

# AMD-8111™ HyperTransport™ I/O Hub



## Data Sheet

The AMD Athlon™ 64 and AMD Opteron™ processors power the next generation in computing platforms, designed to deliver the ultimate performance for cutting-edge applications and an unprecedented computing experience. The AMD-8000™ series of chipset components is a highly integrated system logic solution that delivers enhanced performance and features for the AMD Athlon 64 and AMD Opteron processors.

The AMD-8111™ HyperTransport™ I/O hub includes the following features:

- **HyperTransport Technology Link**
  - Supports up to 800 megabytes per second of total bandwidth, using 8-bit HyperTransport input and output links running simultaneously with a 200 MHz (double pumped) clock
  - Supports multiple bit widths, including eight bits, four bits, and two bits (input and output)
  - Supports a 200-MHz(double pumped) HyperTransport clock
- **PCI Bus**
  - Utilizes a 33-MHz, 32-bit interface
  - PCI version 2.2 compliant
  - Includes PCI bus arbiter with support for up to eight external devices
- **AC '97 Support**
  - AC '97 version 2.2 compatible
  - Soft modem and 6-channel soft audio interface
- **Advanced Communication Riser (ACR) Rev. 1.0 Support**
- **Ethernet LAN Controller**
  - 10/100-Mbit/s
  - Uses MII interface to connect to the Ethernet PHY
- **System Management Bus**
  - One System Management Bus 1.0 host controller
  - One System Management Bus 2.0 host controller
- **USB Support for Six Ports**
  - USB 1.1 support provided by two OHCI-based USB hosts, each supporting three ports
  - USB 2.0 support provided by one EHCI-based host, which supports all six ports
- **Enhanced IDE Controller**
  - Support for a primary and a secondary dual-drive port
  - PIO modes 0–4, multi-word DMA modes 0–2, UDMA modes 0–6 (through to ATA-133), and ATAPI
  - Two independent controllers for DMA accesses
- **LPC Bus**
  - Connects peripherals such as super I/O and BIOS
- **High Precision Event Timer**
  - Supports one 32-bit counter with one periodic and two non-periodic timers
- **Serial IRQ Protocol**
- **Extensive ACPI-Compliant Power Management**
  - Programmable C2, C3, power-on-suspend, suspend to RAM, suspend to disk, and soft off states
  - Throttling
  - Device monitors
  - Hardware traps
  - System inactivity timers
- **Thirty-Two General Purpose I/O (GPIO) Pins**
  - Many are multiplexed with other hard-wired functions
- **Privacy/Security Logic; ROM Access Control**

- **Legacy AT-Compatible Logic**
  - Programmable interrupt controller
  - Programmable interval timer
  - DMA controller (LPC bus)
  - Legacy ports
- **IOAPIC controller**
- **Real-time Clock**
  - Includes 256 bytes of CMOS, battery-powered RAM, and ACPI-compliant extensions
- **Battery-Powered RAM**
  - 256 bytes for storing CPU memory controller context
- **Random Number Generator**
- **492-Pin BGA; 26-by-26 BGA Grid; 35-by-35 Millimeters Square**
- **1.8-V Core; 3.3-V Output Drivers; 5-V Tolerant Input Buffers**

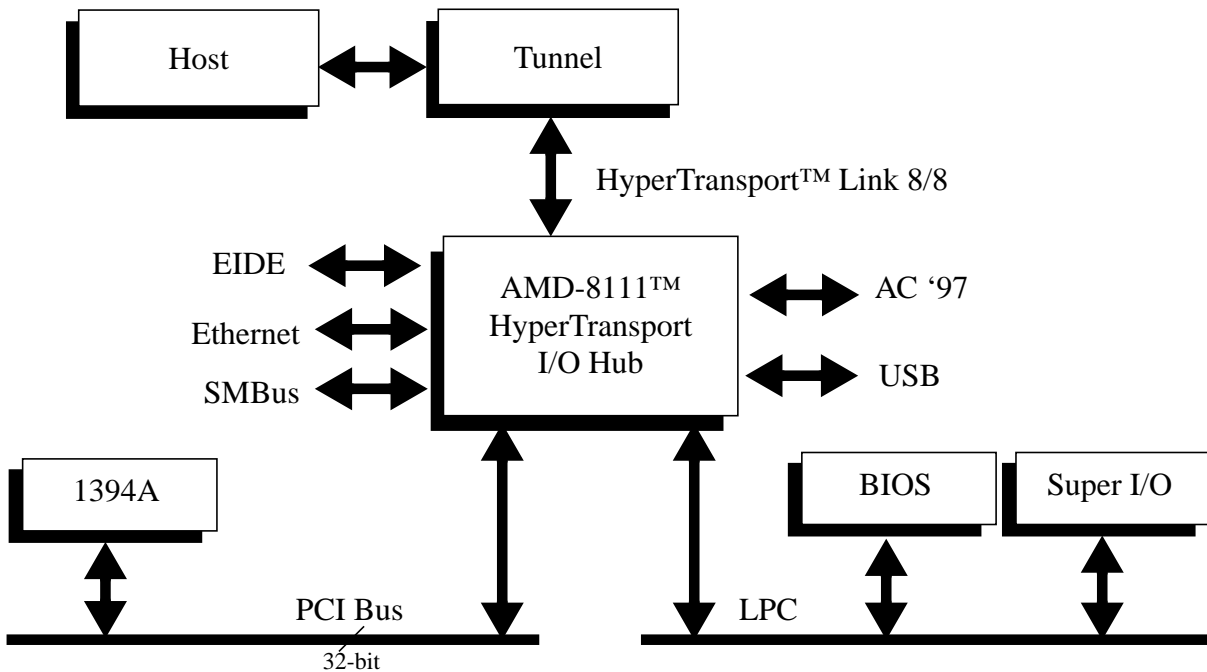


Figure 1. System Block Diagram

---

**Trademarks**

AMD, the AMD Arrow logo, and combinations thereof, AMD Athlon, AMD Opteron AMD-8000, AMD-8111, Magic Packet, and PCnet are trademarks of Advanced Micro Devices, Inc.

HyperTransport is a licensed trademark of the HyperTransport Technology Consortium.

Microsoft is a registered trademark of Microsoft Corporation.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

**Disclaimer**

The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

© 2001 - 2003 Advanced Micro Devices, Inc. All rights reserved.



# Table of Contents

---

<b>Chapter 1</b>	<b>Ordering Information</b>	<b>17</b>
<b>Chapter 2</b>	<b>Signal Descriptions</b>	<b>19</b>
2.1	Terminology	19
2.2	Host HyperTransport™ Technology Interface	20
2.3	Secondary PCI Interface	21
2.4	LPC Bus and Legacy Support Pins	22
2.5	Ultra DMA Enhanced IDE Interface	23
2.6	System Management Pins	23
2.7	Universal Serial Bus Interface	29
2.8	AC '97 Interface	30
2.9	MII Interface	31
2.10	Test	33
2.11	Miscellaneous	34
2.12	Power and Ground	34
<b>Chapter 3</b>	<b>Functional Operation</b>	<b>37</b>
3.1	Overview	37
3.1.1	Resets	37
3.1.2	Error Reporting and Handling	37
3.1.3	Clocks	39
3.2	Host Interface	39
3.2.1	HyperTransport™ Technology Host Interface	39
3.3	Secondary PCI Bridge	40
3.4	LPC Bridge And Legacy Logic	40
3.4.1	Legacy and Miscellaneous Support Logic	40
3.4.2	Interrupt Controllers	40
3.4.3	Watchdog Timer (WDT)	46
3.4.4	High Precision Event Timer (HPET)	46
3.4.5	Real-Time Clock (Logic Powered by VDD_COREAL)	50

3.5	Enhanced IDE Controller	50
3.6	System Management Bus 2.0 Controller	50
3.6.1	Functional Overview	50
3.6.2	Interrupts	50
3.7	System Management Logic	53
3.7.1	Power Management	53
3.7.2	Serial IRQ Protocol	65
3.7.3	SMBus 1.0	65
3.7.4	Plug And Play IRQs	66
3.7.5	General Purpose I/O	66
3.8	AC '97 Controller	67
3.8.1	Introduction	67
3.8.2	AC '97 Serial Link Interface	69
3.8.3	AC '97 PCI Interface and Bus Master Controller	70
3.8.4	AC '97 Data Buffering	72
3.8.5	AC '97 Power Management Logic	74
3.8.6	AC '97 Link Reset	75
3.9	USB Controller	76
3.9.1	USB Interrupts	76
3.10	LAN Ethernet Controller	77
3.10.1	Interfaces	77
3.10.2	Device Operation	77
3.10.3	Software Interrupt Timer	83
3.10.4	Media Access Control	84
3.10.5	Transmit Operation	88
3.10.6	Receive Operation	93
3.10.7	Statistics Counters	98
3.10.8	VLAN Support	104
3.10.9	Loopback Operation	106
3.10.10	Full-Duplex Operation	108
3.10.11	External PHY Interface	108

3.10.12	Network Port Manager	115
3.10.13	Auto-Negotiation Support	115
3.10.14	Regulating Network Traffic	118
3.10.15	Delayed Interrupts	123
3.10.16	Power Management Support	124
3.10.17	Reset	131
<b>Chapter 4</b>	<b>Registers</b>	<b>135</b>
4.1	Register Overview	135
4.1.1	Configuration Space	135
4.1.2	Register Naming and Description Conventions	136
4.1.3	Positively- and Subtractively-Decoded Spaces	138
4.2	PCI Bridge Configuration Registers (DevA:0xXX)	139
4.3	LPC Bridge Configuration Registers (DevB:0xXX)	149
4.4	Legacy Registers	164
4.4.1	Miscellaneous Fixed I/O Space Registers	164
4.4.2	Legacy DMA Controller (DMAC) Registers	167
4.4.3	Legacy Programmable Interval Timer (PIT) Registers	169
4.4.4	Legacy Programmable Interrupt Controller (PIC)	170
4.4.5	IOAPIC Registers	173
4.4.6	Watchdog Timer Registers	175
4.4.7	High Precision Event Timer Registers	176
4.4.8	Real-Time Clock Registers	183
4.5	Enhanced IDE Controller Registers	184
4.5.1	Enhanced IDE Configuration Registers (DevB:1xXX)	184
4.5.2	EIDE Bus Master I/O Registers	193
4.6	System Management Bus 2.0 Controller Registers	195
4.6.1	System Management Bus Configuration Registers (DevB:2xXX)	195
4.6.2	SMBus Controller Interface Registers	197
4.6.3	Host Controller Interface Registers	200
4.7	System Management Registers	203
4.7.1	System Management Configuration Registers (DevB:3xXX)	203

4.7.2	System Management I/O Mapped Registers (PMxx) .....	220
4.8	AC '97 Controller Registers .....	253
4.8.1	AC '97 Audio Register .....	253
4.8.2	AC '97 Modem Registers .....	264
4.9	USB Controller Registers .....	274
4.9.1	USB Open Host Controller Registers .....	274
4.9.2	USB Enhanced Host Controller Registers .....	279
4.10	LAN Ethernet Controller Registers .....	306
4.10.1	LAN Ethernet Controller User Accessible Registers .....	306
4.10.2	LAN Ethernet Controller Configuration Registers (Dev1:0xXX) .....	308
4.10.3	Memory-Mapped Registers .....	312
4.10.4	Receive Descriptors .....	346
4.10.5	Transmit Descriptors .....	349
<b>Chapter 5</b>	<b>Electrical Data .....</b>	<b>353</b>
5.1	Absolute Ratings .....	353
5.2	Operating Ranges .....	354
5.3	Current Consumption .....	354
5.4	Signal Groups .....	355
5.5	DC Characteristics .....	356
5.6	AC Characteristics .....	358
<b>Chapter 6</b>	<b>Pin Designations .....</b>	<b>361</b>
<b>Chapter 7</b>	<b>Package Specification .....</b>	<b>367</b>
<b>Chapter 8</b>	<b>Test .....</b>	<b>371</b>
8.1	Pin Select Options .....	371
8.1.1	NAND Tree Mode .....	371
8.1.2	High Impedance Mode .....	374
<b>Appendix A</b>	<b>Glossary .....</b>	<b>375</b>
<b>Appendix B</b>	<b>References .....</b>	<b>376</b>
<b>Appendix C</b>	<b>Conventions .....</b>	<b>377</b>



# List of Figures

Figure 1.	System Block Diagram .....	2
Figure 2.	Interrupt Sources .....	41
Figure 3.	Vectored Interrupt Routing .....	42
Figure 4.	High Precision Event Timer Block Diagram .....	47
Figure 5.	High Precision Event Timer Interrupt Routing .....	49
Figure 6.	SMBus Controller Interrupt Model .....	52
Figure 7.	STS to Interrupt Routing .....	55
Figure 8.	Device Monitors and Retrigger Timers .....	56
Figure 9.	System Power State Transitions .....	59
Figure 10.	Transitions from MOFF to SOFF and from SOFF/STD/STR to FON .....	61
Figure 11.	Transitions from FON to SOFF/STD/STR .....	62
Figure 12.	GPIO Pin Format .....	67
Figure 13.	AC '97 Codec Connections .....	68
Figure 14.	Output Frame with Audio Samples in Half Rate Transfer Mode .....	74
Figure 15.	Input Frame with Audio Samples in Half Rate Transfer Mode .....	74
Figure 16.	USB Controller Building Blocks .....	76
Figure 17.	Receive and Transmit Descriptor Rings .....	81
Figure 18.	ISO 8802-3 (IEEE/ANSI 802.3) Data Frame .....	90
Figure 19.	Conduit Mode Transmission with One Collision .....	93
Figure 20.	Conduit Mode Excessive Collisions Error .....	93
Figure 21.	Address Match Logic .....	96
Figure 22.	IEEE 802.3 Frame and Length Field Transmission Order .....	97
Figure 23.	VLAN-Tagged Frame Format .....	105
Figure 24.	Internal and External Loopback Paths .....	107
Figure 25.	Media Independent Interface .....	109
Figure 26.	Frame Format at the MII Interface Connection .....	111
Figure 27.	Auto-Poll and Auto-Poll Data Registers .....	113
Figure 28.	Indirect Access to Auto-Poll Data Registers .....	114
Figure 29.	Pattern Match RAM .....	129
Figure 30.	Configuration Space (Primary and Secondary Bus) .....	136
Figure 31.	FERR_L and IGNNE_L Logic .....	166
Figure 32.	Receive Descriptor Format .....	347
Figure 33.	Format of FLAGS in Receive Descriptor, Offset 08h .....	347
Figure 34.	Transmit Descriptor Format .....	349
Figure 35.	Format of TFLAGS1 in Transmit Descriptor, Offset 00h .....	350
Figure 36.	BGA Designations (Top Side View) .....	361



## List of Tables

Table 1.	I/O Cell Types .....	19
Table 2.	Host HyperTransport™ Technology Pin Descriptions .....	20
Table 3.	Secondary PCI Interface Pin Descriptions .....	21
Table 4.	LPC Bus and Legacy Support Pin Descriptions .....	22
Table 5.	Ultra DMA Enhanced IDE Pin Descriptions .....	23
Table 6.	System Management Pin Descriptions.....	24
Table 7.	USB Pin Descriptions .....	29
Table 8.	AC '97 Pin Descriptions .....	30
Table 9.	MII Interface Pin Descriptions.....	31
Table 10.	Test Pin Descriptions .....	33
Table 11.	Miscellaneous Pin Descriptions .....	34
Table 12.	Error Handling .....	37
Table 13.	IC Clock Pins .....	39
Table 14.	HyperTransport™ Protocol Unit IDs.....	40
Table 15.	Interrupt Routing Configuration .....	41
Table 16.	IOAPIC Redirection Register Connections .....	45
Table 17.	HPET Specifications .....	46
Table 18.	SC04[SCI_EVT] Event Sources .....	51
Table 19.	System Management Events .....	53
Table 20.	System Power States .....	58
Table 21.	Resume Events .....	59
Table 22.	GPIO Output Clock Options .....	67
Table 23.	PCI Interrupt Sources.....	70
Table 24.	Descriptor Format .....	71
Table 25.	PCM Audio Sample Order.....	72
Table 26.	Maximum Valid Frame Size .....	98
Table 27.	Receive Statistics Counters.....	99
Table 28.	Transmit Statistics Counters .....	102
Table 29.	VLAN Tag Control Command.....	106
Table 30.	VLAN Tag Type .....	106
Table 31.	Auto-Negotiation Capabilities .....	115
Table 32.	Sources of Auto-negotiation Advertisement Register (R4) Bits.....	116
Table 33.	MAC Control Pause Frame Format .....	119
Table 34.	FCCMD Bit Functions .....	121
Table 35.	Pause Encoding .....	122
Table 36.	Format of PMR Pointer Words .....	127
Table 37.	Format of PMR Data Words .....	127
Table 38.	ARP Packet Example .....	129
Table 39.	Directed IP Packet Example.....	130
Table 40.	NBT Name Query/Registration Example .....	130
Table 41.	PMR Programming Example .....	130

Table 42.	PCI Configuration Spaces .....	136
Table 43.	PCI Configuration Spaces .....	136
Table 44.	Fixed Address Spaces .....	137
Table 45.	Relocatable Address Spaces.....	138
Table 46.	Register Behavior Types (Read, Write, Etc.).....	138
Table 47.	Registers Affected by DevB:0x41[SHEN] .....	153
Table 48.	OARx Lock Locations .....	162
Table 49.	OARx Lock Bit Descriptions .....	162
Table 50.	DMA Controller Register Summary .....	168
Table 51.	DMA Command Register Bits (Master and Slave DMAC).....	168
Table 52.	DMA Mode Register Bits (Master and Slave DMAC).....	168
Table 53.	PIT Register Summary .....	169
Table 54.	PIC Register Summary.....	171
Table 55.	Watchdog Timer Registers.....	175
Table 56.	HPET Registers.....	177
Table 57.	SMBus Status .....	201
Table 58.	Device Monitor Events .....	233
Table 59.	GPIO X[1:0] Bit Decoding .....	245
Table 60.	GPIO Register Default States.....	246
Table 61.	Audio Mixer Register.....	256
Table 62.	Audio Controller Bus Master Registers .....	258
Table 63.	Modem Mixer Register .....	267
Table 64.	Modem Bus Master Registers .....	268
Table 65.	USB OHCI Memory-Mapped Register Summary .....	277
Table 66.	Receive Descriptor Bit Definitions .....	347
Table 67.	Transmit Descriptor Bit Definitions.....	350
Table 68.	Absolute Ratings .....	353
Table 69.	Operating Ranges .....	354
Table 70.	Current Consumption .....	354
Table 71.	Signal Groups.....	355
Table 72.	DC Characteristics for Signal Group A .....	356
Table 73.	DC Characteristics for Signal Group C.....	357
Table 74.	DC Characteristics for Signal Group E .....	357
Table 75.	DC Characteristics for Signal Group F .....	357
Table 76.	DC Characteristics for Signal Group G .....	357
Table 77.	DC Characteristics for Signal Group H .....	358
Table 78.	AC Characteristics for PCLK.....	359
Table 79.	AC Characteristics for the OSC .....	359
Table 80.	AC Characteristics for USBCLK.....	360
Table 81.	Alphabetical Listing of Signals and Corresponding BGA Designators.....	362
Table 82.	Pin Select Options .....	371
Table 83:	.....	371
Table 84.	NAND Tree 1: Output GNT_L[0] .....	371
Table 85.	NAND Tree 2: Output GNT_L[1] .....	372

Table 86. NAND Tree 3: Output NC24 .....372  
Table 87. NAND Tree 4: Output DCSTOP\_L.....372  
Table 88. NAND Tree 5: Output DIOWP\_L .....373  
Table 89. NAND Tree 6: Output GNT\_L[3] .....373





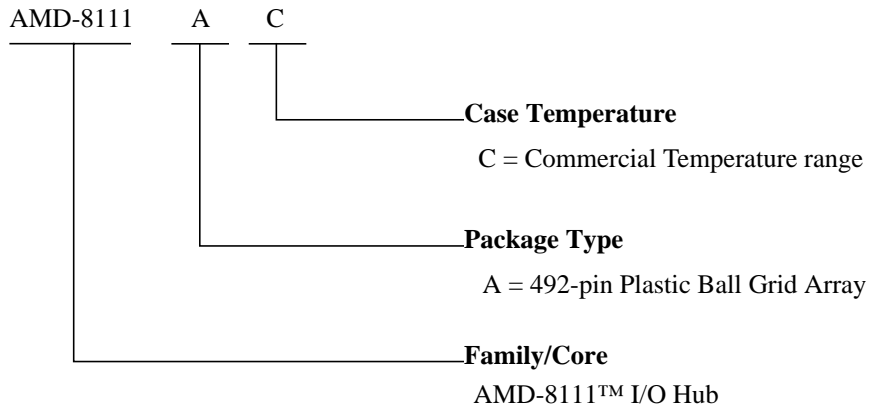




# Chapter 1 Ordering Information

---

The Ordering Part Number (OPN) is formed by a combination of the elements shown below. Contact your AMD representative for detailed ordering information.





