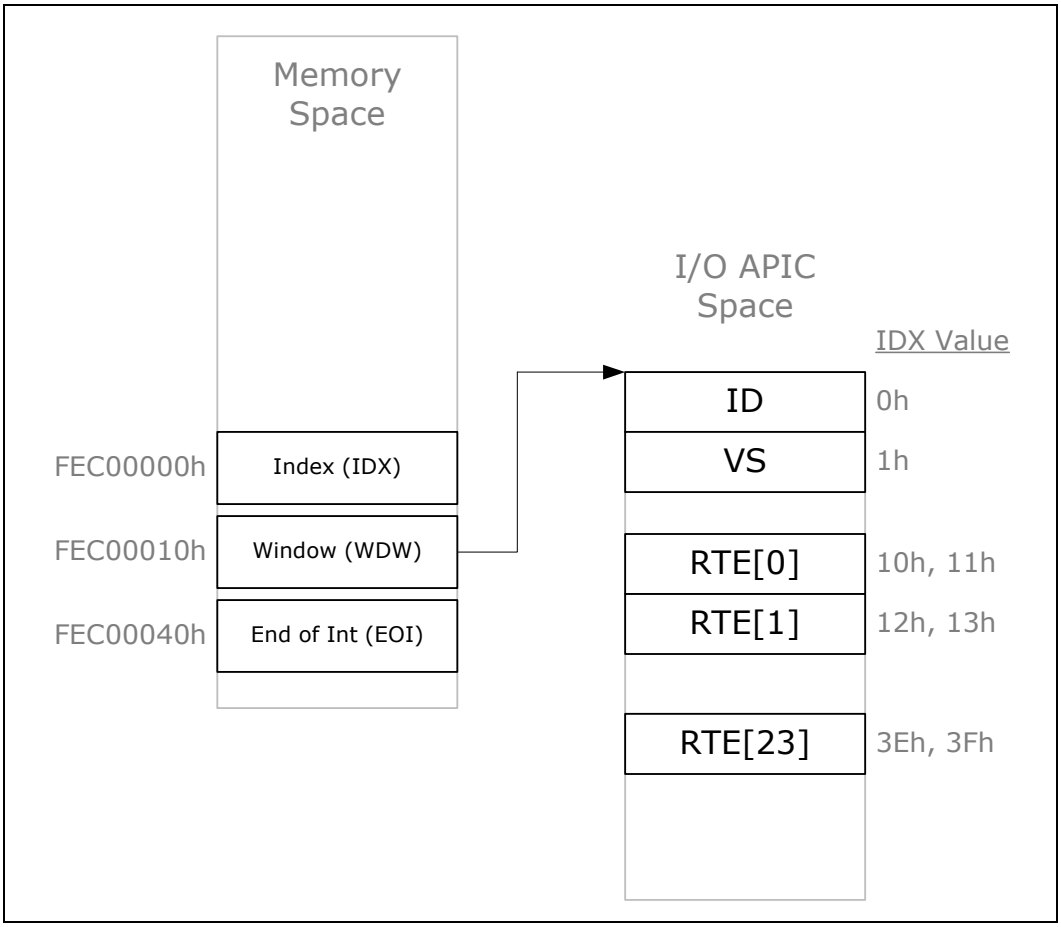
These delivery modes are not supported for the following reasons:

- **NMI/INIT:** This cannot be delivered while the CPU is in the Stop Grant state. In addition, this is a break event for power management.
- **SMI:** There is no way to block the delivery of the SMI#, except through BIOS.
- **Virtual Wire Mode B:** The Legacy Bridge does not support the INTR of the 8259 routed to the I/OxAPIC pin 0.

21.13.4 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 54. I/O APIC Register Map



21.13.5 Memory Mapped Registers

The APIC is accessed via an indirect addressing scheme. These registers are mapped into memory space. The registers are shown below.

Table 134. I/O APIC Memory Mapped Registers

Address	Symbol	Register
FEC00000h	IDX	Index Register
FEC00010h	WDW	Window Register
FEC00040h	EOI	End of Interrupt Register

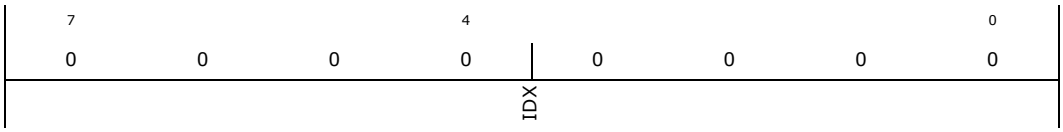
21.13.5.1 Index Register (IDX)—Offset FEC00000h

Access Method

Type: Memory Mapped I/O Register (Size: 8 bits)

IDX: FEC00000h

Default: 00h



Bit Range	Default & Access	Description
7: 0	0h RW	**Index (IDX):** This 8-bit register selects which indirect register appears in the window register to be manipulated by software. Software will program this register to select the desired APIC internal register.