## Chapter 2 Signal Descriptions

### 2.1 Terminology

See Section 4.1.2 on page 136 for a description of the register naming conventions used in this document. See Section 3.7.1.6 on page 57 for a description of the system power states: MOFF, SOFF, STD, STR, POS, C3, C2, and FON. See Section 2.12 on page 34 for a description of the power planes. See Section 3.1.1 on page 37 for a description of the types of resets.

Signals with a `_L` suffix are active Low.

Signals described in this chapter utilize the following I/O cell types:

**Table 1. I/O Cell Types**

Name	Notes
I	Input signal only.
O	Output signal only. This includes outputs that are capable of being in the high-impedance state.
OD	Open drain output. These pins are driven Low and expected to be pulled High by external circuitry.
IO	Input or output signal.
IOD	Input or open-drain output.
A	Analog signal.

The following tables provide definitions and reference data about each pin of the IC. In the table:

- The “During Reset” column provides that state of the pin while the pin’s power plane is being reset (while `RESET_L` is Low for the MAIN power plane; while the internal `RST_SOFT` signal is asserted for the AUX power plane).
- The “After Reset” column provides the state of the pin immediately after that reset.
- The “During POS” column provides the state of the pin while in the power on suspend system sleep states.
- The “During S3:S5” column provides the state of the pin while in the suspend to disk, suspend to RAM, or soft off system sleep states.
- The abbreviation “Func.” means that the signal is functional and operating per its defined function.
- The MAIN power plane comprises `VDD_IO` and `VDD_CORE`; the AUX power plane comprises `VDD_IOX` and `VDD_COREX`.
- The pins are only in a defined state when both the specified I/O power plane and the associated core power planes are valid.
  - Pins on the `VDD_IO` power plane are in a defined state when `VDD_IO` and `VDD_CORE` are valid.

- Pins on the VDD\_LDT power plane are in a defined state when VDD\_LDT and VDD\_CORE are valid.
- Pins on the VDD\_IOX power plane are in a defined state when VDD\_IOX and VDD\_COREX are valid.
- Pins on the VDD\_IOAL power plane are in a defined state when VDD\_IOAL and VDD\_COREAL are valid.
- Pins on the VDD\_USB power plane are in a defined state when VDD\_USB, VDD\_USBA, and VDD\_COREX are valid.

## 2.2 Host HyperTransport™ Technology Interface

**Table 2. Host HyperTransport™ Technology Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5												
<p><b>LDTCOMP[3:0]</b>. HyperTransport™ compensation pins. These should be connected as follows:</p> <table border="1"> <thead> <tr> <th>Bit</th> <th>Function</th> <th>External Connection</th> </tr> </thead> <tbody> <tr> <td>[0]</td> <td>Positive RX compensation</td> <td>50 ohms ±10% to VDD_LDT</td> </tr> <tr> <td>[1]</td> <td>Negative RX compensation</td> <td>50 ohms ±10% to VSS</td> </tr> <tr> <td>[3:2]</td> <td>TX compensation</td> <td>100 ohms ±1% between LDTCOMP[3] and LDTCOMP[2]</td> </tr> </tbody> </table>	Bit	Function	External Connection	[0]	Positive RX compensation	50 ohms ±10% to VDD_LDT	[1]	Negative RX compensation	50 ohms ±10% to VSS	[3:2]	TX compensation	100 ohms ±1% between LDTCOMP[3] and LDTCOMP[2]	A	VDD_LDT				
Bit	Function	External Connection																
[0]	Positive RX compensation	50 ohms ±10% to VDD_LDT																
[1]	Negative RX compensation	50 ohms ±10% to VSS																
[3:2]	TX compensation	100 ohms ±1% between LDTCOMP[3] and LDTCOMP[2]																
<b>LDTREQ_L</b> . HyperTransport request to resume pin. External devices assert this pin in order to cause the IC to take the system out of the C3 low-power state. See PM00[BM_STS].	I	VDD_IO																
<b>LDTSTOP_L</b> . HyperTransport disconnect control signal. This is asserted by the IC as specified by DevB:3x[70, 74, 78]. It is always asserted for a minimum of 1 microsecond.	IOD	VDD_IO	3-State	3-State	Low													
<b>LDTRST_L</b> . Reset to other HyperTransport devices. It is identical to RESET_L in timing (but open drain, not push-pull). The logic for this pin includes a debounce circuit. When the signal is asserted High for less than 60 nanoseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 90 nanoseconds to be safely detected by the internal logic.	IOD	VDD_IOX	Low	Func.	Func.	Low												
<b>LRXCAD_H/L[7:0]</b> . HyperTransport receive link command-address-data bus.	LDT input	VDD_LDT																
<b>LRXCLK_H/L</b> . HyperTransport receive link clock.	LDT input	VDD_LDT																

**Table 2. Host HyperTransport™ Technology Pin Descriptions (Continued)**

<b>LRXCTL_H/L.</b> HyperTransport receive link control signal.	LDT input	VDD_LDT				
<b>LTXCAD_H/L[7:0].</b> HyperTransport transmit link command-address-data bus.	LDT output	VDD_LDT	Diff High <sup>1</sup>	Func.	Func.	
<b>LTXCLK_H/L.</b> HyperTransport transmit link clock.	LDT output	VDD_LDT	Func.	Func.	Func.	
<b>LTXCTL_H/L.</b> HyperTransport transmit link control signal.	LDT output	VDD_LDT	Diff Low <sup>1</sup>	Func.	Func.	

**Note:**

1. Diff High and Diff Low for these HyperTransport pins specifies differential High and Low; e.g., Diff High means that the *\_H* signal is High and the *\_L* signal is Low.

## 2.3 Secondary PCI Interface

**Table 3. Secondary PCI Interface Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>AD[31:0].</b> PCI address-data bus. Some of these pins require pull-up and pull-down resistors for strapping options. See DevB:3x48.	IO	VDD_IO	3-State	Parked	Parked	
<b>CBE_L[3:0].</b> PCI command-byte enable bus.	IO	VDD_IO	3-State	Parked	Parked	
<b>DEVSEL_L.</b> PCI device select signal.	IO	VDD_IO	3-State	3-State	3-State	
<b>FRAME_L.</b> PCI frame signal.	IO	VDD_IO	3-State	3-State	3-State	
<b>GNT_L[6:0].</b> PCI master grant signals.	O	VDD_IO	High	High	High	
<b>IRDY_L.</b> PCI master ready signal.	IO	VDD_IO	3-State	3-State	3-State	
<b>PAR.</b> PCI parity signal.	IO	VDD_IO	3-State	Func.	Func.	
<b>PCLK.</b> 33 MHz PCI clock.	I	VDD_IO				
<b>PERR_L.</b> PCI parity error. This signal is only applicable to parity errors on the secondary PCI bus interface. This signal is enabled by DevA:0x3C[PEREN].	IO	VDD_IO	3-State	3-State	3-State	
<b>PGNT_L.</b> Priority PCI master grant. See PREQ_L.	O	VDD_IO	High	High	High	

**Table 3. Secondary PCI Interface Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>PIRQ[A, B, C, D]_L.</b> PCI interrupt requests.	IOD	VDD_ IO	3-State	3-State	Func.	
<b>PREQ_L.</b> Priority PCI master request. PREQ_L/PGNT_L have no functional differences from REQ_L[6:0]/GNT_L[6:0].	I	VDD_ IO				
<b>REQ_L[6:0].</b> PCI master request signals.	I	VDD_ IO				
<b>SERR_L.</b> PCI system error signal. This may be asserted by the system to indicate a system error condition. If enabled by RTC70[7], an NMI interrupt may be generated.	I	VDD_ IO				
<b>STOP_L.</b> PCI target abort signal.	IO	VDD_ IO	3-State	3-State	3-State	
<b>TRDY_L.</b> PCI target ready signal.	IO	VDD_ IO	3-State	3-State	3-State	

## 2.4 LPC Bus and Legacy Support Pins

**Table 4. LPC Bus and Legacy Support Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>LAD[3:0].</b> LPC address-data bus.	IO	VDD_ IO	3-State	3-State	3-State	
<b>LDRQ_L[1:0].</b> LPC DMA request signals.	I	VDD_ IO				
<b>LFRAME_L.</b> LPC frame signal.	O	VDD_ IO	High	High	High	
<b>OSC.</b> 14.31818 MHz clock. This is used for the programmable interval timer and various power management timers.	I	VDD_ IO				
<b>SPKR.</b> Speaker driver from programmable interval timer. This pin is an input only while PWROK is Low; it is used to select the default state of DevB:3x48[12].	IO	VDD_ IO	Input	Low	Last state	

## 2.5 Ultra DMA Enhanced IDE Interface

**Table 5. Ultra DMA Enhanced IDE Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>DADDR[P,S][2:0]</b> . IDE controller [primary, secondary] port address.	O	VDD_IO	Low	Low	Low	
<b>DCS1P_L</b> . IDE controller primary port chip select 1. This is active during accesses to the I/O address space 1F7h–1F0h.	O	VDD_IO	High	High	High	
<b>DCS1S_L</b> . IDE controller secondary port chip select 1. This is active during accesses to the I/O address space 177h–170h.	O	VDD_IO	High	High	High	
<b>DCS3P_L</b> . IDE controller primary port chip select 3. This is active during accesses to the I/O address space 3F7h–3F4h.	O	VDD_IO	High	High	High	
<b>DCS3S_L</b> . IDE controller secondary port chip select 3. This is active during accesses to the I/O address space 377h–374h.	O	VDD_IO	High	High	High	
<b>DDACK[P,S]_L</b> . IDE controller [primary, secondary] port DMA acknowledge signal.	O	VDD_IO	High	High	High	
<b>DDATA[P,S][15:0]</b> . IDE controller [primary, secondary] port data bus.	IO	VDD_IO	3-State	3-State	3-State	
<b>DDRQ[P,S]</b> . IDE controller [primary, secondary] port DMA request signal.	I	VDD_IO				
<b>DIOR[P,S]_L</b> . IDE controller [primary, secondary] port I/O read command.	O	VDD_IO	High	High	High	
<b>DIOW[P,S]_L</b> . IDE controller [primary, secondary] port I/O write command.	O	VDD_IO	High	High	High	
<b>DRDY[P,S]</b> . IDE controller [primary, secondary] port ready strobe.	I	VDD_IO				
<b>DRST[P,S]_L</b> . IDE controller [primary IDE port reset signal. This is asserted when RESET_L is asserted. However, it may be separately asserted through DevB:1x54.	O	VDD_IO	Low	High	Func.	

## 2.6 System Management Pins

This group includes all the GPIO pins, most of which are multiplexed with other functions. The default function of GPIO pins after reset is specified by PM[DF:C0]. When programmed as GPIOs, these pins are capable of being programmed to be inputs or push-pull outputs. GPIO pins that are programmed as outputs, remain functional during sleep states if they are powered.

**Table 6. System Management Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>ACAV.</b> AC available input. This may be used to detect changes to the state of system AC power. It controls PM20[ACAV_STS]. This pin may be configured as GPIO0 by PMC0.	I, IO	VDD_IOX	Input	Input	Func.	Func.
<b>AGPSTOP_L.</b> AGPSTOP_L is used in support of the ACPI C3 and S1 states when an external AGP Graphics device is used. AGPSTOP_L is asserted during S3/S4/S5. See Section 3.7.1.6 on page 57 for AGPSTOP_L sequencing requirements. It is controlled by DevB:3x4F[ASTP_C3EN] and DevB:3x50[ASTP]. This pin may be configured as GPIO1 by PMC1.	O, IO	VDD_IOX	High	High	Func.	Func.
<b>BATLOW_L.</b> Battery low input. This may be used to prevent system resumes from sleep states if the battery power is low and AC power is not available. This pin may be configured as GPIO2 by PMC2.	I, IO	VDD_IOX	Input	Input	Func.	Func.
<b>C32KHZ.</b> 32.768 kHz clock output. This signal is active in all states except MOFF. This pin may be configured as GPIO3 by PMC3.	O, IO	VDD_IOX	Func.	Func.	Func.	Func.
<b>CLKRUN_L.</b> Clock run input and open drain output. This signal is defined by the PCI Mobile Design Guide. It may be used to control the activity of PCI clocks. It is available in all power states except STR, STD, SOFF, and MOFF. This pin may be configured as GPIO5 by PMC5. See PM[DF:C0] for the power up defaults of all GPIO pins. See Section 3.7.1.5 on page 56 for more details.	IOD, IO	VDD_IO	See left	See left	Func.	
<b>CPUSLEEP_L.</b> Processor non-snoop sleep mode open drain output. This may be connected to the sleep pin of the processor to place it into a non-snoop-capable low-power state. It is controlled by DevB:3x4F[CSLP_C3EN] and DevB:3x50[CSLP]. This pin may be configured as GPIO6 by PMC6. See PM[DF:C0] for the power up defaults of all GPIO pins.	OD, IO	VDD_IO	See left	See left	Func.	
<b>CPUSTOP_L.</b> Processor clock stop output. This may be connected to the system clock chip to control the host clock signals. It is controlled by DevB:3x50[CSTP]. This pin may be configured as GPIO7 by PMC7.	O, IO	VDD_IO	High	High	Func.	
<b>DCSTOP_L.</b> DRAM controller stop. This may be connected to the system memory controller to indicate that its clock is going to stop. It is controlled by DevB:3x50[DCSTP]. Also see Section 3.7.1.6 on page 57 for a description of DCSTOP_L during ACPI state transitions.	O	VDD_IOX	Func.	Func.	Func.	Func.
<b>EXTSMI_L.</b> External SMI. This pin may be used to generate SMI or SCI interrupts and resume events. It controls PM20[EXTSMI_STS].	I	VDD_IOX				



**Table 6. System Management Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>FANCON[1:0]</b> . Fan control outputs. These may be used to control system fans. The frequency and duty cycle are specified by PMF8. FANCON1 may be configured as GPIO9 by PMC9.	O, IO	VDD_ IO	Low	Func.	Func.	
<b>FANRPM</b> . Fan revolutions per minute input. This may come from the system power supply. It is expected that it represents 2 pulses per fan revolution. PM1F, which is a counter that is clocked by this signal, can be used to determine the speed of the fan. This pin may be configured as GPIO10 by PMCA.	I, IO	VDD_ IO				
<b>GPIO14</b> . General purpose I/O 14. This is controlled by PMCE. See PM[DF:C0] for the power up defaults of all GPIO pins.	IO	VDD_ IOX	See left	See left	Func.	Func.
<b>GPIO[31:26, 17, 16, 8, 4]</b> . General purpose I/O. These are controlled by PM[DF:C0]. See PM[DF:C0] for the power up defaults of all GPIO pins. <b>NMI#</b> . See PM[DF:C0] for the power up defaults and functional description.	IO	VDD_ IO	See left	See left	Func.	
<b>INTIRQ8_L</b> . Real-time clock interrupt output. This is real-time clock interrupt. This pin may be configured as GPIO11 by PMCB.	O, IO	VDD_ IO	High	High	Func.	
<b>INTRUDER_L</b> . Intruder detection. This controls PM46[INTRDR_STS].	I	VDD_ IOAL				
<b>IRQ1</b> . Keyboard interrupt request input. This pin may be configured as GPIO12 by PMCC.	I, IO	VDD_ IO				
<b>IRQ6</b> . Floppy disk controller interrupt request input. This pin may be configured as GPIO13 by PMCD.	I, IO	VDD_ IO				
<b>IRQ12</b> . Mouse interrupt request input. This pin may be configured as GPIO15 by PMCF.	I, IO	VDD_ IO				
<b>IRQ14</b> . IDE primary port interrupt request.	I	VDD_ IO				
<b>IRQ15</b> . IDE secondary port interrupt request.	I	VDD_ IO				
<b>KA20G</b> . Keyboard A20 gate. This is expected to be the gate A20 signal from the system keyboard controller. It affects A20M_L.	I	VDD_ IO				
<b>KBRC_L</b> . Keyboard reset command. This is expected to be the processor reset signal from the system keyboard controller. When asserted, it generates an INIT interrupt to the processor.	I	VDD_ IO				

**Table 6. System Management Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>LID.</b> Lid change-state detect input. This may be used to detect state changes in notebook shell lids. The logic for this pin includes a debounce circuit. When the signal is asserted High or Low for less than 12 milliseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 16 milliseconds to be safely detected by the internal logic. This pin may be configured as GPIO18 by PMD2.	I, IO	VDD_ IOX				
<b>PCISTOP_L.</b> PCI bus clock stop. This may be used to control the system clock chip to control the PCI bus clock signals. It is controlled by DevB:3x50[PSTP]. It may also be used in association with CLKRUN_L; see Section 3.7.1.5 on page 56 for more details.	O	VDD_ IO	High	High	Func.	
<b>PME_L.</b> Power management interrupt. This pin may be used to generate SMI or SCI interrupts and resume events. It controls PM20[PME_STS].	I	VDD_ IOX				
<b>PNPIRQ[2:0].</b> Plug and play interrupt request inputs. These may be assigned to control any of 12 of the internal IRQ signals by DevB:3x44. PNPIRQ0 may be configured as GPIO19 by PMD3; PNPIRQ1 may be configured as GPIO20 by PMD4; PNPIRQ2 may be configured as GPIO21 by PMD5.	I, IO,	VDD_ IO	Input	Input	Input	
<b>PRDY.</b> Processor ready. When this is asserted, the IC freezes the timers specified by DevB:3x4C.	I	VDD_ IO				
<b>PWRBTN_L.</b> Power button. This may be used to control the automatic transition from a sleep state to FON. It controls PM00[PWRBTN_STS]. Also, if it is asserted for four seconds from any state other than SOFF, then a power button override event is generated. A power button override event causes the PWRON_L pin to be driven High and PM00[PBOR_STS] to be set High. The logic for this pin includes a debounce circuit. When the signal is asserted High or Low for less than 12 milliseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 16 milliseconds to be safely detected by the internal logic.	I	VDD_ IOX				

**Table 6. System Management Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>PWROK.</b> Power OK. This is required to be Low while the main power planes are not valid, stay Low for at least 50 milliseconds after they become valid, and then go High. It is the reset source for the main power supplies of the IC, VDD_CORE and VDD_IO. The logic for this pin includes a debounce circuit. When the signal is asserted High for less than 30 microseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 60 microseconds to be safely detected by the internal logic.	I	VDD_IOX				
<b>PWRON_L.</b> Main power on. This is expected to control the main power supplies to the system board. It is asserted during the FON, C2, C3, and POS states; it is deasserted during the STR, STD and SOFF states. When power is applied to the AUX plane, this signal is forced inactive until RST_SOFT is deasserted. See Section 3.7.1.6.2 on page 60 for more details.	OD	VDD_IOX	3-State	Low	Low	3-State
<b>RESET_L.</b> System reset. This is the system reset signal for logic that is powered by the main power supplies. See Section 3.1.1 on page 37 and Section 3.7.1.6.2 on page 60 for more details.	O	VDD_IOX	Func.	Func.	High	Low
<b>RI_L.</b> Ring indicate. This is intended to cause the system to resume to the FON states and generate SCI or SMI interrupts. It controls PM20[RI_STS].	I	VDD_IOX				
<b>RPWRON.</b> RAM power on. This is intended to control power to the system memory power plane. When High, it is expected that power to system memory is enabled. When Low, it is expected that power to system memory is disabled. This pin is Low during STD and SOFF and High in all other states. See Section 3.7.1.6.2 on page 60 for more details.	OD	VDD_IOX	Func.	3-State	3-State	Func.
<b>RTCX_IN.</b> Real-time clock 32.768 kHz crystal input. This pin is expected to be connected through a crystal oscillator to RTCX_OUT.	A	VDD_IOAL	Func.	Func.	Func.	Func.
<b>RTCX_OUT.</b> Real-time clock 32.768 kHz crystal output.	A	VDD_IOAL	Func.	Func.	Func.	Func.
<b>S3PLL_LF.</b> S3PLL external loop filter pin.	A	VDD_IOX	Func.	Func.	Func.	Func.
<b>S3PLL_LF_VSS.</b> S3PLL external loop filter pin.	A	VDD_IOX	Func.	Func.	Func.	Func.
<b>SERIRQ.</b> Serial IRQ signal. This pin supports the serial IRQ protocol. Control for this is in DevB:3x4A.	IO	VDD_IO	3-State	3-State	Func.	

**Table 6. System Management Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>SLPBTN_L.</b> Sleep button input. This may be used to control the automatic transition from a sleep state to FON. It controls PM00[SLPBTN_STS]. Also, if it is asserted for four seconds from any state other than SOFF, then a power button override event is generated. A power button override event causes the PWRON_L pin to be driven High and PM00[PBOR_STS] to be set High. The logic for this pin includes a debounce circuit. When the signal is asserted High or Low for less than 12 milliseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 16 milliseconds to be safely detected by the internal logic. This pin may be configured as GPIO23 by PMD7.	I, IO	VDD_ IOX				
<b>SMBALERT0_L.</b> SMBus 1.0 alert input. This may be used to generate an SMI or SCI interrupt or a resume event associated with the SMBus logic. This pin may be configured as GPIO22 by PMD6. Note: although this pin resides on the VDD_IOX power plane, it can only be used for the SMBus alert function while the main power supply is valid.	I, IO	VDD_ IOX				
<b>SMBALERT1_L.</b> SMBus 2.0 alert input. This may be used to generate an SMI or SCI interrupt or a resume event associated with the SMBus logic. This pin may be configured as GPIO24 by PMD8.	I, IO	VDD_ IOX				
<b>SMBUSC[1:0].</b> System management bus (SMBus) clock. SMBUSC[0] is associated with the SMBus 1.0 host controller and SMBUSC[1] is associated with the SMBus 2.0 host controller. The logic for the SMBUSC1 pin includes a debounce circuit. When the signal is asserted High or Low for less than 80 nanoseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 100 nanoseconds to be safely detected by the internal logic.	IOD	VDD_ IOX	3-State	3-State	Func.	Func.
<b>SMBUSD[1:0].</b> System management bus (SMBus) data. SMBUSD[0] is associated with the SMBus 1.0 host controller and SMBUSD[1] is associated with the SMBus 2.0 host controller. The logic for the SMBUSD1 pin includes a debounce circuit. When the signal is asserted High or Low for less than 80 nanoseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 100 nanoseconds to be safely detected by the internal logic.	IOD	VDD_ IOX	3-State	3-State	Func.	Func.

**Table 6. System Management Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>SUSPEND_L.</b> This may be used to gate RESET_L during Suspend to RAM (S3) or to enable power reduction during Power on Suspend (S1). It is controlled by DevB:3x50[SUSP]. This pin may be configured as GPIO25 by PMD9.	O, IO	VDD_IOX	High	High	Func.	
<b>THERM_L.</b> Thermal warning detect. This may be used to automatically enable processor throttling as specified by DevB:3x4D[TTH_RATIO, TTH_EN, TTH_LOCK]. This pin is also controlled by DevB:3x40[TH2SD, TTPER, THMINEN] and DevB:3x70[TTLS, TTSMF], see also PM20[THERM_STS], PM22[THERM_EN], PM2A[THMSMI], PMF8[FC[1:0]THERM]. See Section 3.7.1.4 on page 56 for more details.	I	VDD_IO				
<b>THERMTRIP_L.</b> Processor Thermal Trip Point exceeded. When asserted while PWRON_L =0, PWROK=1, and RESET_L= 1 the IC sets PM46[TT_STS] and forces the system to S5. The logic for this pin includes a debounce circuit. When the signal is asserted Low for less than 960 nanoseconds, the debounce logic does not propagate a change of the signal value to the internal logic; the signal must be asserted for at least 990 nanoseconds to be safely detected by the internal logic.	I	VDD_IO				

## 2.7 Universal Serial Bus Interface

**Table 7. USB Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>USBCLK.</b> 48 MHz USB clock.	I	VDD_IO				
<b>USB0_H/L[2:0].</b> USB hub 0 ports. These are the three pairs of differential USB signals. USB0P[2:0] are the positive signals and USB0N[2:0] are the negative signals.	IO	VDD_USB	Low	Low	3-State; can detect resume	3-State; can detect resume
<b>USB1_H/L[2:0].</b> USB hub 1 ports. These are the three pairs of differential USB signals. USB1P[2:0] are the positive signals and USB1N[2:0] are the negative signals.	IO	VDD_USB	Low	Low	3-State; can detect resume	3-State; can detect resume

**Table 7. USB Pin Descriptions (Continued)**

<b>USBOC0_L.</b> USB over current detect 0. This goes to the USB logic to report the occurrence of an over-current condition on the voltage supplied to the USB ports.	I	VDD_ IO				
<b>USBOC1_L.</b> USB over current detect 1. When enabled to do so, this can be routed to the USB block to be a second source of USB port over-current detection.	I	VDD_ IO				
<b>USB_REXT.</b> USB PHY external resistor. This signal connects to an external resistor. The resistor should be 3.09 Kohm with a tolerance of 1% across the 0 °C to +125 °C temperature range.	A	VDD_ USB				

## 2.8 AC '97 Interface

**Table 8. AC '97 Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During POS	During S3:S5
<b>ACCLK.</b> AC '97 bit clock. 12.288 MHz fixed frequency from primary codec. An external 10-Kohm pull-down resistor is required on this pin in support of powering down the AC-link codec.	I	VDD_ IO				
<b>ACRST_L.</b> AC '97 asynchronous reset.	O	VDD_ IOX	Low	Low	Low	Low
<b>ACSDI0.</b> AC '97 primary codec serial data in (slot 0 codec ID field = 00b) from first codec in system. An external 10-Kohm pull-down resistor is required on this pin in support of powering down the AC-link codec.	I	VDD_ IOX			Can detect resume	Can detect resume
<b>ACSDI1.</b> AC '97 secondary codec serial data in (slot 0 codec ID field = 01b) from second codec in system. An external 10-Kohm pull-down resistor is required on this pin in support of powering down the AC-link codec.	I	VDD_ IOX			Can detect resume	Can detect resume
<b>ACSDO.</b> AC '97 serial data out.	O	VDD_ IO	Low	Low	Low	
<b>ACSYNC.</b> AC '97 frame synchronization pulse.	O	VDD_ IO	Low	Low	Low	

## 2.9 MII Interface

**Table 9. MII Interface Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During S1:S2	During S3:S5
<b>MII_TX_CLK.</b> MII Transmit Clock. MII_TX_CLK is a continuous clock input that provides the timing reference for the transfer of the MII_TX_EN and MII_TXD[3:0] signals. MII_TX_CLK must provide a nibble rate clock (25% of the network data rate). Hence, an MII transceiver operating at 10 Mbit/s must provide a MII_TX_CLK frequency of 2.5 MHz and an MII transceiver operating at 100 Mbit/s must provide a MII_TX_CLK frequency of 25 MHz.	I	VDD_IOX				
<b>MII_TXD[3:0].</b> MII Transmit Data. MII_TXD[3:0] is the nibble-wide MII transmit data bus. Valid data is generated on MII_TXD[3:0] on every MII_TX_CLK rising edge while MII_TX_EN is asserted. While MII_TX_EN is deasserted, MII_TXD[3:0] values are driven Low. MII_TXD[3:0] transitions synchronous to MII_TX_CLK rising edges.	O	VDD_IOX	Low	Low	Func.	Func.
<b>MII_TX_EN.</b> MII Transmit Enable. MII_TX_EN indicates when valid transmit nibbles are presented on the MII. While MII_TX_EN is asserted, MII_TXD[3:0] data are generated on MII_TX_CLK rising edges. MII_TX_EN is asserted with the first nibble of preamble and remains asserted throughout the duration of a packet until it is deasserted prior to the first MII_TX_CLK following the final nibble of the frame. MII_TX_EN transitions synchronous to MII_TX_CLK rising edges.	O	VDD_IOX	Low	Low	Func.	Func.
<b>MII_COL.</b> MII Collision. MII_COL is an input that indicates that a collision has been detected on the network medium.	I	VDD_IOX				
<b>MII_CRS.</b> MII Carrier Sense. MII_CRS is an input that indicates that a non-idle medium, due either to transmit or receive activity, has been detected.	I	VDD_IOX				
<b>MII_RX_CLK.</b> MII Receive Clock. MII_RX_CLK is a clock input that provides the timing reference for the transfer of the MII_RX_DV, MII_RXD[3:0], and MII_RX_ER signals. MII_RX_CLK must provide a nibble rate clock (25% of the network data rate). Hence, an MII transceiver operating at 10 Mbit/s must provide an MII_RX_CLK frequency of 2.5 MHz and an MII transceiver operating at 100 Mbit/s must provide an MII_RX_CLK frequency of 25 MHz. When the external PHY switches the MII_RX_CLK and MII_TX_CLK, it must provide glitch-free clock pulses.	I	VDD_IOX				

**Table 9. MII Interface Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During S1:S2	During S3:S5
<b>MII_RXD[3:0].</b> MII Receive Data. MII_RXD[3:0] is the nibble-wide MII receive data bus. Data on MII_RXD[3:0] is sampled on every rising edge of MII_RX_CLK while MII_RX_DV is asserted. MII_RXD[3:0] is ignored while MII_RX_DV is de-asserted.	I	VDD_IOX				
<b>MII_RX_DV.</b> MII Receive Data Valid. MII_RX_DV is an input used to indicate that valid received data is being presented on the MII_RXD[3:0] pins and MII_RX_CLK is synchronous to the receive data. In order for a frame to be fully received on the MII, MII_RX_DV must be asserted prior to the MII_RX_CLK rising edge, when the first nibble of the Start of Frame Delimiter is driven on MII_RXD[3:0], and must remain asserted until after the rising edge of MII_RX_CLK, when the last nibble of the MII_CRC is driven on MII_RXD[3:0]. MII_RX_DV must then be deasserted prior to the MII_RX_CLK rising edge which follows this final nibble. MII_RX_DV transitions are synchronous to MII_RX_CLK rising edges.	I	VDD_IOX				
<b>MII_RX_ER.</b> MII Receive Error. MII_RX_ER is an input that indicates that the MII transceiver device has detected a coding error in the receive frame currently being transferred on the MII_RXD[3:0] pins. When MII_RX_ER is asserted while MII_RX_DV is asserted, a MII_CRC error is indicated in the receive descriptor for the incoming receive frame. MII_RX_ER is ignored while MII_RX_DV is deasserted. Special code groups generated on MII_RXD while MII_RX_DV is deasserted are ignored (e.g., Bad SSD in TX and IDLE in T4). MII_RX_ER transitions are synchronous to MII_RX_CLK rising edges.	I	VDD_IOX				



**Table 9. MII Interface Pin Descriptions (Continued)**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During S1:S2	During S3:S5
<b>MII_PHY_RST.</b> MII PHY Reset. MII_PHY_RST is an output pin that is used to reset the external PHY. The output polarity is determined by ENC054[PHY_RST_POL].	O	VDD_IOX	Low	Low	Func.	Func.
<b>MII_MDC.</b> MII Management Data Clock. MII_MDC is a non-continuous clock output that provides a timing reference for bits on the MII_MDIO pin. During MII management port operations, MII_MDC runs at a nominal frequency of 2.5 MHz. When no management operations are in progress, MII_MDC is driven Low. If the MII Management port is not used, the MII_MDC pin can be left floating.	O	VDD_IOX	Low	Low	Func.	Func.
<b>MII_MDIO.</b> MII Management Data I/O. MII_MDIO is the bidirectional MII management port data pin. MII_MDIO is an output during the header portion of the management frame transfers and during the data portions of write transfers. MII_MDIO is an input during the data portions of read data transfers. When an operation is not in progress on the management port, MII_MDIO is not driven. MII_MDIO transitions are synchronous to MII_MDC falling edges. If the PHY is attached through an MII physical connector, then the MII_MDIO pin should be externally pulled Low with a 10-Kohm 5% resistor. If the PHY is permanently connected, then the MII_MDIO pin should be externally pulled High with a 10-Kohm 5% resistor.	IO	VDD_IOX			Func.	Func.

## 2.10 Test

**Table 10. Test Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During S1:S2	During S3:S5
<b>TEST_L.</b> Scan, NAND tree, and high-impedance mode enable.	I	VDD_IO				

## 2.11 Miscellaneous

**Table 11. Miscellaneous Pin Descriptions**

Pin Name and Description	I/O Cell Type	Power Plane	During Reset	After Reset	During S1:S2	During S3:S5
<b>NC[32:0]</b> . Must be left unconnected.						
<b>STRAPH[3:0]</b> . Must be tied High.						
<b>STRAPL[3:0]</b> . Must be tied Low.						

## 2.12 Power and Ground

See Section 3.7.1.6 on page 57 for a description of the system power states. The following power and ground planes are connected to the IC through BGA pins.

**VDD\_IO**. 3.3-V supply. This plane is required to be valid in the FON and POS power states. It is part of the MAIN power supply.

**VDD\_CORE**. 1.8-V supply. This plane is required to be valid in the FON and POS power states. It is part of the MAIN power supply.

**VDD\_IOX** and **VDD\_COREX**. **VDD\_IOX** is a 3.3-V plane and **VDD\_COREX** is a 1.8-V plane. Both of these planes are required to be valid at the same times. They are valid in all system power states except MOFF. Both power planes are part of the AUX power supply. Register bits that are on the **VDD\_COREX** plane are reset by the internal **RST\_SOFT** pulse that is generated for about 30 milliseconds after **VDD\_COREX** becomes valid.

**VDD\_LDT**. 1.2-V HyperTransport™ link reference voltage. This plane is required to be valid in the FON and POS power states. It is part of the MAIN power supply.

**VDD\_REF**. 5.0-V reference supply. This plane is required to be valid in all power states except MOFF. It is part of the AUX power supply.

**VDD\_RTC**. 3.3-V supply. This plane is required to be valid in all power states. It is typically powered by a battery. It supplies power for the internal AL power plane when AUX is not valid.

**VDD\_USB**. 3.3-V supply filtered for the USB transceivers. This plane is required to be valid in all power states except MOFF.

**VDD\_USBA**. 1.8-V supply filtered for the USB transceivers. This plane is required to be valid in all power states except MOFF.

**VSS**. Main ground plane.

**VSS\_USBA**. Ground plane filtered for the USB transceivers.

The IC also includes the following two internal power planes.

**VDD\_IOAL** and **VDD\_COREAL**. VDD\_IOAL is a 3.3-V plane and VDD\_COREAL is a 1.8-V plane. Both power planes are part of the internal AL (always) power plane. AL is supplied by AUX when that plane is valid or by VDD\_RTC when AUX is not valid. AL powers the real-time clock and some system management circuitry.

General requirements:

- The voltage level of VDD\_CORE is required to be less than VDD\_IO at all times.
- The voltage level of VDD\_COREX is required to be less than VDD\_IOX at all times.
- The voltage level of VDD\_IO and VDD\_IOX is required to be less than VDD\_REF at all times.
- VDD\_COREX, VDD\_IOX and VDD\_REF are required to be powered before VDD\_CORE and VDD\_IO may be powered.



## Chapter 3 Functional Operation

### 3.1 Overview

#### 3.1.1 Resets

The IC generates an internal reset for the AUX power planes called RST\_SOFT. RST\_SOFT lasts for about 30 milliseconds after the AUX planes become valid.

PWROK is the source of reset for the VDD\_IO and VDD\_CORE logic on the IC. From this signal, RESET\_L, INIT\_L and LDTRST\_L are derived.

It is possible to generate a reset to the system through DevB:0x47[SWRST]. These cause RESET\_L and LDTRST\_L to be asserted for greater than 1.5 milliseconds.

Various system resets are also possible through PORTCF9.

It is also possible to generate a reset to the processor (without clearing the cache) with INIT interrupt through the external keyboard controller using KBRC\_L, the PORT92 register, or from a shutdown command from the host.

#### 3.1.2 Error Reporting and Handling

The following is a summary of the errors that are reported and how they are handled in the IC; *secondary* refers to the secondary side of the PCI bridge, which includes an external PCI bus, the two USB controllers and the Ethernet controller.

**Table 12. Error Handling**

Error Type	Status Bit	Handler Enable	Handler Action	Notes
Received target abort on host	DevA:0x04[RTA]	DevB:0x40[NMIONERR]	Generate NMI	
Received master abort on host	DevA:0x04[RMA]	DevB:0x40[NMIONERR]	Generate NMI	
Host bus CRC error	DevA:0xC4[LKFAIL]	DevA:0xC4[CRCFEN]	Flood outgoing HyperTransport™ link with sync packets	1
	DevA:0xC4[CRCEERR]	DevB:0x40[NMIONERR]	Generate NMI	3
Link incoming sync flood is detected	none	DevB:0x47[RSTONLE]	System Reset	1
Signaled target abort on secondary bus	DevA:0x1C[STA]	DevB:0x40[NMIONERR]	Generate NMI	
Received target abort on secondary bus	DevA:0x1C[RTA]	DevB:0x40[NMIONERR]	Generate NMI	3

**Table 12. Error Handling (Continued)**

Error Type	Status Bit	Handler Enable	Handler Action	Notes
Received target abort on secondary bus while executing a downstream posted command	DevA:0x1C[RTA]	DevB:0x40[SERREN]	Flood outgoing HyperTransport link with sync packets	1
Received master abort on secondary bus	DevA:0x1C[RMA]	DevB:0x40[NMIONERR]	Generate NMI	3
Received master abort on secondary bus while executing a downstream posted command	DevA:0x1C[RMA]	DevB:0x40[SERREN] DevA:0x3C[MARSP]	Flood outgoing HyperTransport link with sync packets	1
SERR_L assertion detected	DevA:0x1C[RSE]	DevB:0x40[NMIONERR]	Generate NMI	3
	PORT61[SERR]	PORT61[CLRSERR]	Generate NMI	3
	DevA:0x1C[RSE]	DevA:0x04[SERREN] DevA:0x3C[SERREN]	Flood outgoing HyperTransport link with sync packets	1
Detected data parity error on secondary bus while receiving data as master or target	DEV:0x1C[DPE]	DevB:0x40[NMIONERR]	Generate NMI; assert PERR_L if target	2
Detected data parity error on secondary bus while receiving data as master	DEV:0x1C[MDPE]	DevA:0x3C[PEREN]	Generate NMI	
Detected address parity error during cycle from external master	DevA:0x1C[DPE]	DevA:0x04[SERREN] DevA:0x3C[SERREN] DevA:0x3C[PEREN]	Flood outgoing HyperTransport link with sync packets	1
PERR_L assertion detected while sending data as master	DevA:0x1C[MDPE]	DevA:0x3C[PEREN]	Generate NMI	3
PERR_L assertion detected while sending posted data as master	DevA:0x1C[MDPE]	DevA:0x3C[PEREN] DevA:0x04[SERREN]	Flood outgoing HyperTransport link with sync packets	1
Secondary discard timer expires	DevA:0x3C[DTSTAT]	DevB:0x40[NMIONERR]	Generate NMI	3
	DevA:0x3C[DTSTAT]	DevA:0x3C[DTSERREN] DevA:0x04[SERREN]	Flood outgoing HyperTransport link with sync packets	1

**Table 12. Error Handling (Continued)**

Error Type	Status Bit	Handler Enable	Handler Action	Notes
LPC protocol error	DevB:0x40[LPCERR], PORT61[IOCHK]	PORT61[CLRIOCHK]	Generate NMI	
Received posted write targeting LPC while LPC bus master is active	DevB:0x40[PW2LPC]	DevB:0x40[NMIONERR]	Generate NMI	
<i>Notes:</i>				
1. The link is flooded with sync packets and DevA:0x04 gets set. System reset is possible when setting DevB:0x47[RSTONLE].				
2. PERR_L is only asserted if it is enabled by DevA:0x3C[PEREN].				
3. If both sync flood and NMI are enabled then the NMI will not reach the host because the link gets flooded before.				

### 3.1.3 Clocks

The IC includes the following clocks on its pins.

**Table 13. IC Clock Pins**

Name	Pin Type	Description
OSC	Input	14.31818 MHz. This is used by the legacy programmable interval timer (PIT) and by some of the power management logic.
PCLK	Input	Up-to 33.333 MHz PCI clock.
USBCLK	Input	48 MHz. This is used by the USB controller.
RTC_XIN, RTC_XOUT	Oscillator	32.768 kHz oscillator pair. The clock generated by this oscillator is used by the real time clock and by some of the system management logic; it is powered by VDD_IOAL.
ACCLK	Input	12.288 MHz. This is used as serial input clock for AC '97.
MII_RXCLK	Input	2.5 or 25 MHz. This is the receive clock from Ethernet Phy0
MII_TXCLK	Input	2.5 or 25 MHz. This is the transmit clock from Ethernet Phy0

## 3.2 Host Interface

### 3.2.1 HyperTransport™ Technology Host Interface

The HyperTransport™ technology interface provides the host connection for the IC. The link supports up to 800 megabytes per second of total bandwidth.

The HyperTransport input and output links support multiple bit widths, including 8, 4, and 2 bits, with a 200 MHz HyperTransport clock.

#### 3.2.1.1 HyperTransport™ Protocol Unit IDs

The HyperTransport protocol Unit ID values are assigned by the platform designer by programming DevA:0xC0[BUID].