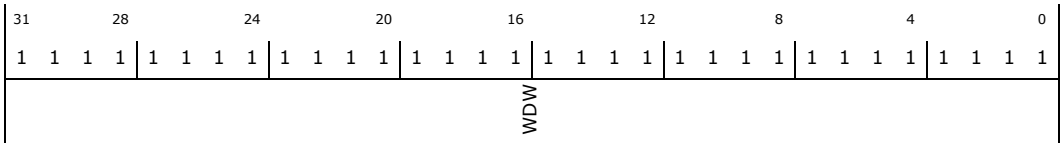
21.13.5.2 Window Register (WDW)—Offset FEC00010h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

WDW: FEC00010h

Default: FFFFFFFFh



Bit Range	Default & Access	Description
31: 0	FFFFFFFFh RW	**Window (WDW):** This 32-bit register specifies the data to be read or written to the register pointed to by the IDX register. This register can be accessed only in DW quantities.

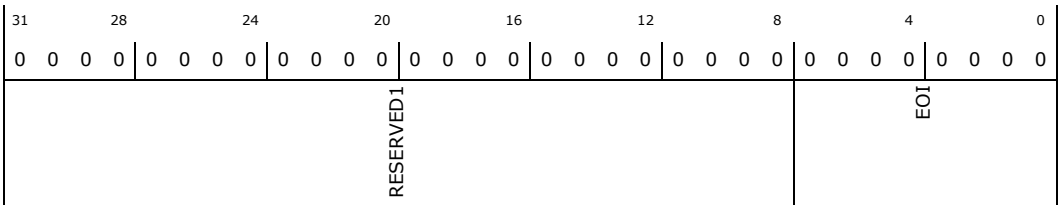
21.13.5.3 End of Interrupt Register (EOI)—Offset FEC00040h

Access Method

Type: Memory Mapped I/O Register (Size: 32 bits)

EOI: FEC00040h

Default: 00000000h



Bit Range	Default & Access	Description
31: 8	0b RO	**RESERVED (RESERVED1):** Reserved.
7: 0	0h RO	**EOI (EOI):** When a write is issued to this register, the IOxAPIC will check the lower 8 bits written to this register, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, RTE.RIRR for that entry will be cleared. If multiple entries have the same vector, each of those entries will have RTE.RIRR cleared.

21.13.6 Index Registers

These registers are selected with the IDX register, and read/written through the WDW register. Accessing these registers must be done as DW requests, otherwise unspecified behavior will result. Software should not attempt to write to reserved registers. Reserved registers may return non-zero values when read.

Note: There is one pair of redirection (RTE) registers per interrupt line. Each pair forms a 64-bit RTE register.

Note: Specified offsets should be placed in IDX, not added to IDX.

Table 135. Index Registers

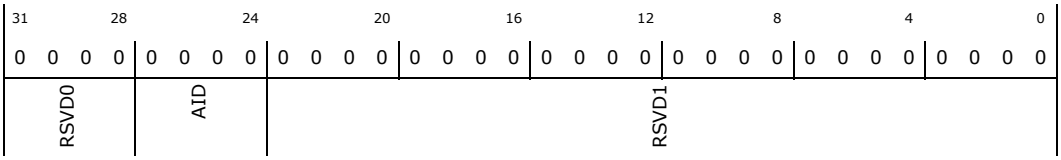
Offset	Symbol	Register
00	ID	Identification
01	VS	Version
02-0F	-	Reserved
10-11	RTE0	Redirection Table 0
12-13	RTE1	Redirection Table 1
...	...	...
3E-3F	RTE23	Redirection Table 23
40-FF	-	Reserved

21.13.6.1 Identification Register (ID)—Offset 0h

Access Method

Type: Indirect I/O APIC Register (Size: 32 bits) **ID:** 0h

Default: 00000000h



Bit Range	Default & Access	Description
31: 28	0h RW	**Reserved (RSVD0):** Reserved.
27: 24	0h RW	**APIC Identification (AID):** Software must program this value before using the APIC.
23: 0	0h RW	**Reserved (RSVD1):** Reserved.