**Table 14. HyperTransport™ Protocol Unit IDs**

Unit	Unit ID	Data Associated with Unit ID
First	BUID	PCI (all functions except IDE)
Second	BUID + 1	IDE primary port
Third	BUID + 2	IDE secondary port
Fourth	BUID + 3	EHCI-based USB Controller

### 3.3 Secondary PCI Bridge

The secondary PCI bridge interfaces to the internal USB and Ethernet controllers and the external PCI bus. Peer to peer transactions are supported between devices on the external PCI bus. However, all other transactions pass through the host interface; peer to peer transactions between internal and external devices are not supported.

### 3.4 LPC Bridge And Legacy Logic

The LPC bridge supports external peripherals and BIOS devices through the LPC bus. It also includes legacy devices and logic required for compatibility with existing software.

#### 3.4.1 Legacy and Miscellaneous Support Logic

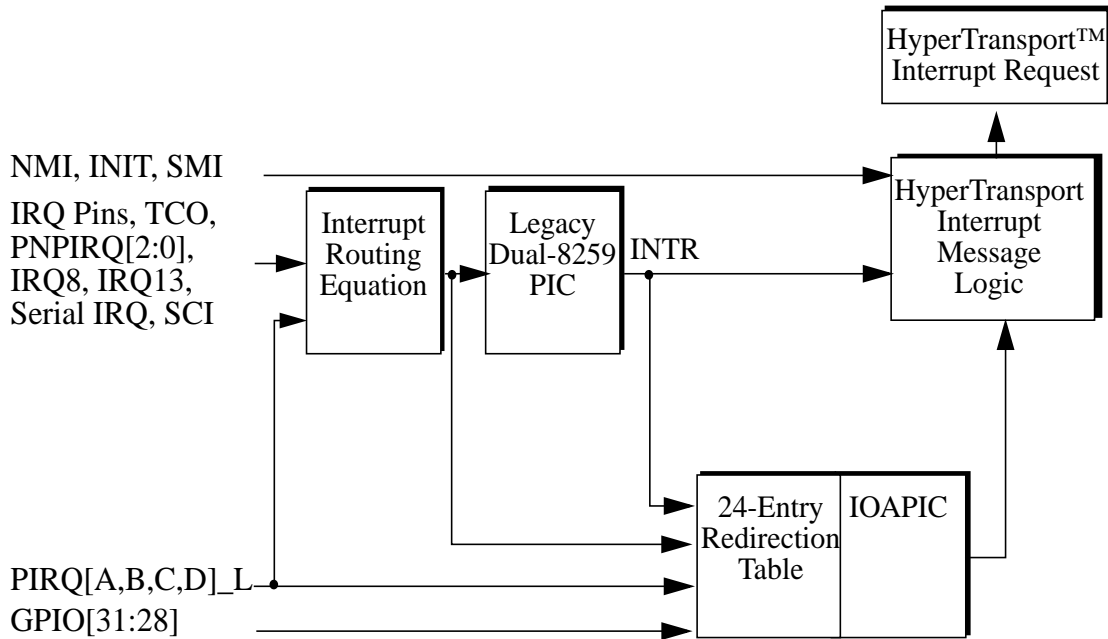
The IC includes the following legacy support logic:

- PORT61 and PORT92 legacy registers.
- FERR\_L and IGNNE interrupt logic.
- PORT4D0 legacy interrupt edge-level select logic.
- PORTCF9 reset logic.
- Legacy dual-8237 DMA controller.
- Legacy 8254 PIT.

#### 3.4.2 Interrupt Controllers

Interrupt types include vectored interrupts—INTR, ExtINT, fixed, and lowest priority—and non-vectored interrupts NMI, SMI, and INIT. Based on DevB:0x4B[APICEN], these are sent through routing equations, the legacy PIC, and the IOAPIC to be transmitted to the host through HyperTransport messages. As HyperTransport messages, they can be sourced from either the internal IOAPIC or other internal HyperTransport interrupt message logic (see Figure 2).





**Figure 2. Interrupt Sources**

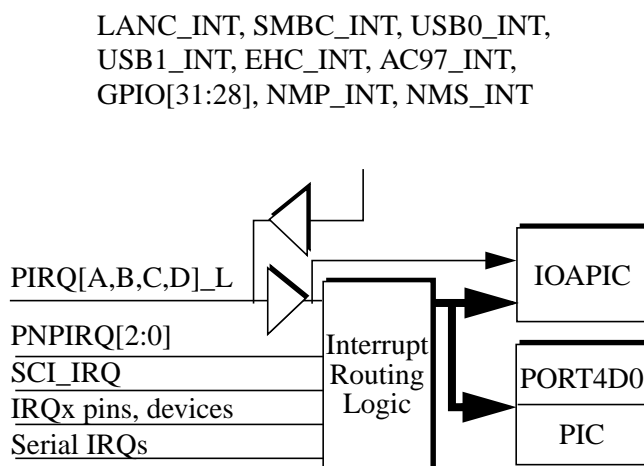
The following table specifies how interrupts are routed based on the configuration bit.

**Table 15. Interrupt Routing Configuration**

APIC-EN	Interrupt Delivery Mechanisms.
0	HyperTransport interrupt message logic is used to send all interrupt requests.
1	Interrupts from the IOAPIC and internally generated non-vectorred interrupts are translated into HyperTransport interrupt request messages.

### 3.4.2.1 Interrupt Control and Routing

Vectorred interrupt requests are routed to the legacy PIC and APIC as shown in Figure 3. The interrupt signals to the PIC can be either rising-edge triggered or active-Low level triggered. It is expected that edge-triggerred interrupts such as IRQ14 from the IDE controller rise into the PIC to indicate the presence of an interrupt. Conversely, level-sensitive interrupts are Low into the PIC to indicate the presence of an interrupt. Edge and level sensitivity for each IRQ are programmed into the PIC through PORT4D0.



**Figure 3. Vectored Interrupt Routing**

Several internal interrupts are shared with the PCI interrupts pins. These internal interrupt signals drive the PIRQ[A,B,C,D]\_L pins Low as outputs; when the internal interrupts are deasserted, the pins are left in the high impedance state. These pins are wire-ORed into the active state with external interrupts. The result enters the IC and goes to the interrupt routing logic. The internal interrupts are mapped to the PIRQ[A,B,C,D]\_L pins as follows:

- GPIO[31:28] can be specified by DevB:0x4B[MPIRQ] to drive onto PIRQ[A,B,C,D]\_L, respectively.
- The Ethernet controller interrupt—LANC\_INT—drives onto the PIRQA\_L pin.
- The primary and secondary port, when these ports are in native mode, IDE controller interrupts—NMP\_INT, NMS\_INT—drive onto the PIRQA\_L pin.
- The AC '97 interrupt—AC97\_INT—drives onto the PIRQB\_L pin.
- The USB controller interrupts—USB0\_INT, USB1\_INT, EHC\_INT—drives onto the PIRQD\_L pin.
- The SMBus 2.0 controller interrupt—SMBC\_INT—drives onto the PIRQD\_L pin.

Alternatively, these interrupts may all be mapped to PIRQD\_L, as specified by DevB:0x47[ALLTOD].

Here are the interrupt routing logic equations:

```
PIRQ_POLA = ~PIRQA_L | SERINTA;
PIRQ_POLB = ~PIRQB_L | SERINTB;
PIRQ_POLC = ~PIRQC_L | SERINTC;
PIRQ_POLD = ~PIRQD_L | SERINTD;
```

```
PCI_IRQx = PIRQ_POLA & (DevB:3x56[3:0] == 4'hx) | PIRQ_POLB & (DevB:3x56[7:4] == 4'hx)
          | PIRQ_POLC & (DevB:3x56[11:8] == 4'hx) | PIRQ_POLD & (DevB:3x56[15:12] == 4'hx);
```

```

PNP_IRQx = PNP_IRQ2 & (DevB:3x44[11:8] == 4'hx) | PNP_IRQ1 & (DevB:3x44[7:4] == 4'hx)
          | PNP_IRQ0 & (DevB:3x44[3:0] == 4'hx);

SCI_IRQx = SCI_IRQ & (DevB:3x42[3:0] == 4'hx);

ISA_IRQx = ~(IRQx & SERIRQx) &
          ~( (DevB:3x56[3:0] == 4'hx) | (DevB:3x56[7:4] == 4'hx) | (DevB:3x56[11:8] == 4'hx)
            | (DevB:3x56[15:12] == 4'hx) | (DevB:3x44[11:8] == 4'hx) | (DevB:3x44[7:4] == 4'hx)
            | (DevB:3x44[3:0] == 4'hx) | (DevB:3x42[3:0] == 4'hx) );

KIRQ1 = IRQ1 & SERIRQ1; // to the USB keyboard emulation logic
KIRQ12 = IRQ12 & SERIRQ12; // to the USB keyboard emulation logic

USB0_IRQ1 = ~ExternalIRQEn & KIRQ1 // signal names from USB OHCI spec
           | EmulationEnable & IRQEn & OutputFull & ~AuxOutputFull;
USB0_IRQ12 = ~ExternalIRQEn & KIRQ12 // signal names from USB OHCI spec
            | EmulationEnable & IRQEn & OutputFull & AuxOutputFull;

ISA_IRQ1 = ~(USB0_IRQ1) &
           ~( (DevB:3x56[3:0] == 4'h1) | (DevB:3x56[7:4] == 4'h1) | (DevB:3x56[11:8] == 4'h1)
             | (DevB:3x56[15:12] == 4'h1) | (DevB:3x44[11:8] == 4'h1) | (DevB:3x44[7:4] == 4'h1)
             | (DevB:3x44[3:0] == 4'h1) | (DevB:3x42[3:0] == 4'h1) );

ISA_IRQ12 = ~(USB0_IRQ12) &
            ~( (DevB:3x56[3:0] == 4'hC) | (DevB:3x56[7:4] == 4'hC) | (DevB:3x56[11:8] == 4'hC)
              | (DevB:3x56[15:12] == 4'hC) | (DevB:3x44[11:8] == 4'hC) | (DevB:3x44[7:4] == 4'hC)
              | (DevB:3x44[3:0] == 4'hC) | (DevB:3x42[3:0] == 4'hC) );

ISA_IRQ14 = ~(IRQ14 & SERIRQE) &
            ~( (DevB:3x56[3:0] == 4'hE) | (DevB:3x56[7:4] == 4'hE) | (DevB:3x56[11:8] == 4'hE)
              | (DevB:3x56[15:12] == 4'hE) | (DevB:3x44[11:8] == 4'hE) | (DevB:3x44[7:4] == 4'hE)
              | (DevB:3x44[3:0] == 4'hE) | (DevB:3x42[3:0] == 4'hE) | DevB:1x08[8] );

ISA_IRQ15 = ~(IRQ15 & SERIRQF) &
            ~( (DevB:3x56[3:0] == 4'hF) | (DevB:3x56[7:4] == 4'hF) | (DevB:3x56[11:8] == 4'hF)
              | (DevB:3x56[15:12] == 4'hF) | (DevB:3x44[11:8] == 4'hF) | (DevB:3x44[7:4] == 4'hF)
              | (DevB:3x44[3:0] == 4'hF) | (DevB:3x42[3:0] == 4'hF) | DevB:1x08[10] );

PIC_IRQx = ~(ISA_IRQx | PCI_IRQx | PNP_IRQx | SCI_IRQx );

NMP_INT = DevB:1x08[8] & IRQ14;
NMS_INT = DevB:1x08[10] & IRQ15;

```

Where:

- x The PIC IRQ number, 1, 3–7, 9–12, 14, and 15.
- PIRQ[A,B,C,D]\_L The input PCI interrupts (with the polarity of the external signals).
- SERINT[y] The PCI interrupts captured from the SERIRQ pin (with the same polarity as the SERIRQ pin).
- SERIRQ[x] The ISA interrupts captured from the SERIRQ pin (with the same polarity as the SERIRQ pin).
- USB0\_IRQ[12,1] Outputs of the keyboard emulation logic from the USB controller.
- EKIRQ[12,1] External keyboard controller's interrupts from IRQ1 and IRQ12 when the interrupt function is selected by GPIO12 and GPIO15.

PNPIRQ[2:0]	PNP IRQ pins (with the polarity specified by the associated GPIO control register).
SCI_IRQ	Active-High SCI interrupt.
IRQx	External interrupt.
PIC_IRQx	The interrupt signals that go to the PIC.

Notes from the interrupt routing equations:

- When a PCI, PNP, or SCI interrupt is enabled onto a PIC\_IRQ, then the external and serial IRQ capability for the IRQ is disabled.
- External IRQs and serial IRQs are expected to be edge triggered.
- PCI pin and SCI interrupts are intended to be level triggered.
- PNP interrupts can be level or edge triggered. The inverter available in the GPIO control register must be used to preserve the polarity from the external signal to the PIC; if this inverter is not selected, then there is an inversion from the external signal to the PIC.
- IRQ14 and IRQ15 change from external interrupts to native mode interrupts driven by the IDE drives if DevB:1x08[8 and 10] are set respectively. As native mode interrupts, they are still expected to be active High (externally); they are combined with PIRQA\_L logic to become level-triggered, active Low signals into the PIC.
- The keyboard and mouse interrupt pins, IRQ1 and IRQ12, are ANDed with the serial IRQ versions to go to the USB keyboard emulation logic. The outputs of this logic enter the routing equations.
- In order for the USB keyboard and mouse emulation interrupts to function properly, either the IRQ1 and IRQ12 pins must be strapped Low or an external keyboard controller must keep the serial IRQ slots for IRQ1 and IRQ12 Low.

### 3.4.2.2 PIC/SMI/NMI/INIT to HyperTransport™ Link Translation

Traditionally, processors have treated SMI, NMI, and INIT as edge triggered interrupts and INTR as a level triggered interrupt. The PIC, however, is required to generate an edge on its output INTR signal for each interrupt. Therefore, it too can be treated like it is edge triggered.

SMI, NMI, and INIT all behave the same way. When an active-going edge is detected on the internal version of these signals, the IC generates the appropriate interrupt message to the host. In order for another interrupt to occur, the signal must be deasserted and then asserted again (matching the historical edge-sensitive behavior of these signals). For these messages, MT = SMI, NMI, or INIT (as appropriate), TM = edge, DM = physical; INTRDEST = 'hFF (all); and VECTOR = 'h00 (does not matter).

When the PIC detects an interrupt, it asserts its internal INTR signal. If the IOAPIC is enabled, then this INTR signal is ignored by the HyperTransport interrupt message logic (with the expectation that the equivalent interrupt will be generated by the IOAPIC). Otherwise, when INTR is asserted, the IC generates an interrupt request HyperTransport message as follows: MT = ExtINT; TM = edge; DM = physical; INTRDEST = 'hFF (all); VECTOR = 'h00 (does not matter). The host responds with an

interrupt acknowledge message to the PIC. The interrupt vector is in the 8 LSBs of the response and the 24 MSBs are zero. The PIC clears the INTR line for a minimum of one PCLK when the interrupt acknowledge is received.

### 3.4.2.3 IOAPIC

The IOAPIC supports 24 interrupt signals which come from the interrupt routing logic, the PCI interrupts, GPIOs, and internal signals. Based on configuration, it sends interrupt messages through the HyperTransport bus.

#### 3.4.2.3.1 IOAPIC Redirection Registers

The IOAPIC supports 24 interrupt request signals. Each interrupt request input is combined with its corresponding redirection register to specify the behavior of the interrupt. These interrupt request signals are connected to redirection registers (APIC IRQs) as shown in the following table.

**Table 16. IOAPIC Redirection Register Connections**

APIC IRQ	Connection	APIC IRQ	Connection
0	PIC INTR output	12	PIC IRQ12
1	PIC IRQ1	13	PIC IRQ13 (floating-point error)
2	PIC IRQ0 (PIT)	14	PIC IRQ14
3	PIC IRQ3	15	PIC IRQ15
4	PIC IRQ4	16	PIRQA_L
5	PIC IRQ5	17	PIRQB_L
6	PIC IRQ6	18	PIRQC_L
7	PIC IRQ7	19	PIRQD_L
8	PIC INT8 (RTC)	20	GPIO28
9	PIC IRQ9	21	GPIO29
10	PIC IRQ10	22	GPIO30
11	PIC IRQ11	23	GPIO31

See Section 3.4.2.1 on page 41 for the definition of the “PIC IRQ” signals. See DevB:0x4B[MPIRQ] for a description of how the mask bits from redirection table entries[23:20] can affect the routing of GPIO[31:28] onto PIRQ[A,B,C,D]\_L. See DevB:3x44[TCO\_INT\_SEL] for a description of how other interrupts may be directed to the IOAPIC.

### 3.4.2.4 NMI

NMI is controlled as described in the following equation:

```

NMI      = ~PORT70[NMIDIS] &
          ( PM48[NMI_NOW]
          | ~PM48[NMI2SMI_EN] &
            ( PORT61[SERR] &      ~PORT61[CLRSERR]
            | PORT61[IOCHK] &    ~PORT61[CLRIOCHK]
            | DevB:0x40[NMIONERR] & [status bits described in section 3.1.2]
            | DevA:0x1C[MDPE] &   DevA:0x3C[PEREN] ) );

```

### 3.4.3 Watchdog Timer (WDT)

The watchdog timer is a down counter starting at a programmed value. It resets or shuts down the system if the count reaches zero. Operating system services periodically restart the timer so that if the operating system, drivers or services stop functioning, the system is automatically restarted or shut down.

The watchdog timer is enabled or disabled by programming DevB:0xA8. When disabled, the watchdog timer stops counting and cannot be started by the operating system. When enabled, the watchdog timer supports two sub-states, the Running and the Stopped states. The watchdog timer allows the operating system to set it in either Running or Stopped state by programming WDT00[RSTOP]. In the Enabled/Stopped state the timer does not count down. In the Enabled/Running state the counter counts down to zero once triggered. These states are visible to the operating system through WDT00[WDE\_ALIAS] and WDT00[RSTOP].

When DevB:0xA8[WDSILENT] is set, the watchdog timer operates in silent mode. In that mode no action, as defined in WDT00[WACT], is caused when the timer expires. It is in the responsibility of external software to cause either a power down/power up or a system reset and to clear WDT00[WFIR] directly.

The status of WDT00[WFIR] can be observed externally on GPIO4 when the GPIO is programmed for its alternate function through PMC4.

The count down time range can be programmed by writing WDT08.

### 3.4.4 High Precision Event Timer (HPET)

#### 3.4.4.1 Overview

The HPET consists of a block of three timers. This block contains a 32-bit up counter with three 32-bit output comparators each for one timer. Timer 0 can operate in either periodic or non-periodic mode, timer 1 and 2 only in non-periodic mode.

**Table 17. HPET Specifications**

Item	Implementation
Main Counter	32-bit up-counter
Clock Frequency	14.31818 MHz
Number of comparators	3 32-bit comparators
Number of Periodic Capable Timer	1 Timer 0
Number of One-Shot Capable Timer	3 Timer 0, 1, 2
Interrupt Routing	Through PIC, IOAPIC and HyperTransport link

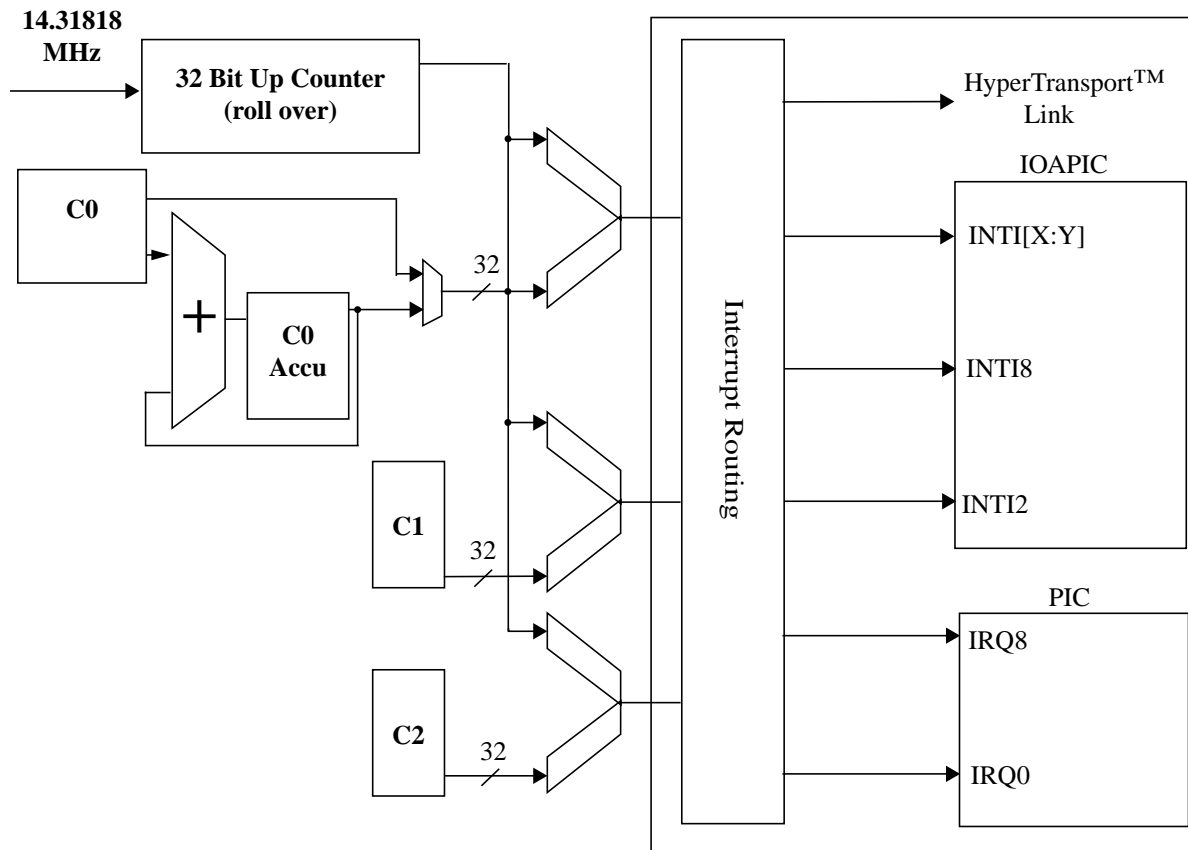


Figure 4. High Precision Event Timer Block Diagram

#### 3.4.4.1.1 Periodic Mode

Only Timer 0 supports the periodic mode.

In the periodic mode the comparator consists of a comparator value and an accumulator register. Depending on the state of HPET100[SETVAL], a write to HPET108 is stored either to the comparator value register or to the accumulator register. If the main counter value matches the accumulator value, an interrupt is generated (if enabled) and the last value written to the comparator value register is added to the accumulator value register. In periodic mode the value of the accumulator register is given back for read accesses to HPET108.

To change the comparator value register, the software should either halt the main counter or disable the comparator to avoid races.

#### **3.4.4.1.2 Non-Periodic Mode**

All timers support the non-periodic (one-shot) mode.

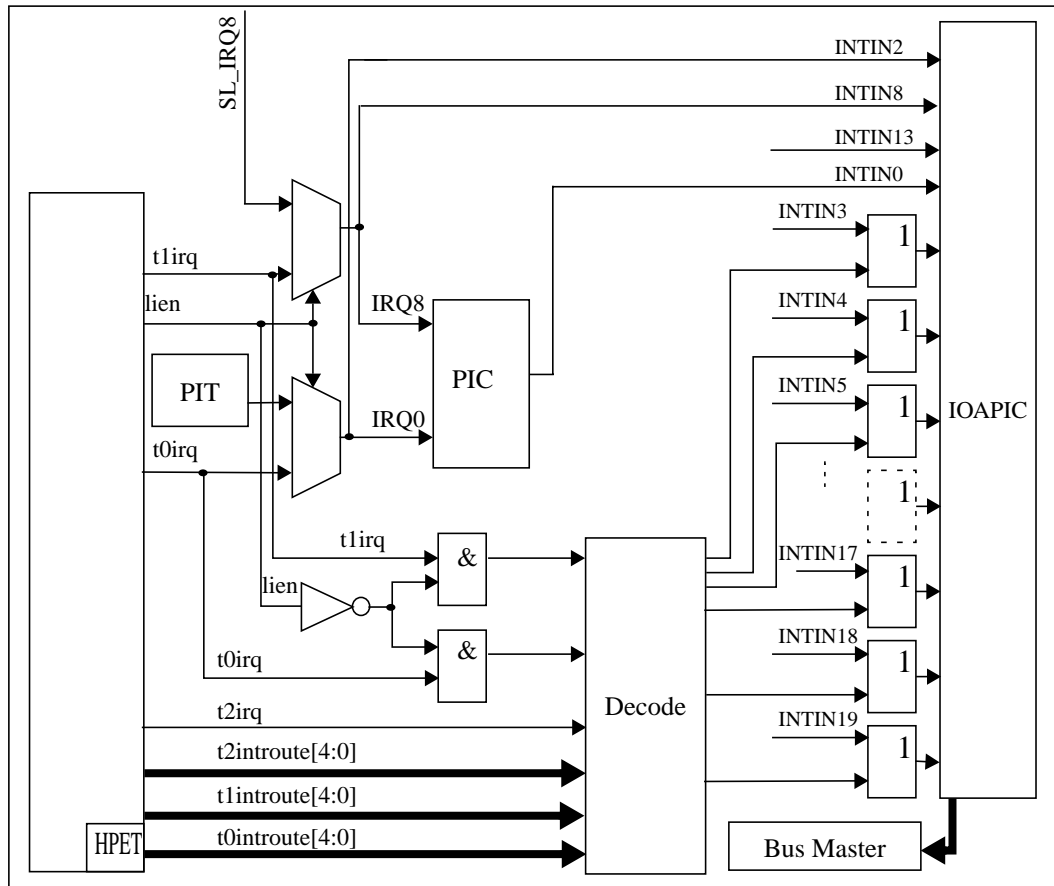
The timer generates an interrupt if either the comparator value matches the main counter value or the main counter wraps around. During runtime the hardware does not change the value of the comparator value register, but the software may reprogram it.

#### **3.4.4.1.3 Interrupt Routing**

The timer interrupts can be routed to the PIC and/or the IOAPIC. In legacy mode (HPET10[LIEN] set to 1b), Timer0 interrupt is routed to IRQ0/INTIN2 and Timer1 interrupt to IRQ8/INTIN8 of PIC/IOAPIC. Timer2 is only routed directly to the IOAPIC. In non-legacy mode all three timers are routed to the IOAPIC.

The timers can be routed to all IOAPIC inputs with the exception of INTIN0, INTIN2, INTIN8, INTIN13 and INTIN20 to INTIN23.





**Note:** High Precision Event Timers can be routed only to INTINx inputs 1, 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, 15, 16, 17, 18 and 19 (i.e., all except 0, 2, 8, 13 and 20 to 23).

**Figure 5. High Precision Event Timer Interrupt Routing**

### 3.4.4.2 Register Mapping

All HPET registers are memory mapped and 64-bit aligned. The BIOS reports the base address location and the allocated space to the operating system. The operating system should never change the base address register. This implementation consists of a 1-Kbyte address space. The base address register for these registers is DevB:0xA0.

Software should not write to read-only registers. Software must not attempt to read or write across register boundaries. That means 32-bit accesses are only allowed on 32-bit boundaries. Software should perform read-modify-write on reserved bits.

### 3.4.4.3 Power Management

The Timer Registers are not powered during S3, S4 and S5. All functions of the timer are only used during S0, and all interrupts should be disabled before leaving S0.

### 3.4.5 Real-Time Clock (Logic Powered by VDD\_COREAL)

The real-time clock logic requires an external 32-kHz oscillator. It includes a clock and calendar timer, an alarm which generates an interrupt, and 256 bytes of non-volatile RAM. It is register compatible with the legacy PC real-time clock. It meets ACPI real-time clock requirements. The real-time clock resides on the VDD\_COREAL power plane.

## 3.5 Enhanced IDE Controller

The enhanced IDE controller supports independent primary and secondary ports. Each port supports two drives. Supported protocols include PIO modes 0-4, multi-word DMA modes 0-2, ultra DMA modes 0-1, 2 (ATA-33), 3, 4 (ATA-66), 5 (ATA-100), and 6 (ATA-133).

The IDE ports can be individually controlled through DevB:1x54 such that the drives can be powered down.

## 3.6 System Management Bus 2.0 Controller

### 3.6.1 Functional Overview

The SMBus controller is designed to implement an SMBus 2.0 compliant host and slave device. The SMBus controller constitutes a PCI function with its according PCI configuration header. Host accesses to the SMBus controller are accomplished through an ACPI 2.0 Chapter 13 compliant host interface comprising a data and a status/command port. See the ACPI 2.0 specification, Sections 13.1–13.7 for operational details of that interface.

#### *Notes:*

- 1. The SMBus controller target state machine accepts neither non-defined commands nor out-of-order data accesses like write accesses to the data port without a preceding write access to the command port. In those situations, the access is terminated by a master abort.*
- 2. An out-of-order write access to the command port re-initializes the target state machine to that latter command. To this effect a write access to the command port starting a write command following a preceding read command terminates that read command and re-initializes the target state machine to the write command.*

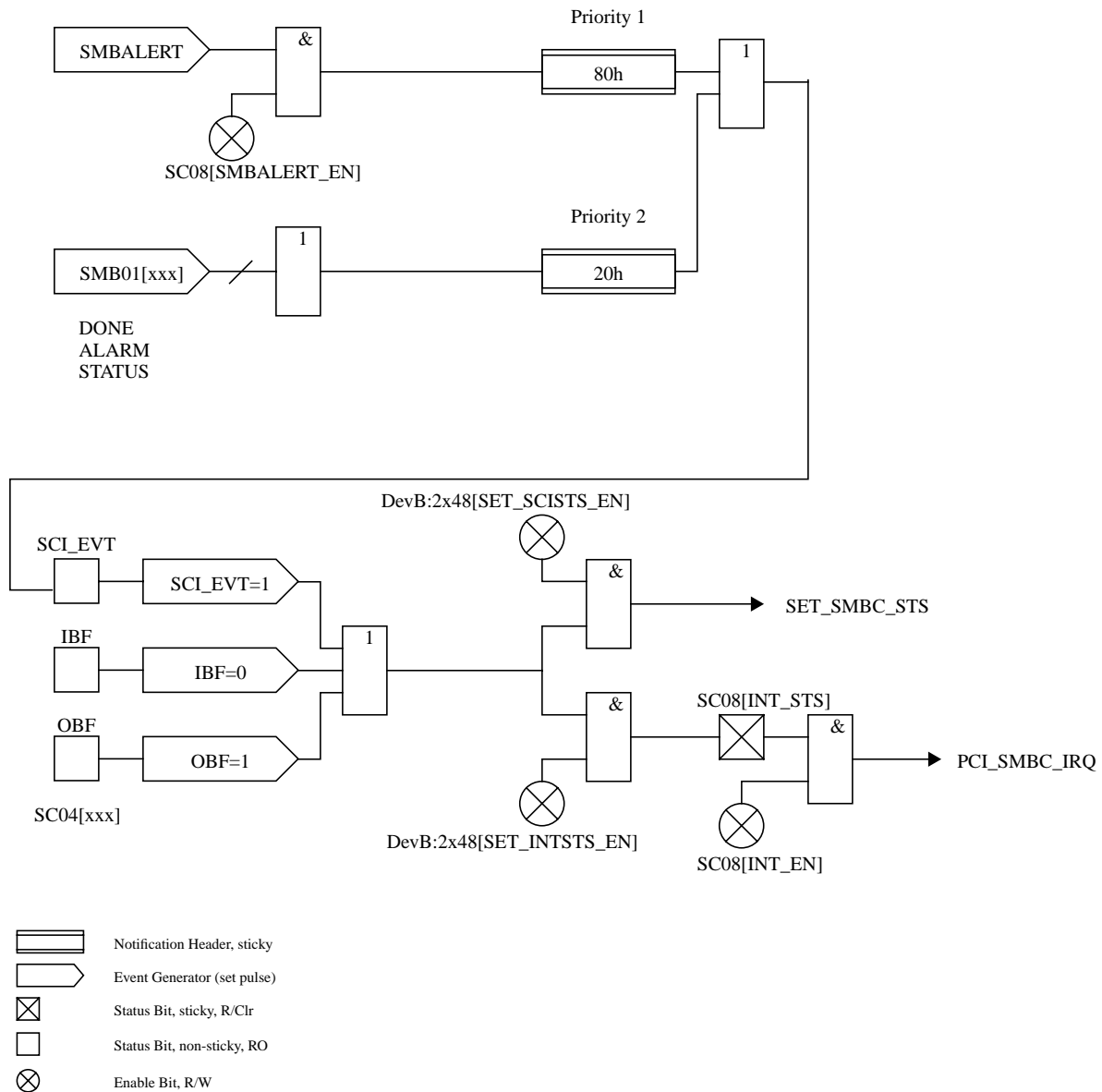
### 3.6.2 Interrupts

See Figure 6 on page 52 for the interrupt structure of the SMBus controller. All interrupts can be configured by DevB:2x48[SET\_SCISTS\_EN, SET\_INTSTS\_EN] to generate an SMI or SCI through PM20[SMBC\_STS] or a PCI interrupt through SC08[INT\_STS]. The internal interrupt sources comprise the events associated with SC04[SCI\_EVT, OBF, IBF] as described by chapter 13 of the

ACPI 2.0 specification. SC04[SCI\_EVT] comprises three sources of events with each source having an unique notification header associated with it (see Table 18). If there are more than one with SC04[SCI\_EVT] associated event pending, the notification header with the highest priority is returned upon a query command from the host with priority 1 being the highest priority.

**Table 18. SC04[SCI\_EVT] Event Sources**

Event source	Priority	Notification Header
SMBALERT	1	80h
SMBus host controller	2	20h



**Figure 6. SMBus Controller Interrupt Model**

Note that all notification headers are set by event generators, i.e., by edge sensitive sources. In the case of an SMBALERT detection, software is expected to query the attached SMBus devices until all devices that asserted SMBALERT in parallel are serviced.

## 3.7 System Management Logic

System management includes logic for most of the multiplexed-function pins—such as general-purpose I/O (GPIO) pins, the power management (PM) pins, system management bus 1.0 (SMBus 1.0) pins, the processor interface pins, and the plug and play (PNP) interrupt pins—as well as the logic required for ACPI-compliant power management for desktop and mobile systems.

Programmable register access to most of this logic is contained in the DevB:3xXX configuration space and the PMxx I/O space. The major functions are:

- ACPI interrupt (SCI or SMI based on the state of PM04[SCI\_EN] status bits and enables)
- SMI status bits and enables
- System power state machine (SPSM)
- Resume event logic (to place the SPSM into the full-on state)
- SMBus 1.0 interface
- System power state control pins and general purpose pins
- Device monitors (hardware traps, interrupt traps, DMA request traps, re-trigger timers)
- System inactivity timer
- Serial IRQ logic
- CLKRUN\_L control
- Watchdog Timer

### 3.7.1 Power Management

The following table summarizes all the system management events that can be detected by the system management logic and the hardware response enable registers. The columns are: SCI/SMI Parent STS/EN, where the status and enable bits for ACPI interrupts are accessible; and SMI-only STS/EN, where the status and enable bits for SMI interrupts are accessible.

**Table 19. System Management Events**

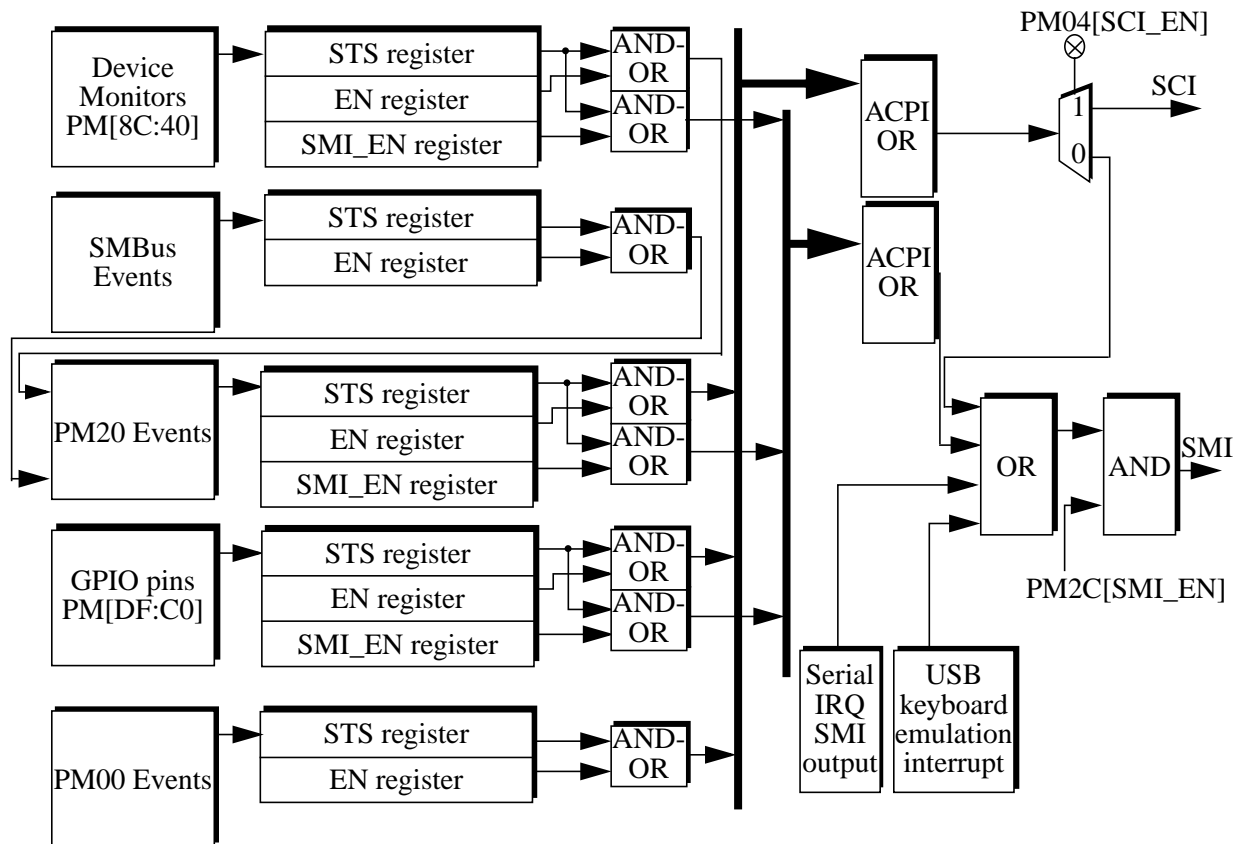
Events	SCI/SMI Parent STS/EN	SMI-only STS/EN	Notes, Other Functions
OS release (PM04[GBL_RLS])		PM28/2A	SMI only
BIOS release (PM2C[BIOS_RLS])	PM00/02		SCI only
Software SMI through PM1E, PM2F		PM28/2A	SMI only
Device monitors; GP timer time out	PM20/22	PMA0/A8	See PM[8C:50] for definitions; child STS/EN bits in PMA0/A4
System inactivity timer time out	PM20/22	PM20/2A	See PM98 for SIT; reload registers in PM[8C:50] and PMAC
ACPI timer overflow	PM00/02		SCI only
SMBus 1.0 events	PM20/22	PM28/2A	Child STS/EN bits in PME0/E2

**Table 19. System Management Events (Continued)**

Events	SCI/SMI Parent STS/EN	SMI-only STS/EN	Notes, Other Functions
USB bus resume event	PM20/22	PM20/2A	
AC '97 events	PM20/22	PM20/2A	
SMBus 2.0 events	PM20/22	PM20/2A	
Power button override	PM00		No SCI; PM26 control only used to go to SOFF
TCO events	PM20/22	PM28/2A	
Miscellaneous SMI events		PM30/32	
PIC INTR signal (unmasked IRQs)			Reload SIT through PMAC; also causes C2/C3 resume; POS resume through PM28/2A
Real-time clock IRQ	PM00/02		SCI only
NMI to processor			Reload SIT through PMAC; also causes C2/C3/POS resume
INIT to processor			Reload SIT through PMAC; also causes C2/C3/POS resume
SMI to processor			Also causes C2/C3/POS resume
PCI bus masters	PM00		Resumes from C3 with no SCI; reload SIT through PMAC; re-trigger timer through PM54
32 GPIO inputs	PMB0/B4	PMB0/B8	
PWRBTN_L pin	PM00/02	PM00/2A	
EXTSMI_L pin	PM20/22	PM002A	
PME_L pin	PM20/22	PM20/2A	
RI_L pin	PM20/22	PM20/2A	
SLPBTN_L pin	PM00/02	PM00/2A	
THERM_L pin	PM20/22	PM20/2A	
LID pin	PM20/22	PM20/2A	
ACAV pin	PM20/22	PM20/2A	

### 3.7.1.1 SCI And SMI Control

System management events cause corresponding STS registers to be set. STS registers can be enabled to generate SCI and SMI interrupts. The following diagram shows how the STS registers are routed to the interrupts.