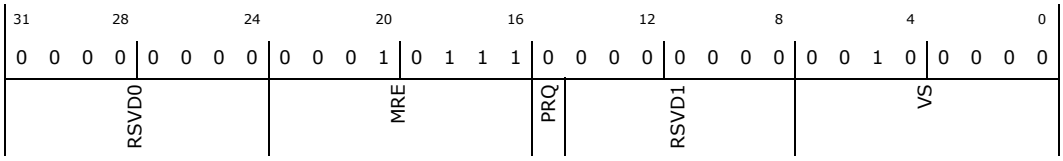
21.13.6.2 Version Register (VS)—Offset 1h

Access Method

Type: Indirect I/O APIC Register (Size: 32 bits)

VS: 1h

Default: 00170020h



Bit Range	Default & Access	Description
31: 24	0h RW	**Reserved (RSVD0):** Reserved.
23: 16	17h RO	**Maximum Redirection Entries (MRE):** This is the entry number (0 being the lowest entry) of the highest entry in the redirection table. This field is hardwired to indicate the total number of interrupts.
15	0b RO	**Pin Assertion Register Supported (PRQ):** The I/O APIC does not implement the Pin Assertion Register.
14: 8	0h RW	**Reserved (RSVD1):** Reserved.
7: 0	20h RO	**Version (VS):** Identifies the implementation version as I/O APIC.

21.13.6.3 Redirection Table Entry Lower (RTE[0-23]L)—Offset 10h - 3Eh

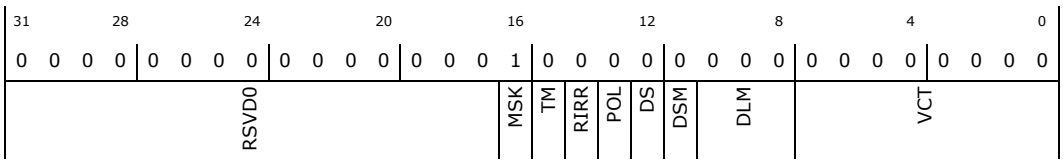
Lower 32-bits of the RTE register.

Access Method

Type: Indirect I/O APIC Register (Size: 32 bits)

RTE[0-23]L: 10h - 3Eh

Default: 00010000h



Bit Range	Default & Access	Description
31: 17	0h RW	**Reserved (RSVD0):** Reserved.
16	1b RW	**Mask (MSK):** When set, interrupts are not delivered nor held pending. When cleared, and edge or level on this interrupt results in the delivery of the interrupt.
15	0b RW	**Trigger Mode (TM):** When cleared, the interrupt is edge sensitive. When set, the interrupt is level sensitive.
14	0b RW	**Remote IRR (RIRR):** This is used for level triggered interrupts; its meaning is undefined for edge triggered interrupts. This bit is set when I/O APIC sends the level interrupt message to the CPU. This bit is cleared when an EOI message is received that matches the VCT field. This bit is never set for SMI, NMI, INIT, or ExtINT delivery modes.
13	0b RW	**Polarity (POL):** This specifies the polarity of each interrupt input. When cleared, the signal is active high. When set, the signal is active low. 0: Active High 1: Active Low
12	0b RO	**Delivery Status (DS):** This field contains the current status of the delivery of this interrupt. When set, an interrupt is pending and not yet delivered. When cleared, there is no activity for this entry.
11	0b RW	**Destination Mode (DSM):** This field is used by the local APIC to determine whether it is the destination of the message.
10: 8	0h RW	**Delivery Mode (DLM):** This field specifies how the APICs listed in the destination field should act upon reception of this signal. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. These encodings are: 000: Fixed 001: Lowest Priority 010: SMI - Not supported. 011: Reserved 100: NMI - Not supported. 101: INIT - Not supported. 110: Reserved 111: ExtINT
7: 0	0h RW	**Vector (VCT):** This field contains the interrupt vector for this interrupt. Values range between 10h and FEh.

21.13.6.4 Redirection Table Entry Upper (RTE[0-23]U)—Offset 11h - 3Fh

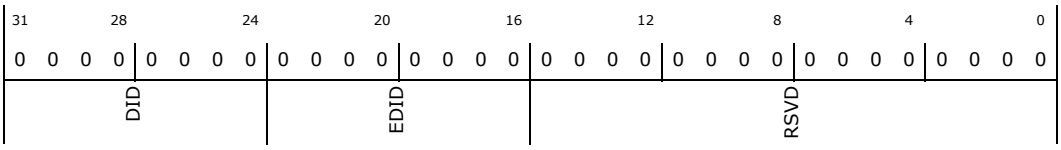
Upper 32-bits of the RTE register.

Access Method

Type: Indirect I/O APIC Register (Size: 32 bits)

RTE[0-23]U: 11h - 3Fh

Default: 00000000h



Bit Range	Default & Access	Description
31: 24	0h RW	**Destination ID (DID):** Destination ID of the local APIC.
23: 16	0h RW	**Extended Destination ID (EDID):** Extended destination ID of the local APIC.

Bit Range	Default & Access	Description
15: 0	0h RW	**Reserved (RSVD):** Reserved.

21.14 Watchdog Timer

The Watchdog timer can be used to trigger a reset in the event that the system has become unresponsive.

21.14.1 Features

Selectable Prescaler - approximately 1 MHz (1 μs to 1 s) and approximately 1 KHz (1 ms to 17 min)

- 33 MHz Clock (30 ns Clock Ticks)
- WDT Mode:
  - Second stage drives WDT_TOUT high or inverts the previous value. Used only after first timeout occurs.
  - Status bit preserved in RTC well for possible error detection and correction
  - Drives WDT_TOUT if OUTPUT is enabled
- Timer can be disabled (default state) or Locked (Hard Reset required to disable WDT)
- WDT Automatic Reload of Preload value when WDT Reload Sequence is performed

Note: WDT_TOUT is not available as a top-level SoC output.

21.14.2 Use

This Watchdog timer provides a resolution that ranges from 1 μs to ~17 minutes. The timer uses a 35-bit down-counter.

The counter is loaded with the value from the 1st Preload register. The timer is then enabled and it starts counting down. The time at which the WDT first starts counting down is called the first stage. If the host fails to reload the WDT before the 35-bit down counter reaches zero the WDT generates an internal interrupt within the WDT.

After the internal interrupt is generated when the first stage has counted down to zero, the WDT loads the value from the 2nd Preload register into the WDT's 35-bit Down-Counter and starts counting down. The WDT is now in the second stage. If the host still fails to reload the WDT before the second timeout, the WDT drives the WDT_TOUT signal high and sets the timeout bit (WDT_TIMEOUT). This bit indicates that the System has become unstable. The WDT_TOUT signal is held high until the system is Reset or the WDT times out again (Depends on WDT Timeout Configuration). The process of reloading the WDT involves the following sequence of writes:

- Write 80 to offset WDTBA + 0Ch
- Write 86 to offset WDTBA + 0Ch
- Write 1 to WDT_RELOAD in Reload Register.

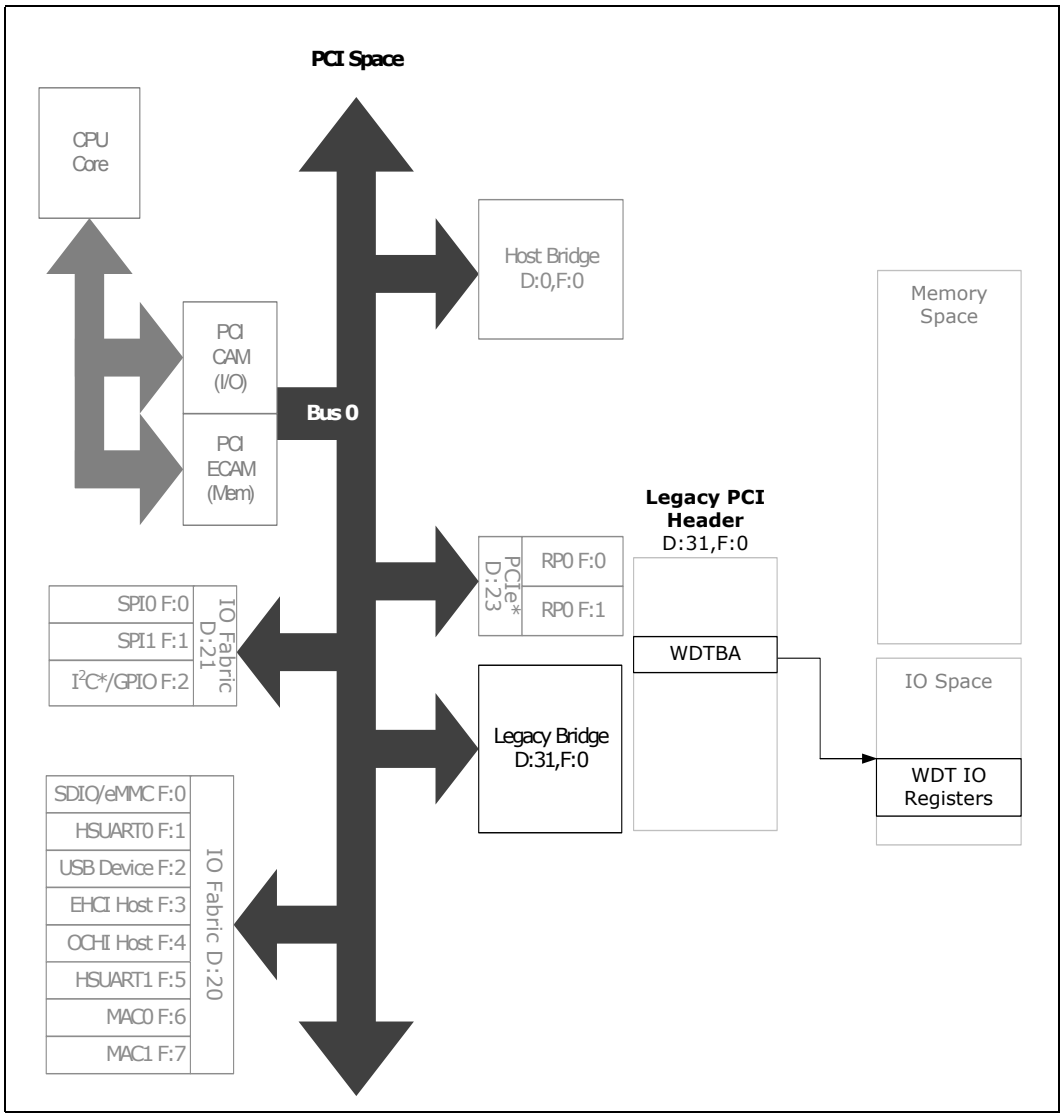
The same process is used for setting the values in the preload registers. The only difference exists in step 3. Instead of writing a '1' to the WDT_RELOAD, you write the desired preload value into the corresponding Preload register. This value is not loaded into the 35-bit down counter until the next time the WDT reenters the stage. For example, if Preload Value 2 is changed, it is not loaded into the 35-bit down counter until the next time the WDT enters the second stage.