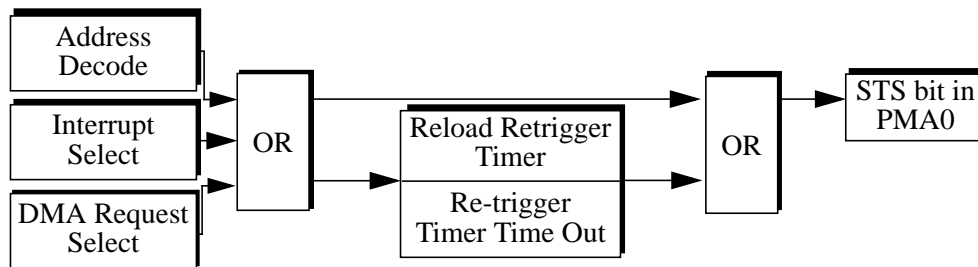
**Figure 7. STS to Interrupt Routing**

The “AND-OR” boxes in the middle of the diagram specify the logical AND of the STS and EN registers (or STS and SMI\_EN registers, as the case may be), the results of which are logically ORed together; for example:  $(STS1 \ \& \ EN1) \ | \ (STS2 \ \& \ EN2) \dots$ , and so on.

All enabled ACPI interrupts may be routed to either SCI or SMI interrupts by PM04[SCI\_EN]. Or these STS registers may be routed directly to SMI by using the SMI\_EN registers, regardless of the state of PM04[SCI\_EN]. The USB controllers and the serial IRQ logic also provide sources of SMI that is ORed into the logic. SMI and SCI are inputs to the interrupt routing logic; see Section 3.4.2 on page 40.

### 3.7.1.2 Device Monitors And Re-Trigger Timers

Device monitors consist of a set of registers that may be used to enable traps on transactions, interrupts, DMA activity, and device inactivity. Each monitor circuit includes transaction decode logic, optional interrupt and DMA channel enables, a re-trigger timer that gets reloaded every time the monitored event occurs, and enable bits for passing events to SCI or SMI interrupts. Interrupt device monitor events occur whenever the monitored interrupt signal changes state (High to Low or Low to High). DMA device monitor events occur whenever the monitored DMA request signal (to the legacy 8237 DMA controller from the LPC interface logic) is asserted.



**Figure 8. Device Monitors and Retrigger Timers**

### 3.7.1.2.1 Traps

Configuration registers DevB:3x[D8:B4] specify several traps for memory, I/O, and configuration space address ranges. These traps are generated for the specified transactions that (1) are targeted at the IC or any device on or behind the secondary PCI bus or any devices on the LPC bus, (2) are not targeted to the configuration space or I/O space of the IDE controller, and (3) are not targeted at any of the DevA:0x[FF:C0] configuration registers.

### 3.7.1.3 System Inactivity Timer

Any of the hardware traps, IRQ lines, or PCI bus master activity can be enabled to reload the system inactivity timer (SIT). If the SIT decrements to zero, then, if enabled, an interrupt is generated.

### 3.7.1.4 Throttling Logic

When throttling, the IC repetitively places the processor into the Stop Grant state for a specified percentage of time in order to reduce the power being consumed by the processor. STPCLK\_L is used to control the processor Stop Grant state with a period specified by DevB:3x40[NTPER, TTPER] and a duty cycle specified by DevB:3x40[THMINEN], DevB:3x4D and PM10.

Two types of throttling are possible: normal and thermal. Normal throttling is controlled by software. Thermal throttling is controlled by the THERM\_L pin (see also DevB:3x40[TH2SD]). If both are triggered active simultaneously, then the duty cycle specified for thermal throttling is used. Throttling is only possible when in the FON state. If throttling is enabled when entering other states, then it stops; after exiting the state, throttling resumes.

### 3.7.1.5 CLKRUN\_L PCI Bus Clock Control

PCI bus clock control is performed with the CLKRUN\_L signal and associated protocol as defined by the PCI Mobile Design Guide. The IC is the “Central Resource” with regard to CLKRUN\_L protocol.

#### 3.7.1.5.1 Enabling PCI CLKRUN\_L Protocol

CLKRUN\_L protocol is enabled by PMC5. CLKRUN\_L assertion by an external device is ignored during the S1 sleep state.



### 3.7.1.5.2 PCISTOP\_L Assertion

The PCLK input to the IC is required to always operate regardless of the CLKRUN\_L protocol. The PCI clock to external devices is stopped when the IC asserts PCISTOP\_L.

If the PCI bus has been idle for 26 PCI clocks (internally and externally) and there is no host access to or bus master request from any device in the IC other than IDE, the IC deasserts CLKRUN\_L by driving it High for one PCI clock period and then three-stating it. If the IC does not sample CLKRUN\_L asserted by an external device within 3 PCI clock periods of CLKRUN\_L deassertion, then the IC asserts PCISTOP\_L.

### 3.7.1.5.3 PCISTOP\_L Prevention

If the IC has detected an idle condition as specified in the previous section and has deasserted CLKRUN\_L, an external device may keep the external PCI clock running by asserting CLKRUN\_L within 2 clocks of CLKRUN\_L deassertion. If the IC detects CLKRUN\_L asserted within 3 clocks of deasserting CLKRUN\_L, then the IC reasserts CLKRUN\_L, clears its 26 clock idle counter and does not assert PCISTOP\_L.

If the IC has deasserted CLKRUN\_L and detects any host access to or bus master request from any device in the IC other than IDE or an ACPI system state transition within 3 PCI clocks of deasserting CLKRUN\_L, then the IC reasserts CLKRUN\_L, clears its 26 clock idle counter and does not assert PCISTOP\_L.

### 3.7.1.5.4 PCISTOP\_L Deassertion

The IC deasserts PCISTOP\_L under the following conditions:

- It detects CLKRUN\_L asserted by an external device.
- It detects any host access to or bus master request from any device in the IC other than IDE.
- It detects a resume event from S1.
- It asserts RESET\_L.

The IC restarts the PCI clock to external devices by deasserting PCISTOP\_L. The IC asserts CLKRUN\_L when it deasserts PCISTOP\_L.

The CLKRUN\_L pin requires a large (100-Kohm or greater) external pull-up resistor to the VDD\_IO plane.

### 3.7.1.6 System Power State Controller (SPSC)

The system power state controller (SPSC) supports the following system power states:

**Table 20. System Power States**

System Power State	VDD_IO, VDD_CORE	VDD_IOX, VDD_COREX	VDD_RTC
Full on (FON; S0)	On	On	On
C2	On	On	On
C3	On	On	On
Power on suspend (POS; S1)	On	On	On
Suspend to RAM (STR; S3)	Off	On	On
Soft off (SOFF; S5); suspend to disk (STD; S4)	Off	On	On
Mechanical off (MOFF; G3)	Off	Off	On

**Mechanical off (MOFF or ACPI G3 state).** MOFF is the state when only VDD\_RTC is powered. This can happen at any time, from any state, due to the loss of power to the AUX planes (e.g., a power outage, the power supply is unplugged, or the power supply's mechanical switch). When power is applied to AUX, then the system transitions to either FON or SOFF.

**Soft off (SOFF or ACPI G2/S5 states).** In the SOFF state, the system appears to the user to be off. The AUX planes of the IC are powered, but the main supplies are not; RPWRON is Low to disable power system DRAM. The system normally uses PWRBTN\_L to transition from SOFF to FON. Table 21 on page 59 lists all resume events that may cause a transition from SOFF to FON.

**Suspend to disk (STD or ACPI S4 state).** The IC's behavior in this state equivalent to SOFF.

**Suspend to RAM (STR or ACPI S3 state).** In the STR state, the system's context is stored in system memory (which remains powered; RPWRON is High) and the main power supplies are shut off (PWRON\_L High). The behavior of the IC in the STR state is similar to SOFF; the main difference is that RPWRON is asserted in STR.

**Power on suspend (POS or ACPI S1 state).** All power planes to the IC are valid in POS. Signal control during POS is specified by DevB:3x50. Table 21 on page 59 lists all resume events that may cause a transition from POS to FON.

**Stop Grant caches snoopable (C2).** In C2, the processor is placed into the Stop Grant state. Signal control during C2 is specified by DevB:3x4F. It is expected that the processor's cache may be snooped while in this state. Table 21 on page 59 lists all resume events that may cause a transition from C2 to FON.

**Stop Grant caches not snoopable (C3).** In C3, the processor is placed into the Stop Grant state such that the processor's cache cannot be snooped; requests that result in snoops are resume events (see PM04[BM\_RLD]). Signal control during C3 is specified by DevB:3x4F. Table 21 on page 59 lists all resume events that may cause a transition from C3 to FON.

**Full on (FON or ACPI S0).** In FON, all the power planes are powered and the processor is not in the Stop Grant state.

Figure 9 shows the system power state transitions.

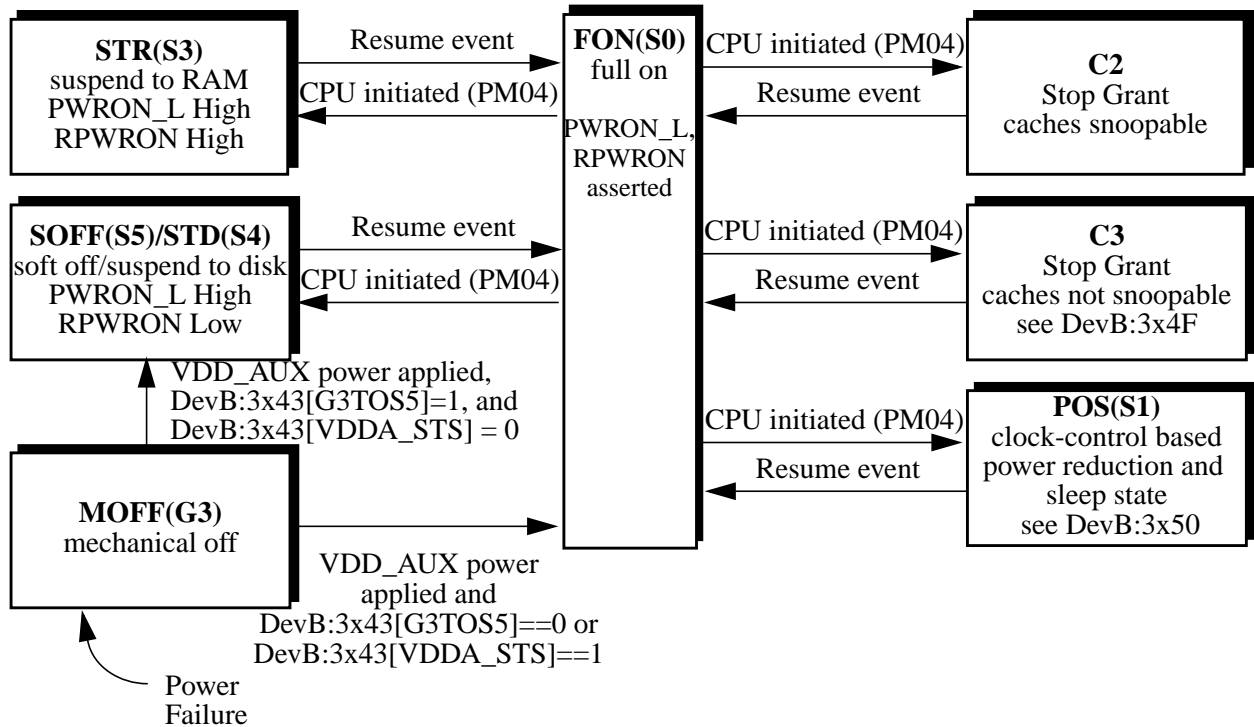


Figure 9. System Power State Transitions

3.7.1.6.1 Summary of Resume Events

In general, resume events occur when there is an event that sets a status bit while the ACPI interrupt enable for that status bit is active. If LDTSTOP\_L is asserted in the low-power state, then resume events do not initiate the resume sequence until the minimum assertion time of 1 microsecond for LDTSTOP\_L has completed.

Table 21. Resume Events

Resume Event	Resumes from Sleep States	Resume Event Enable
PWRBTN_L	S5-S1, C3/C2	N/A
SLPBTN_L	S5-S1, C3/C2	PM02[SLPBTN_EN]
RTC Alarm	S5-S1, C3/C2	PM02[RTC_EN]
RI_L	S5-S1, C3/C2	PM22[RI_EN]
PME_L	S5-S1, C3/C2	PM22[PME_EN]
Ethernet controller PME	S5-S1, C3/C2	PM22[PME_EN], Ethernet controller PM registers
ACAV	S5-S1, C3/C2	PM22[ACAV_EN]
LID	S5-S1, C3/C2	PM22[LID_EN]
SMBus 1.0 events	S5-S1, C3/C2	PM22[SMBUS_EN]
SMBALERT0_L	S1, C3/C2	PM22[SMBUS_EN], PME0[SMBA_STS]
USB events	S5-S1, C3/C2	PM22[USBRSM_EN]

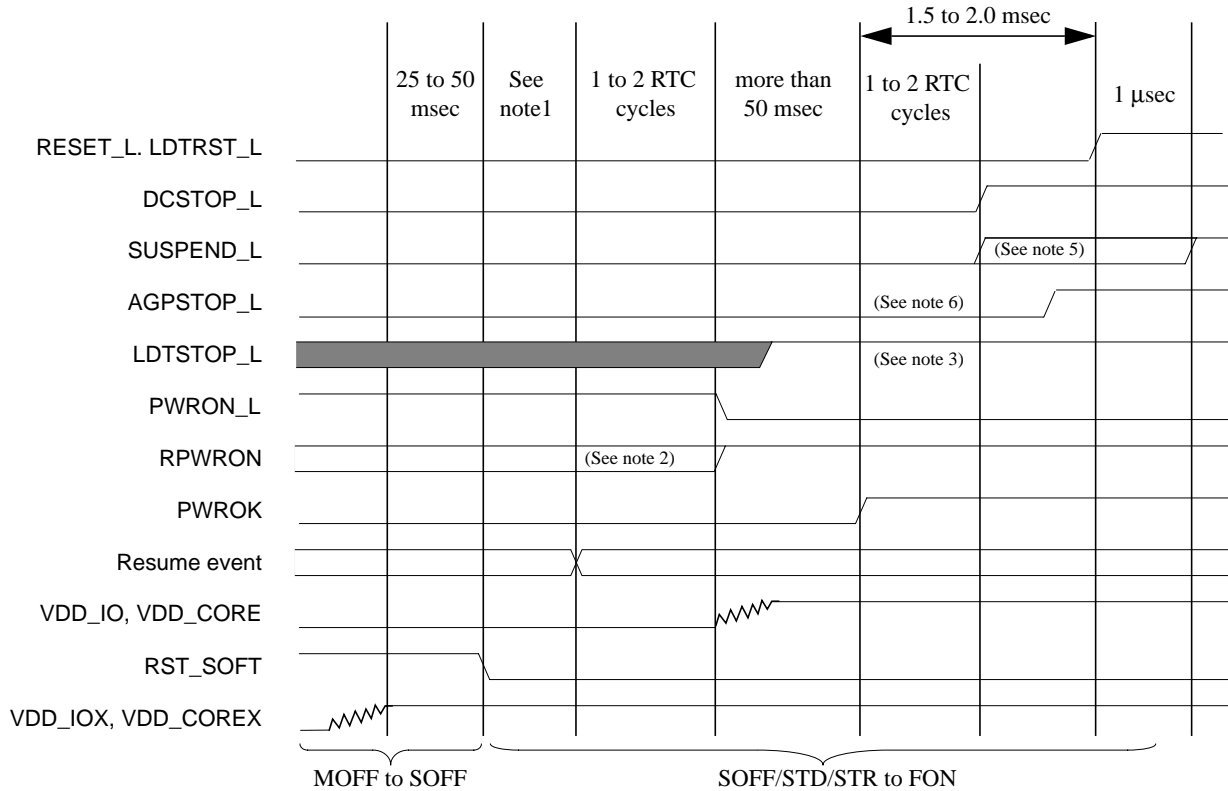
**Table 21. Resume Events (Continued)**

Resume Event	Resumes from Sleep States	Resume Event Enable
EXTSMI_L	S5-S1, C3/C2	PM22[EXTSMI_EN]
SMBus 2.0 events	S5-S1, C3/C2	PM22[SMBC_EN]
SMBALERT1_L	S5-S1, C3/C2	PM22[SMBC_EN], SC08[SMBALERT_EN]
AC '97 events	S5-S1, C3/C2	PM22[AC97_EN]
GPIO[25, 24, 23, 22, 18, 14, 3, 2, 1, 0]	S5-S1, C3/C2	PMB4[25, 24, 23, 22, 18, 14, 3, 2, 1, 0]
THERM_L	S1, C3/C2	PM22[THERM_EN]
TCO SCI	S1, C3/C2	PM22[TCOSCI_EN]
System inactivity timer	S1, C3/C2	PM22[SIT_EN]
Device monitor events	S1, C3/C2	PM22[DM_EN]
Device monitor events	S1, C3/C2	PMA4
GPIO[31:26, 21:19, 17:15, 13:4]	S1, C3/C2	PMB4[31:26, 21:19, 17:15, 13:4]
Unmasked interrupt	S1, C3/C2	PM2A[IRQ_RSM] for S1, N/A for C3/C2
NMI, INIT, SMI	C3/C2	N/A
Bus master request	C3	PM04[BM_RLD]

PM26[BATLOW\_CTL] can be used to inhibit resume events in S5-S1.

### 3.7.1.6.2 Transitions between MOFF/SOFF/STD/STR and FON

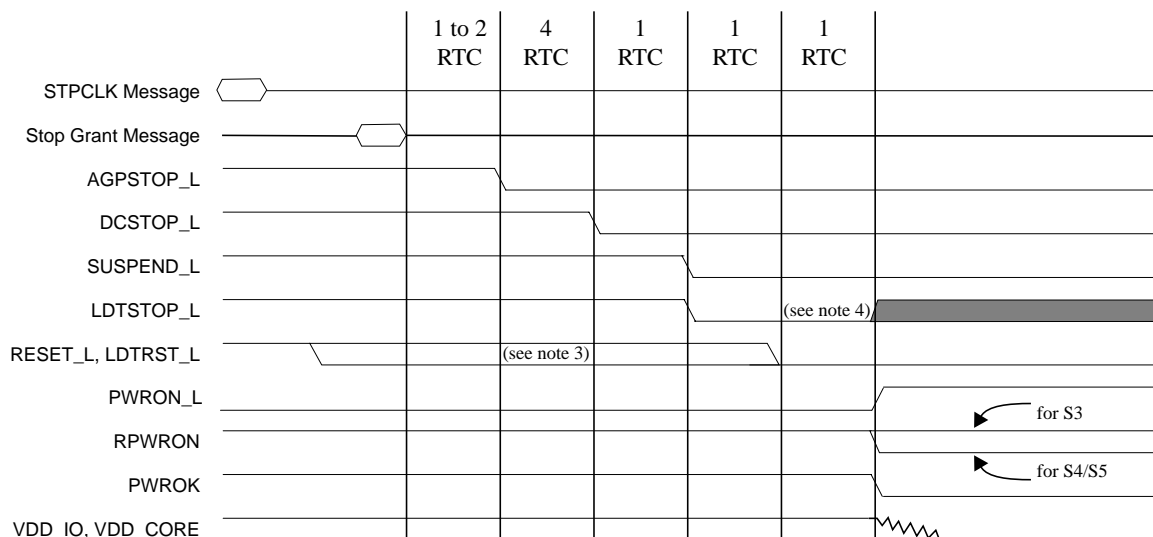
In the following timing diagrams, RTC cycles refer to 32 kHz clocks cycles.



**Notes:**

1. If DevB:3x43[G3TOS5] = 0, then the time from the end of RST\_SOFT to PWRON\_L assertion is 1 to 2 RTC clocks. If DevB:3x43[G3TOS5] = 1, then the resume event must occur before PWRON\_L is asserted.
2. RPWRON is High during STR and Low during STD and SOFF.
3. LDTSTOP\_L powers up deasserted simultaneously with the VDD\_IO power plane.
4. VDD\_IO is representative of all MAIN power planes. VDD\_IOX is representative of all AUX power planes.
5. When transitioning from STR to FON, SUSPEND\_L is not deasserted until one microsecond after RESET\_L has been deasserted.
6. When transitioning from SOFF/STD/STR to FON, AGPSTOP\_L deassertion is delayed from DCSTOP\_L deassertion by 500us to 800us.

**Figure 10. Transitions from MOFF to SOFF and from SOFF/STD/STR to FON**



**Notes:**

1. For transitions to SOFF that are initiated by a power/sleep button override event, or THERMTRIP\_L assertion, or PORTCF9[FULLRST], the STPCLK\_L assertion and Stop Grant cycles are skipped; the sequence starts with the assertion of AGPSTOP\_L.
2. STPCLK is transmitted by a HyperTransport system management message with the 3-bit System Management Action Field (SMAF) as defined by DevB:3x70[STRSMAF] or DevB:3x70[S45SMAF]. This STPCLK system management message is issued before the response message to the write of PM04 is issued.
3. RESET\_L is asserted within 1200ns of THERMTRIP\_L assertion.
4. These signals go undefined when power is removed.