Note: The WDT output, WDT_TOUT, is not available as a top-level SoC output pin

21.14.3 Register Map

See Chapter 5.0, "Register Access Methods" for additional information.

Figure 55. Watchdog Timer Register Map



21.14.4 I/O Mapped Registers

Table 136. Summary of I/O Registers—WDTBA

Offset Start	Offset End	Register ID—Description	Default Value
0h	0h	"Preload Value 1 Register 0 (PV1R0)—Offset 0h" on page 913	FFh
1h	1h	"Preload Value 1 Register 1 (PV1R1)—Offset 1h" on page 913	FFh
2h	2h	"Preload Value 1 Register 2 (PV1R2)—Offset 2h" on page 914	0Fh

Table 136. Summary of I/O Registers—WDTBA (Continued)

Offset Start	Offset End	Register ID—Description	Default Value
4h	4h	"Preload Value 2 Register 0 (PV2R0)—Offset 4h" on page 914	FFh
5h	5h	"Preload Value 2 Register 1 (PV2R1)—Offset 5h" on page 915	FFh
6h	6h	"Preload Value 2 Register 2 (PV2R2)—Offset 6h" on page 915	0Fh
Ch	Ch	"Reload Register 0 (RR0)—Offset Ch" on page 915	00h
Dh	Dh	"Reload Register 1 (RR1)—Offset Dh" on page 916	00h
10h	10h	"WDT Configuration Register (WDTCR)—Offset 10h" on page 916	00h
18h	18h	"WDT Lock Register (WDTLR)—Offset 18h" on page 917	00h

21.14.4.1 Preload Value 1 Register 0 (PV1R0)—Offset 0h

Access Method

Type: I/O Register
(Size: 8 bits)

PV1R0: [WDTBA] + 0h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: FFh

7			4				0
1	1	1	1	1	1	1	1
PV1							

Bit Range	Default & Access	Description
7:0	FFh RW	**Preload Value 1[7:0] (PV1):** This register is used to hold the bits 7 down to 0 of the Preload Value 1 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.2 Preload Value 1 Register 1 (PV1R1)—Offset 1h

Access Method

Type: I/O Register
(Size: 8 bits)

PV1R1: [WDTBA] + 1h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: FFh

7			4				0
1	1	1	1	1	1	1	1
PV1							

Bit Range	Default & Access	Description
7:0	FFh RW	**Preload Value 1[15:8] (PV1):** This register is used to hold the bits 15 down to 8 of the Preload Value 1 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.3 Preload Value 1 Register 2 (PV1R2)—Offset 2h

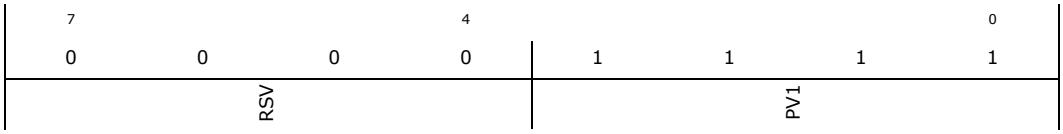
Access Method

Type: I/O Register (Size: 8 bits)

PV1R2: [WDTBA] + 2h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 0Fh



Bit Range	Default & Access	Description
7:4	0b RO	**Reserved (RSV):** Reserved.
3:0	Fh RW	**Preload Value 1[19:16] (PV1):** This register is used to hold the bits 19 down to 16 of the Preload Value 1 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.4 Preload Value 2 Register 0 (PV2R0)—Offset 4h

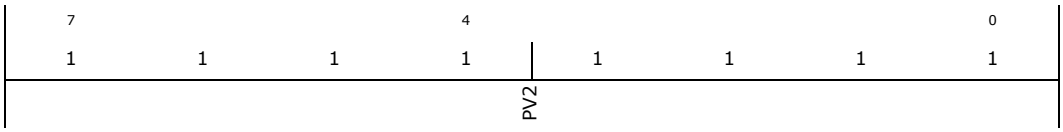
Access Method

Type: I/O Register (Size: 8 bits)

PV2R0: [WDTBA] + 4h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: FFh



Bit Range	Default & Access	Description
7:0	FFh RW	**Preload Value 2[7:0] (PV2):** This register is used to hold the bits 7 down to 0 of the Preload Value 2 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.5 Preload Value 2 Register 1 (PV2R1)—Offset 5h

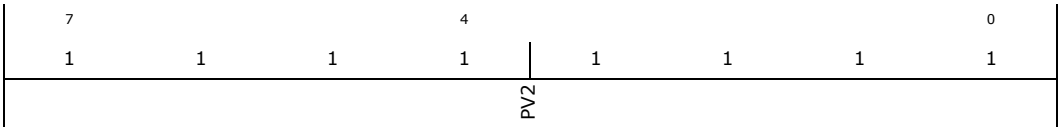
Access Method

Type: I/O Register
(Size: 8 bits)

PV2R1: [WDTBA] + 5h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: FFh



Bit Range	Default & Access	Description
7:0	FFh RW	**Preload Value 2[15:8] (PV2):** This register is used to hold the bits 15 down to 8 of the Preload Value 2 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.6 Preload Value 2 Register 2 (PV2R2)—Offset 6h

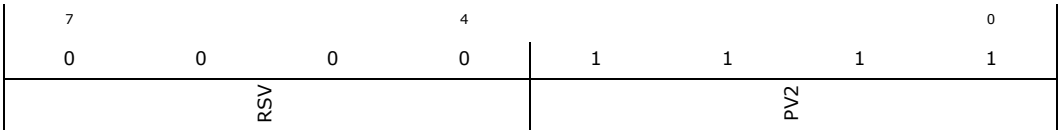
Access Method

Type: I/O Register
(Size: 8 bits)

PV2R2: [WDTBA] + 6h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 0Fh



Bit Range	Default & Access	Description
7:4	0b RO	**Reserved (RSV):** Reserved.
3:0	Fh RW	**Preload Value 2[19:16] (PV2):** This register is used to hold the bits 19 down to 16 of the Preload Value 2 for the Watch Dog Timer. The Value in the Preload Register is Automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (i.e. zero is counted as part of the decrement).

21.14.4.7 Reload Register 0 (RR0)—Offset Ch

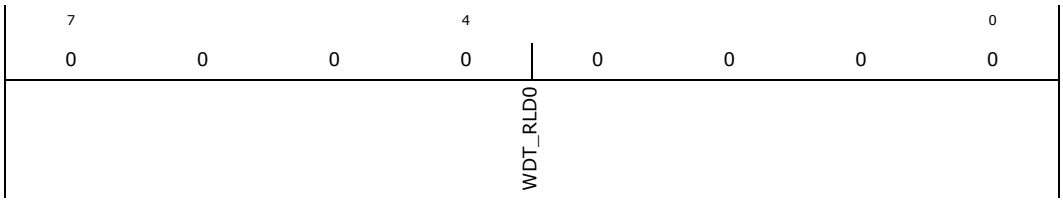
Access Method

Type: I/O Register
(Size: 8 bits)

RR0: [WDTBA] + Ch

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 00h



Bit Range	Default & Access	Description
7:0	0b WO	**WDT Reload 0 (WDT_RLD0):** The reload sequence is only necessary for the Reload register and Preload Value registers and is not used in Free Running mode.