**Figure 11. Transitions from FON to SOFF/STD/STR**

**3.7.1.6.3 Transitions From any state to G3 (MOFF)**

Transitions from any state to G3 (MOFF) occur whenever there is a power failure. No specific sequence is ensured when a power failure occurs. However the IC isolates its AL power plane from all other power planes when a transition to G3 occurs.

**3.7.1.6.4 Transitions From S0/C0 (FON) to C2/C3**

DevB:3x4F specifies the enabled functions for transitions to C2 and C3. The sequence of entering C2/C3 is as follows:

- **Processor initiation.** Software initiates the transition to C2 by reading PM14 or to C3 by reading PM15.
- **Stop-grant.** IC issues a HyperTransport STPCLK system management message with the 3-bit System Management Action Field (SMAF) as defined by DevB:3x70[C2SMAF] for C2 or DevB:3x70[C3SMAF] for C3. This STPCLK system management message is issued before the

response message to the read of PM14 or PM15. The IC then waits for a Stop Grant special cycle broadcast message from the host. When the IC receives the broadcast Stop Grant message, the transition to C2 is complete; the remaining steps only apply to the transition to C3.

- **AGPSTOP\_L.** If DevB:3x4F[ASTP\_C3EN]=1b, the IC asserts AGPSTOP\_L when it recognizes the Stop Grant special cycle associated with the C3 state.
- **CPUSLEEP\_L.** If DevB:3x4F[CSLP\_C3EN]=1b, the IC asserts CPUSLEEP\_L when it recognizes the Stop Grant special cycle associated with the C3 state.
- **LDTSTOP\_L.** After the Stop Grant special cycle is recognized, the IC waits the time interval dictated by DevB:3x74[C3S1LST], then if DevB:3x70[C3LS]=1b the IC asserts LDTSTOP\_L.

### 3.7.1.6.5 Transitions From C2 to C0 (FON)

When the IC detects an enabled resume event, it issues a HyperTransport STPCLK system management message with the STPCLK bit deasserted.

### 3.7.1.6.6 Transitions From C3 to C0 (FON)

The following is the resume sequence from C3, once an enabled resume event occurs. Note: if CPUSLEEP\_L is not enabled to be asserted in C3 by DevB:3x4F, then the C3 resume sequence starts from “LDTSTOP\_L” below:

- **CPUSLEEP\_L.** If DevB:3x4F[CSLP\_C3EN]=1b then, when an enabled resume event occurs, CPUSLEEP\_L is deasserted and a 250 μs delay is inserted before the rest of the C3 resume sequence occurs. If DevB:3x4F[CSLP\_C3EN]=0b then CPUSLEEP\_L is not part of the C3 sequence, and the 250 μs delay is not inserted between recognizing an enabled resume event, and deassertion of LDTSTOP\_L.
- **LDTSTOP\_L.** If DevB:3x70[C3LS]=1b then LDTSTOP\_L is deasserted when the IC detects an enabled resume event (unless CPUSLEEP\_L was asserted). See DevB:3x74[FVLST].
- **AGPSTOP\_L.** If CPUSLEEP\_L is used and LDTSTOP\_L is not used then AGPSTOP\_L is deasserted 250 μs after CPUSLEEP\_L deassertion (as mentioned above). If LDTSTOP\_L is used then after LDTSTOP\_L deassertion a delay dictated by DevB:3x52[C3\_ASTP\_DT] occurs before AGPSTOP\_L is deasserted. If LDTSTOP\_L and CPUSLEEP\_L are not used then after a programmable time period dictated by DevB:3x52[C3\_ASTP\_DT] after the resume event AGPSTOP\_L is deasserted.
- After LDTSTOP\_L and AGPSTOP\_L are deasserted, the IC issues a HyperTransport STPCLK system management message with the STPCLK bit deasserted.

### 3.7.1.6.7 Transitions From FON (S0/C0) to POS (S1)

DevB:3x50 specifies the enabled functions for transitions to S1. The transition to S1 occurs as follows for each of the enabled pin controls:

- **Processor initiation.** Software initiates the transition to S1 by writing the appropriate value to PM04[SLP\_TYP, SLP\_EN].

- **Stop-grant.** IC issues a HyperTransport STPCLK system management message with the 3-bit System Management Action Field (SMAF) as defined by DevB:3x70[POSSMAF]. This STPCLK system management message is issued before the response message to the write of PM04. The IC then waits for a Stop Grant special cycle broadcast message from the host.
- **CPUSLEEP\_L and CPUSTOP\_L.** If DevB:3x50[CSLP]=1b, DevB:3x50[CSTP]=1b, then CPUSLEEP\_L, CPUSTOP\_L respectively are asserted when the IC recognizes Stop Grant special cycle broadcast.
- **AGPSTOP\_L.** If DevB:3x50[ASTP]=1b then AGPSTOP\_L is asserted immediately after the Stop Grant broadcast has been recognized by the IC.
- **DCSTOP\_L.** If DevB:3x50[DCSTP]=1b the IC waits a minimum of 122 μs after Stop Grant is recognized and then asserts DCSTOP\_L.
- **SUSPEND\_L.** If DevB:3x50[SUSP]=1b SUSPEND\_L is asserted at the same time that DCSTOP\_L is asserted. If DCSTOP\_L is not enabled SUSPEND\_L assertion is delayed a minimum of 122 μs from Stop Grant.
- **PCISTOP\_L.** If DevB:3x50[PSTP]=1b after DCSTOP\_L or SUSPEND\_L assertion (or Stop Grant if neither are enabled) the IC then waits at least one RTC clock before asserting PCISTOP\_L. PCI CLKRUN\_L protocol is not in effect during S1.
- **LDTSTOP\_L.** After DCSTOP\_L or SUSPEND\_L or PCISTOP\_L (if enabled) is asserted, the IC waits the time interval dictated by DevB:3x74[C3S1LST] then, if DevB:3x70[POSLS]=1b, the IC asserts LDTSTOP\_L.

### 3.7.1.6.8 Transitions From S1 (POS) to S0 (FON)

The following is the S1 resume sequence, once an enabled resume event occurs:

- **PCISTOP\_L, CPUSLEEP\_L, CPUSTOP\_L.** If these signals are asserted, then they are deasserted immediately after an enabled resume event is detected.
- **LDTSTOP\_L.** LDTSTOP\_L is deasserted immediately after an enabled resume event is detected. See DevB:3x74[FVLST].
- **DCSTOP\_L, SUSPEND\_L.** The IC waits a programmable amount of time dictated by DevB:3x54[PLLCNT1] after LDTSTOP\_L is deasserted and then deasserts DCSTOP\_L and SUSPEND\_L. If LDTSTOP\_L is not asserted during S1 then the delay is from the detected resume event.
- **AGPSTOP\_L.** The IC waits a programmable amount of time dictated by DevB:3x54[PLLCNT] after DCSTOP\_L or SUSPEND\_L or LDTSTOP\_L deassertion (if DCSTOP\_L or SUSPEND\_L is not used) and then deasserts AGPSTOP\_L.
- After the IC has deasserted AGPSTOP\_L, it issues a HyperTransport STPCLK system management message with the STPCLK bit deasserted.



### 3.7.2 Serial IRQ Protocol

The IC supports the serial IRQ protocol. This logic controls the SERIRQ pin and outputs IRQs to the PIC and IOAPIC blocks. This logic runs off of PCLK. It is specified by DevB:3x4A. The serial IRQ logic does not provide support for generating IRQ0, IRQ2, IRQ8, or IRQ13. In order to use IRQ[15, 14, 12, 6, or 1], the corresponding external IRQ pin must be pulled High. The PCI IRQs from the serial IRQ logic are routed to the interrupts specified by DevB:3x56.

### 3.7.3 SMBus 1.0

The IC includes a system management bus 1.0, or SMBus 1.0, interface. SMBus is a two-wire serial interface typically used to communicate with system devices such as temperature sensors, clock chips, and batteries. The control registers for this bus are PME0 through PMEF.

The SMBus interface includes a host controller and a host-as-slave controller.

**Host controller.** The host controller is used to generate cycles over the SMBus as a master. Software accomplishes this by setting up PME2[CYCTYPE] to specify the type of SMBus cycle desired and then (or concurrently) writing a 1 to PME2[HOSTST]. This triggers an SMBus cycle with the address, command, and data fields as specified by the registers called out in PME2[CYCTYPE].

Writes to the host controller registers PME2[3:0], PME4, PME8, and PME9 are illegal while the host is busy with a cycle. If a write occurs to PME2 while PME0[HST\_BSY] is active, then the four LSBs are ignored. Writes to PME4, PME8, and PME9 while PME0[HST\_BSY] is active are ignored (the transaction is completed, but no data is transferred to the SMBus controller).

If an SMBus-defined time out occurs while the host is master of the SMBus, then the host logic attempts to generate a SMBus stop event to clear the cycle and PME0[TO\_STS] is set.

The host controller is only available in the FON state.

**Host-as-slave controller.** The host-as-slave controller responds to word-write accesses to either the host address specified by PMEE or the snoop address specified by PMEF. In either case, if the address matches, then the subsequent data is placed in PMEC and PMEAS. In the case of snoop accesses, the command information is stored in PMEC[7:0] and the data is stored in PMEAS[15:0]. In the case of addresses that match the PMEE host-as-slave address register, then the address is stored in PMEC[7:1]—if the transaction includes a 7-bit address—or PMEC[15:1]—if the transaction includes a 10-bit address. After the address match is detected, the logic waits for the subsequent stop command before setting the appropriate status bits in PME0[HSLV\_STS, SNP\_STS]; however, if a time out occurs during the cycle, after the address match is detected, then the appropriate bits in PME0[HSLV\_STS, SNP\_STS] are set.

If one of the slave status bits, PME0[HSLV\_STS, SNP\_STS], is set and another access to the host slave controller is initiated, then it is not acknowledged by the first SMBus acknowledge cycle until the status bit is cleared.

The host-as-slave controller operates in all system power states except MOFF; it can be used to generate interrupts and resume events.

**SMBALERT.** The host controller includes support for the SMBALERT\_L signal. If this signal is asserted, then it is expected that software determines the source by generating a host read cycle to the alert response address, 0001100b. If the SMBus host controller detects this address for a read cycle with PME2[CYCTYPE] set to receive byte (001b), then it stores the address returned by the SMBALERT\_L slave in PME6[7:0]. If bits[7:1] of this address are 1111\_0xxb, indicating a 10-bit address, then it stores the next byte from the slave in PME6[15:8].

### 3.7.4 Plug And Play IRQs

The IC supports three PNP IRQs. The register that specifies these is DevB:3x44. The PNPIRQ[2:0] pins are multiplexed with GPIO functions; the control registers that specify the functions (PMD3, PMD4, and PMD5) must be set up appropriately for the PNP functions to operate.

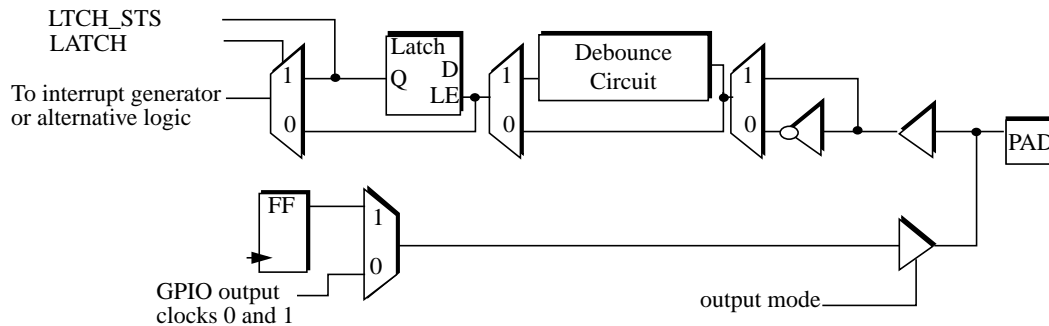
### 3.7.5 General Purpose I/O

The general-purpose I/O pins, GPIO[31:0], can be assigned to be inputs, outputs, interrupt generators, or bus controls. These pins can be programmed to be general-purpose I/O or to serve alternate functions; see the PM[DF:C0] register definitions. Most of these pins are named after their alternate functions. There is one control register for each pin, PM[DF:C0]. IRQ status and enables are available for each pin in registers PMB0 and PMB4; SMI enables are available in PMB8.

**General-purpose I/O functions.** When programmed as a GPIO pin, the following functions are available:

- Outputs
  - Can be set High or Low.
  - Can be controlled by GPIO output clocks 0 or 1 (see PMBC).
- Inputs
  - Active High or active Low programmable.
  - SCI or SMI IRQ capable.
  - Can be latched or not latched.
  - Inputs can be debounce protected.

The following diagram shows the format for all GPIO pins. The input path is not disabled when the output path is enabled or when the pin is used for an alternate function.



**Figure 12. GPIO Pin Format**

**Debounce.** The input signal must be active and stable for 12 to 16 ms before the output signal will be asserted.

**GPIO output clocks.** There are two GPIO output clocks numbered 0 and 1. Their behavior is specified by PMBC. Each output clock includes a 7-bit programmable High time, a 7-bit programmable Low time, and the counter can be clocked by one of four frequencies. Here are the options:

**Table 22. GPIO Output Clock Options**

PMBC[CLK[1,0]BASE]	Base Clock period	Output High Time Range	Output Low Time Range
00b	250 $\mu$ s	250 $\mu$ s to 32 ms	250 $\mu$ s to 32 ms
01b	2 ms	2 ms to 256 ms	2 ms to 256 ms
10b	16 ms	16 ms to 2 s	16 ms to 2 s
11b	128 ms	128 ms to 16.4 s	128 ms to 16.4 s

The output of the two GPIO output clocks can be selected to drive the output of any of the GPIO pins. They may be used to blink LEDs or for other functions.

## 3.8 AC '97 Controller

### 3.8.1 Introduction

The AC '97 host controller implementation supports the following features:

- Independent PCI functions for audio (function 5, address space DevB:5xXX) and modem (function 6, address space DevB:6xXX)
- Independent channels for PCM In, PCM Out, Microphone In, Modem In, Modem Out

- 2-channel stereo for PCM Audio In, 2-, 4-, 6-channel stereo for PCM Audio Out
- 16 bit sample resolution
- Multiple sampling rates
- Up to two codecs

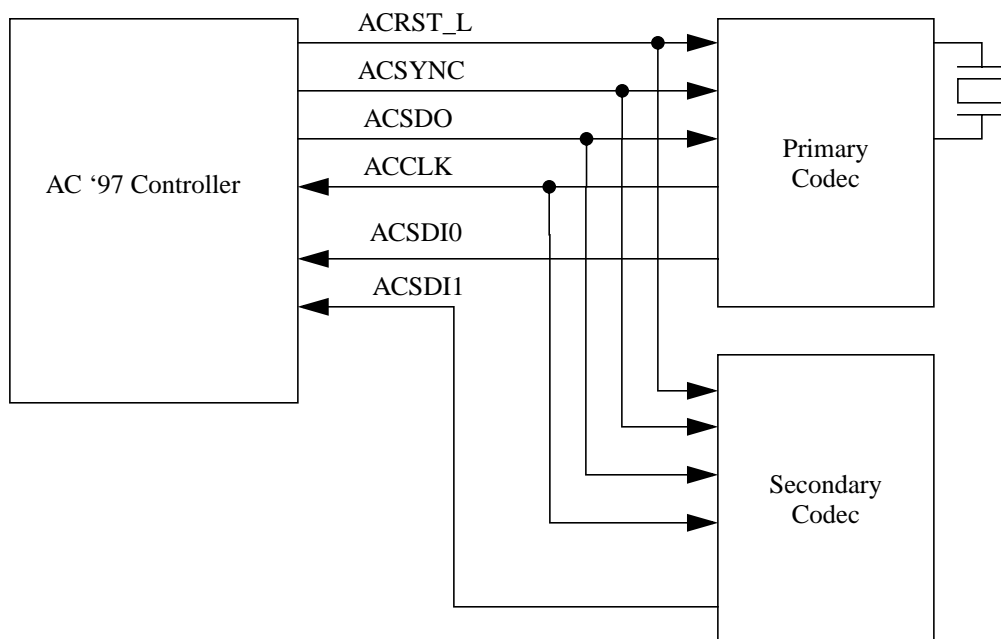
An AC '97 sub-system includes a digital controller and a set of up to two codecs referred to as audio codec (AC) and modem codec (MC). The codecs are in one or two physically separate chips, and are connected to the digital controller through an AC-link serial interface.

The AC '97 controller does not distinguish between primary and secondary codec. Since registers do distinguish between ACSDI0 and ACSDI1 for reporting wake events, codec status etc. the AC '97 controller documentation assumes the primary codec being attached to ACSDI0 and the secondary codec being attached to ACSDI1.

Supported codec configurations are:

- AC as primary
- MC as primary
- AC as primary and MC as secondary
- AC as primary and AC as secondary
- AMC as primary

The IC does not support optional test modes as outlined in the AC '97 specification.



**Figure 13. AC '97 Codec Connections**

See the AC '97 specification for further information.

### 3.8.2 AC '97 Serial Link Interface

The AC '97 serial link interface is designed to be AC '97 revision 2.2 compliant. For a detailed description of the AC-link interface see the AC '97 specification.

The IC supports up to two ACSDI signals for use with a primary and secondary codec. Depending on which codec (AC, MC, AMC) is attached, various input slots might be valid or invalid. With the exception of the input tag slot 0 those input slots must be completely orthogonal, i.e., no two data slots at the same location are valid on both input signals. This precludes the use of two similar codecs (e.g., two ACs or MCs) that use the same data slots.

The codec ready bit 15 of input slot 0 indicates whether the codec on the AC-link is ready for normal operation. The codec ready bits from the input slots 0 at ACSDI0 and ACSDI1 are visible through the Global Status controller register. Software must further probe the Powerdown Control/Status register in the codec to determine exactly which subsections, if any, are ready.

The output slot 1 provides a command port to control features and monitor status of an AC '97 codec. The control interface architecture supports read/write accesses to a maximum of 64 16 bit codec registers, addressable on even byte boundaries. Only the even register addresses are valid.

The input slot 1 tag bit in input slot 0 only pertains to the Control Register Index data from a previous read. Slot request bits are always valid and thus have to be checked independent of the slot 1 tag bit.

The IC does implement transmission of GPIO values to the codec in output slot 12. The values of the bits in this slot are the values written to the GPIO Pin Status codec register at 54h/D4h. The following rules govern the usage of slot 12.

- Slot 12 is marked invalid by default on coming out of reset and remains invalid until a GPIO Pin Status codec register write.
- A GPIO Pin Status codec register write causes the write data to be transmitted in slot 12 in the next possible frame, with slot 12 marked valid, and the address/data information to be transmitted in slots 1 and 2 of the same frame.
- After the first GPIO Pin Status codec register write, slot 12 remains valid for all following frames. The data transmitted in slot 12 is the data last written to the GPIO Pin Status codec register. Any subsequent write to the register causes the new data to be sent out in the next frame.
- Slot 12 gets invalidated after the following events: SM reset, AC '97 cold reset, warm reset, and hence a wake-up from S3/S4/S5. Slot 12 remains invalid until the next GPIO Pin Status codec register write.

The content of the GPIO Pin Status codec register is to be returned in slot 12 of every input frame. Reads from GPIO Pin Status codec register at 54h/D4h are not transmitted across the link in slots 1 and 2. The data from the most recent slot 12 is stored in a controller shadow register and is returned. That data is also accessible in the MC48 AC '97 controller register.

When accessing the codec registers, only one I/O cycle can be pending across the AC-link at any time. The IC internal architecture and AC '97 controller implementation provides arbitration logic to ensure that. For compatibility reasons the CAS bit in the Codec Access Semaphore register is provided. Software is to monitor the CAS bit that indicates that a codec access is pending. The CAS bit is set by reads from software and reset upon completion of the codec register access cycle. Once the CAS bit is cleared, another codec access can be initiated. The exception to this being reads to the GPIO Pin Status codec register that are returned immediately with the most recently received input slot 12 data shadowed in a controller internal register. A read access to the GPIO Pin Status register does not reset the CAS bit upon completion while a write access to the GPIO Pin Status register does reset the CAS bit upon completion.

The controller does not issue back to back reads. It must get a response to the first read before issuing a second. In addition, codec reads and writes are only executed once across the link and are not repeated.

### 3.8.3 AC '97 PCI Interface and Bus Master Controller

The AC '97 controller is a PCI bus master with scatter/gather support. The PCI interface has the following characteristics:

- On reads from a codec, the controller expects the codec to respond within the next frame, after which, if no response is received, it returns a dummy read completion to the processor (with FFh on the data) and also sets the Read Completion Status bit in the Global Status Register. If ACCLK is not operating adequately the same responses occurs immediately (with FFh on the data, and AC30/MC40[RCSTAT] set). Codec register reads by the host might cause a PCI retry cycle.
- On writes to a codec, the controller returns a write completion to the processor when the write data is inserted in a frame to the codec. If ACCLK is not operating adequately the same response occurs immediately.
- FIFO buffers are filled and emptied in one PCI transaction using double-word transfers where possible, and word transfers where necessary. Data sets can be word-aligned. When host memory buffer page boundaries are crossed (i.e., need to switch to the next buffer page) more than one PCI transactions are used.
- Audio and modem interrupts generated by the AC '97 controller are connected to one PCI interrupt request. Table 23 lists the PCI interrupt sources.