21.14.4.8 Reload Register 1 (RR1)—Offset Dh

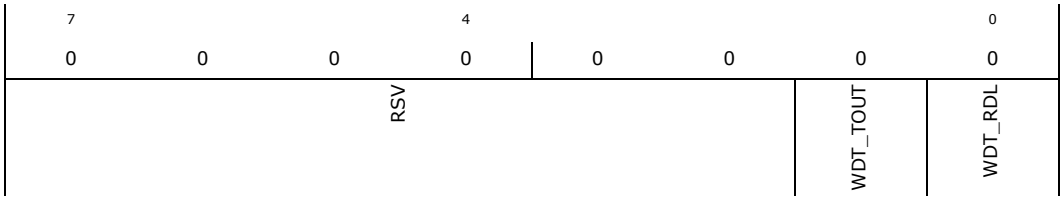
Access Method

Type: I/O Register
(Size: 8 bits)

RR1: [WDTBA] + Dh

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 00h



Bit Range	Default & Access	Description
7:2	0b RO	**Reserved (RSV):** Reserved.
1	0b RW/1C	**WDT Timeout (WDT_TOUT):** This bit is located in the RTC Well and its value is not lost if the host resets the system. It is set to 1 if the host fails to reset the WDT before the 35-bit Down-Counter reaches zero for the second time in a row. This bit is cleared by performing the Register Unlocking Sequence followed by a 1 to this bit. 0 = Normal (Default) 1 = System has become unstable.
0	0b RW	**WDT Reload (WDT_RDL):** To prevent a timeout the host must perform the Register Unlocking Sequence followed by a 1 to this bit.

21.14.4.9 WDT Configuration Register (WDTCR)—Offset 10h

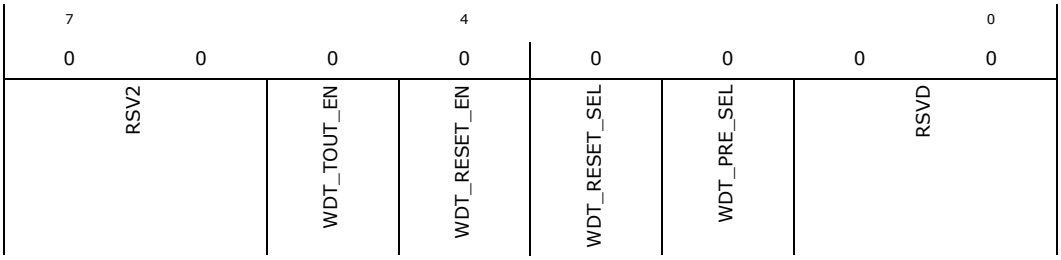
Access Method

Type: I/O Register
(Size: 8 bits)

WDTCR: [WDTBA] + 10h

WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 00h



Bit Range	Default & Access	Description
7:6	0b RO	**Reserved (RSV2):** Reserved.
5	0b RW	**WDT Timeout Output Enable (WDT_TOUT_EN):** This bit indicates whether or not the WDT toggles the external WDT_TOUT signal if the WDT times out. 0 = Enabled (Default) 1 = Disabled
4	0b RW	**WDT Reset Enable (WDT_RESET_EN):** When this bit is enable (set to 1), it allows internal reset to be trigger when WDT timeout in the second stage. It either trigger COLD or WARM reset depend on WDT_RESET_SEL bit. 0 = Disable internal reset (Default) 1 = Enable internal COLD or WARM reset.
3	0b RW	**WDT Reset Select (WDT_RESET_SEL):** This determines which reset to be triggered when WDT_RESET_EN is set. 0 = Cold Reset (Default) 1 = Warm Reset
2	0b RW	**WDT Prescaler Select (WDT_PRE_SEL):** The WDT provides two options for prescaling the main Down Counter. The preload values are loaded into the main down counter right justified. The prescaler adjusts the starting point of the 35-bit down counter. 0 = The 20-bit Preload Value is loaded into bits 34:15 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^15. The approximate clock generated is 1 KHz, (1 ms to 10 min). (Default) 1 = The 20-bit Preload Value is loaded into bits 24:05 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^5. The approximate clock generated is 1 MHz, (1 us to 1sec)
1:0	0b RO	**Reserved (RSVD):** Reserved.

21.14.4.10 WDT Lock Register (WDTLR)—Offset 18h

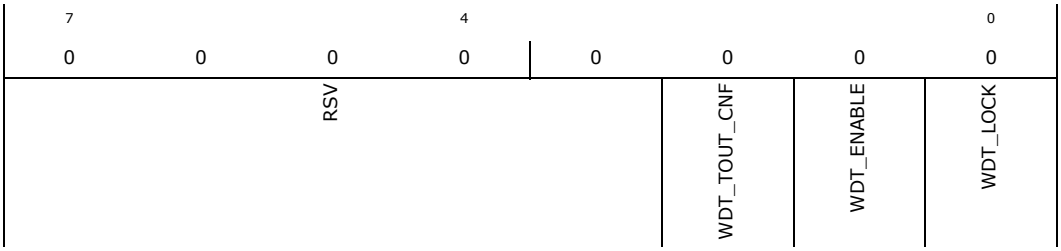
Access Method

Type: I/O Register
(Size: 8 bits)

WDTLR: [WDTBA] + 18h

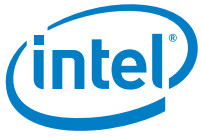
WDTBA Type: PCI Configuration Register (Size: 32 bits)
WDTBA Reference: [B:0, D:31, F:0] + 84h

Default: 00h



Bit Range	Default & Access	Description
7:3	0b RO	**Reserved (RSV):** Reserved.
2	0b RW	**WDT Timeout Configuration (WDT_TOUT_CNF):** This register is used to choose the functionality of the timer. 0 = Watchdog Timer Mode: When enabled (i.e. WDT_ENABLE goes from 0 to 1) the timer reloads Preload Value 1 and start decrementing. (Default) Upon reaching the second stage timeout the WDT_TOUT is driven high once and does not change again until Power is cycled or a hard reset occurs. 1 = Reserved
1	0b RW	**Watchdog Timer Enable (WDT_ENABLE):** The following bit enables or disables the WDT. 0 = Disabled (Default) 1 = Enabled Note: This bit cannot be modified if WDT_LOCK has been set. Note: In WDT mode Preload Value 1 is reloaded every time WDT_ENABLE goes from 0 to 1 or the WDT_RELOAD bit is written using the proper sequence of writes (See Register Unlocking Sequence). When the WDT second stage timeout occurs, a reset must happen. Note: Software must guarantee that a timeout is not about to occur before disabling the timer. A reload sequence is suggested.
0	0b RW/O	**Watchdog Timer Lock (WDT_LOCK):** Setting this bit locks the values of this register until a hard-reset occurs or power is cycled. 0 = Unlocked (Default) 1 = Locked Note: Writing a 0 has no effect on this bit. Write is only allowed from 0 to 1 once. It cannot be changed until either power is cycled or a hard reset occurs

§ §



22.0 Debug Port and JTAG/TAP

The Intel® Quark SoC X1000 provides a TAP (Test Access Port) Controller interface that aids in the debug and test of the SoC. The TAP Controller interface can be used for debug by attaching a JTAG-based debugger. The SoC supports the open source Open On-Chip Debugger (OpenOCD) tools.

22.1 Signal Descriptions

Please see Chapter 2.0, "Physical Interfaces" for additional details.

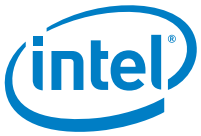
The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction**: The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4.0, "Electrical Characteristics"
- **Description**: A brief explanation of the signal's function

Table 137. Debug Port and JTAG/TAP Signals

Signal Name	Direction/ Type	Description
TCK	I CMOS3.3	**JTAG Test Clock** Provides the clock input for TAP controller. Maximum frequency is 25 MHz.
TDI	I CMOS3.3	**JTAG Test Data In** TDI is used to serially shift data and instructions into the TAP.
TDO	O CMOS3.3	**JTAG Test Data Out** TDO is used to serially shift data out of the TAP.
TMS	I CMOS3.3	**JTAG Test Mode Select** This signal is used to control the state of the TAP controller.
TRST_B	I CMOS3.3	**JTAG Test Reset** Asynchronously resets the TAP logic.
PREQ_B	I CMOS3.3	**Probe Mode Request** Used to request entry into Probe Mode.
PRDY_B	O CMOS3.3	**Probe Mode Ready** Indicates Probe Mode has been entered.

Note: PREQ_B and PRDY_B are not needed for OpenOCD-based debug.



22.2 Features

22.2.1 OpenOCD

Providing a JTAG-only based debug solution allows standard off the shelf USB-to-JTAG dongles to be used for debug. With a suitable JTAG adapter, OpenOCD can be used with the SoC to provide the following debug features:

- Halt/Break
- Step Through
- Register Read/Write
- Memory Read/Write
- Legacy SPI Flash Programming

This allows for a low cost, open source debug solution for the Inte® Quark SoC X1000.

§ §