**Table 23. PCI Interrupt Sources**

Interrupt Register Bit	Enable Register Bit	Interrupt
ACx6/MCx6[BCIS]	ACxB/MCxB[IOCE]	Scatter/gather host memory Buffer Completion Interrupt Status (i.e., buffer full/empty)
ACx6/MCx6[FIFOERR]	ACxB/MCxB[FEIE]	FIFO over-run (modem in buffer) or under-run (modem out buffer) error
ACx6/MCx6[LVBCI]	ACxB/MCxB[LVBCIEN]	Last Valid Buffer Completion Interrupt

**Table 23. PCI Interrupt Sources (Continued)**

Interrupt Register Bit	Enable Register Bit	Interrupt
AC30/MC40[SRINT]	AC2C/MC3C[SRIEN]	Secondary Resume Interrupt
AC30/MC40[PRINT]	AC2C/MC3C[PRIEN]	Primary Resume Interrupt
AC30/MC40[GPIINT]	AC2C/MC3C[GPIEN]	GPI Interrupt

The principal function of scatter/gather (i.e., basically a memory paging mechanism) is to assist the host operating system in managing memory fragmentation. One logical buffer is split among multiple physical blocks or pages of host memory. A logical buffer is mapped to physical host memory pages using a descriptor table. The descriptor table is pointed to by the Buffer Descriptor List Base Address register. Each descriptor table entry contains the physical memory address of a host memory page, the length of the page, and other information. Descriptor tables have 32 entries, and are set up by a software driver. Host memory page sizes of up to 64K samples are supported. The descriptor format is shown in Table 24.

**Table 24. Descriptor Format**

Bit	Description
63	<b>IOC. Interrupt on Completion.</b> When set, indicates that an interrupt should be generated upon completion of data transfer to/from the block/buffer.
62	<b>BUP. Buffer Underrun Policy.</b> When set to 1b the controller transmits zeros in the case that this buffer is completed and the last valid buffer has been processed. Otherwise the controller transmits the last valid sample. This bit typically is set only if this is the last buffer in the current stream.
61:48	Reserved
47:32	<b>Length.</b> Length of host physical memory block/buffer in words (i.e., number of 16 bit locations).
31:1	<b>BASE_ADDR.</b> Base address of host physical memory block/buffer.
0	Reserved

The current descriptor in use is referenced by the Current Index Value CIV. This value basically follows the Last Valid Index LVI by being incremented after filling the associated host memory buffer. The value rolls over from index 31 to index 0. So, to allow a roll-over, the descriptor table has to be set up with 32 valid entries. No roll-over from an index less than 31 can be negotiated. Other than that the CIV value can only be set to index 0 by applying reset to the AC '97 controller.

After reset, the first descriptor at index 0 is prefetched by PIV. After handing over the prefetched descriptor index to CIV, the next descriptor in the descriptor table is prefetched at an index incremented by one (with rolling over after index 31). Also, in the case when CIV equals LVI, the next descriptor is prefetched, but only processed after LVI has been incremented by software, thus providing that descriptor to the controller.

This bus master logic, once set up by a software driver, automatically fetches descriptors, transfers data to and from host memory, generates interrupts, etc. as part of its implemented control mechanism. This scatter/gather control mechanism is implemented for each data stream.

### 3.8.4 AC '97 Data Buffering

Data buffering inside the digital controller is provided with FIFO buffers. The following buffers are provided as separate, independent buffers:

- Audio PCM left out
- Audio PCM right out
- Audio PCM center front out
- Audio PCM left rear out
- Audio PCM right rear out
- Audio subwoofer out
- Audio PCM left in
- Audio PCM right in
- Audio microphone in
- Modem in
- Modem out

Note that the Audio PCM out streams are considered one data channel and have one associated scatter/gather bus master controller (i.e., the PCM out channel). The same is true for Audio PCM left in and Audio PCM right in (i.e., the PCM in channel). The samples are transferred in the following order:

**Table 25. PCM Audio Sample Order**

Audio Channel	AC '97 Timeslot	2-Channel Sample Order	4-Channel Sample Order	6-Channel Sample Order
Left-Front	3	1	1	1
Right-Front	4	2	2	2
Center-Front	6			3
Left-Rear	7		3	5
Right-Rear	8		4	6
Subwoofer	9			4

For a given 2-, 4-, or 6-channel audio stream the audio bus master controller expects each sample compound to start with the left-front sample at the least significant word address proceeded by the following samples at incremented word addresses. For a 4-channel audio stream that leads to the following sample address scheme:

```
base_addr+0h left-front sample 0
base_addr+2h right-front sample 0
base_addr+4h left-rear sample 0
base_addr+6h right-rear sample 0
base_addr+8h left-front sample 1
```



```
base_addr+Ah right-front sample 1
...
```

The sample pairs Left-Front/Right-Front, Center-Front/Subwoofer and Left-Rear/Right-Rear can be swapped by programming DevB:5x4C.

Each FIFO has the following characteristics:

- Contains eight 16 bit samples.
- Either one 16 bit word (i.e., one sample at a time) or one 32 bit doubleword (i.e., two samples at a time) can be read or written each clock using the PCI bus.
- Transfers to/from the AC-link interface controller are done one sample a frame. The 16 bit samples are transferred as the 16 most significant bits of each 20 bit slot, with the low order bits discarded for input data and padded with zeros for output data.
- For output buffers (to codec), filling is initiated by completion of passing the one-fourth empty threshold (i.e., 2 entries are empty, 6 are still full).
- For input buffers (from codec), emptying is initiated by completion of passing the one-fourth full threshold (i.e., 2 entries are full, 6 are still empty).

For a given 2-, 4-, or 6-channel audio configuration two principal transfer modes are supported:

- Full Rate Transfer Mode
- Half Rate Transfer Mode

In full rate transfer mode all samples for the given audio configuration are transferred in one frame. In half rate transfer mode the left front, center front, left rear samples are transferred in one frame and the right front, right rear, subwoofer samples are transferred in the following frame. See Figure 14 and Figure 15 on page 74 for examples of half rate transfer mode transmissions. When mono audio sample streams are transferred, software and codec must ensure both left and right samples are transferring the same data, i.e., for each mono sample two samples are transferred.

The transfer modes are controlled for audio-in samples by the respective valid bits in slot 0 of the input frame and for audio-out samples by the respective request bits in slot 1 of the input frame. For audio-out samples a slot 3 request bit set to 0b causes the controller to read all audio samples for a given audio configuration from the audio-out FIFOs for subsequent transmission as enabled by the request bits. For audio-in samples a slot 4 valid bit set to 1b causes the controller to write received audio samples into the audio-in FIFOs. The codec has to send the appropriate valid and request bits in the correct order as required by the transfer modes. Repeating valid or request bits in subsequent frames out of order causes loss of data.

For the example in Figure 14 the slot 3, 6, and 7 request bits were set to 0b in Input Frame n-1 with the slot 3 request bit causing the controller to read audio samples from all audio-out FIFOs.

According to those request bits, slot 3, 6, and 7 samples are transferred in Output Frame n. The slot 4, 8, and 9 request bits active (Low) in Input Frame n cause the controller to send the already read slot 4, 8, and 9 samples in Output Frame n+1. The slot 3 request bit active (Low) in Input Frame n+1 causes the controller to read new audio samples from all audio-out FIFOs.

Output slots which are not in use for a given configuration are always stuffed with zeros by the IC.

Slot Number	0	1	2	3	4	5	6	7	8	9	10	11	12
Frame n	TAG	CMD ADDR	CMD DATA	x			x	x					GPIO
Frame n+1	TAG	CMD ADDR	CMD DATA		x				x	x			GPIO
Frame n+2	TAG	CMD ADDR	CMD DATA	x			x	x					GPIO

**Figure 14. Output Frame with Audio Samples in Half Rate Transfer Mode**

Slot Number	0	1	2	3	4	5	6	7	8	9	10	11	12
Frame n	TAG	CMD ADDR	CMD DATA	x									GPIO
Frame n+1	TAG	CMD ADDR	CMD DATA		x								GPIO
Frame n+2	TAG	CMD ADDR	CMD DATA	x									GPIO

**Figure 15. Input Frame with Audio Samples in Half Rate Transfer Mode**

### 3.8.5 AC '97 Power Management Logic

See the AC '97 specification for a definition of the wake-up event protocol.

AC30/MC40[PRINT] or AC30/MC40[SRINT] is set when a wake-up event occurred on the AC-link and the AC '97 controller is not in sleep state. AC2C/MC3C[PRIEN] and AC2C/MC3C[SRIEN] enable an AC-link wake-up event to initiate the respective AC '97 controller interrupt.

PM20[AC97\_STS] is set when a wake-up event occurred on the AC-link and the AC '97 controller is in sleep state. PM22[AC97\_EN] enables an AC-link wake-up event to initiate a SCI or SMI for the system to wake up. PM2A[AC97\_EN] enables an AC-link wake-up event to initiate a SMI for the system to wake up.

AC2C/MC3C[SHUTOFF] can be used to disable the AC-link I/O signals. When disabled, all outputs, inclusive ACRST\_L, are forced Low and all inputs are ignored, except for wake-up event detection.

Once the codec has been instructed to halt ACCLK, a special wake-up protocol must be used to bring the AC-link to the active mode. Three methods for waking up the AC-link are supported:

- external wake-up
- warm reset
- cold reset

A transition from Low to High at ACSDI0 or ACSDI1 causes the AC '97 controller to sequence through a wake-up event detection and to finally report this wake-up event in PM20[AC97\_STS], AC2C/MC3C[PRINT] or AC2C/MC3C[SRINT], respectively, depending on the AC '97 controllers's system state and on what ACSDI[1:0] signal the event occurred. If enabled, a corresponding interrupt is issued. Software then has to issue a warm or cold reset to the codec by setting the appropriate bit in the Global Control register to finish the wake-up sequence.

A cold reset drives ACRST\_L Low for a minimum of 1  $\mu$ s. It initializes all codec registers to their default power on reset values. Internal AC-link control registers and FIFOs in the AC '97 are initialized as well. The bus master control registers of the AC '97 are not affected.

A warm reset re-activates the AC-link without altering the current codec register values. It is signaled by driving ACSYNC High for a minimum of 1  $\mu$ s in the absence of ACCLK. The AC '97 controller is not initialized but potentially remaining slot requests from the last frame before power-down are discarded.

Once powered down, activation of the AC-link by re-assertion of the ACSYNC signal must not occur for a minimum of 4 frame times following the frame in which the power down was triggered. The AC '97 controller samples ACSDI[1:0] but delay any wake-up event reporting after the above time frame has expired.

When the AC-link powers up it indicates readiness by way of the codec ready bits in the Global Status controller register. Software needs to check those bits before accessing the codec.

### 3.8.6 AC '97 Link Reset

ACRST\_L is asserted by the IC under the following conditions:

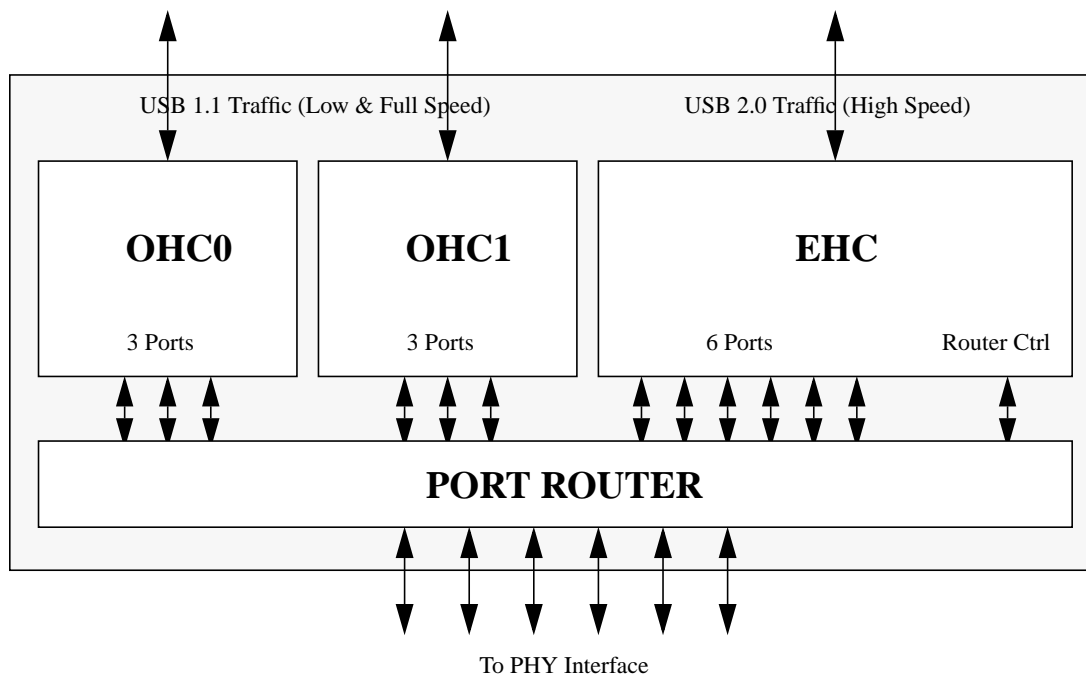
- RESET\_L asserted
- AC2C/MC3C[CRST\_L] asserted

The IC never deasserts ACRST\_L automatically since after reset AC2C/MC3C[CRST\_L] defaults to zero. Software has to deassert this bit. While AC2C/MC3C[CRST\_L] resides in the VDD\_COREX power plane, it remains cleared upon return from S3/S4/S5.

## 3.9 USB Controller

The USB 2.0 host controller is designed to comply with the EHCI (Enhanced Host Controller Interface) specification. It consists of three main components (see Figure 16):

- The **Enhanced Host Controller (EHC)**: This block handles the USB 2.0 high speed traffic. Additionally, it controls the Port Router.
- Two **Companion Host Controllers**. These are two OHCI (Open Host Controller Interface) compliant host controllers (OHC0 & OHC1). These handle all USB1.1 compliant traffic and contain the legacy keyboard emulation (for non-USB aware environments).
- **The Port Router**. This block assigns the physical port interfaces to their respective owners. This ownership is controlled by EHC registers. Per default, all ports are routed to the OHCx, in order to allow for a system with only USB 1.1 aware drivers to function. If a USB 2.0 aware driver is present in the system it will assign the ports to either an OHCx for low and full speed devices and hubs (USB 1.1 traffic) or to the EHC for high speed devices and hubs (USB 1.1 traffic) or to the EHC for high speed devices and hubs.



**Figure 16. USB Controller Building Blocks**

### 3.9.1 USB Interrupts

All USB controllers drive the PIRQD\_L interrupt signal. However, the USB interrupt can be diverted to SMI by the OHCI-defined register HcControl\_InterruptRouting. See Section 3.4.2.1 on page 41 for data on routing of keyboard and mouse emulation interrupts. SMI interrupts are also generated in response to accesses to I/O ports 60h and 64h and to IRQ1 and IRQ12 in support of the emulation logic.

## 3.10 LAN Ethernet Controller

### 3.10.1 Interfaces

#### 3.10.1.1 Software Interface

The software interface to the network controller is divided into three parts. One part is the PCI configuration registers used to identify the network controller and to setup the configuration of the device. The setup information includes the memory mapped I/O base address and the routing of the network controller interrupt channel. This allows for a jumperless implementation.

The second portion of the software interface is the direct access to the I/O resources of the network controller. The host CPU accesses these registers for performance tuning, selecting options, statistics collecting, and starting transmissions.

The third portion of the software interface is the descriptor and buffer areas that are shared between the software and the network controller during normal network operations. The descriptor area boundaries are set by the software and do not change during normal network operations. There is a separate descriptor area for each receive and transmit priority queue. The descriptor space contains relocatable pointers to the network frame data, and it is used to transfer frame status from the network controller to the software. The buffer areas are locations that hold frame data for transmission or that accept frame data that has been received.

#### 3.10.1.2 Network Interface

The network controller can be connected to an IEEE 802.3 or proprietary network through the IEEE 802.3-compliant Media Independent Interface (MII). The MII is a nibble-wide interface to an external 100-Mbit/s and/or 10-Mbit/s transceiver device.

The network controller supports both half-duplex and full-duplex operation on the network interface.

### 3.10.2 Device Operation

#### 3.10.2.1 Initialization

Before the network controller is ready for operation, several registers must be initialized. First certain read-only fields in PCI configuration space must be initialized by writing to alias registers in PCI configuration space. These fields contain values such as Subsystem Vendor ID and Maximum Latency, which depend on the board in which the device is installed and should not be changed by operating system software. Once these hardware parameters have been set up, the normal BIOS initialization software can write to the writable fields in PCI configuration space. In particular, the Memory-Mapped I/O Base Address Register in configuration space must be initialized before any of the memory-mapped registers can be accessed.

Before any network frames can be sent or received, the unique 48-bit IEEE MAC address must be written to the Physical Address Register (PADR). Normally this is done by a chipset initialization routine that runs before the network device driver is loaded.

Finally after the Memory-Mapped I/O Base Address Register has been initialized and the operating system has been loaded, the device driver software can write to various memory-mapped registers to set up software parameters such as pointers to descriptor rings and the Logical Address Filter contents.

### 3.10.2.2 Re-Initialization

Earlier members of the PCnet™ family of controllers had to be re-initialized if the transmitter and/or the receiver were not turned on during the original initialization, and it was subsequently required to activate them, or if either section was shut off due to the detection of a memory error, transmitter underflow, or transmit buffer error condition. This restriction does not apply to this controller. The memory error and transmit buffer error conditions cannot occur in the controller and the transmit underflow condition does not stop the controller's transmitter.

### 3.10.2.3 Run and Suspend

Following reset, the transmitter and receiver of the controller are disabled, so no descriptor or data DMA activity can occur. The receiver does process incoming frames to detect address matches, which are counted in the RcvMissPkts register. No transmits occur except that pause frames may be sent (see flow control section).

Setting the RUN bit in CMD0 causes the controller to begin descriptor polling and normal transmit and receive activity. Clearing the RUN bit causes the controller to halt all transmit, receive, and DMA transfer activities abruptly.

The controller offers suspend modes that allow stopping the device with orderly termination of all network activity. Transmit and receive are controlled separately.

Setting the RX\_FAST\_SPND bit in CMD0 suspends receiver activity after the current frame being received by the MAC is complete. If no frame is being received when RX\_FAST\_SPND is set, the receiver is suspended immediately. After the receiver is suspended, the RX\_SUSPENDED bit in STAT0 is set and SPNDINT interrupt bit in INT0 is set.

Setting the RX\_SPND bit in CMD0 suspends the receiver in the same way as RX\_FAST\_SPND, but the RX\_SUSPENDED bit and SPNDINT interrupt bit are only set after any frames in the receive FIFO have been completely transferred into system memory and the corresponding descriptors updated. No receive data or descriptor DMA activity occurs while the receiver is suspended.

When the receiver is suspended, no frames are received into the receive FIFO, but frames are checked for address match and the RcvMissPkts counter incremented appropriately, and frames are checked for Magic Packet™ frame match if Magic Packet technology mode is enabled.

Setting the TX\_FAST\_SPND bit in CMD0 suspends transmitter activity after the current frame being transmitted by the MAC is complete. If no frame is being transmitted when TX\_FAST\_SPND is set,

the transmitter is suspended immediately. After the transmitter is suspended, the TX\_SUSPENDED bit in STAT0 is set and SPNDINT interrupt bit in INT0 is set.

Setting the TX\_SPND bit in CMD0 suspends the transmitter in the same way as TX\_FAST\_SPND, but the TX\_SUSPENDED bit and SPNDINT interrupt bit are only set after any frames in the transmit FIFO have been completely transmitted. No transmit descriptor or data DMA activity occurs while the transmitter is suspended.

When the transmitter is suspended, no frames are transmitted except for flow control frames (see Flow Control section).

It is not meaningful to set both TX\_SPND and TX\_FAST\_SPND at the same time, nor is it meaningful to set both RX\_SPND and RX\_FAST\_SPND at the same time. Doing so causes unpredictable results. However, transmit and receive are independent of each other, so one can be suspended or fast suspended while the other is running, suspended, or fast suspended.

It is recommended when software polls this register that a delay be inserted between polls. Continuous polling reduces the bus bandwidth available to the controller and delays completion of the suspend operation.

It is recommended that software use the SPNDINT interrupt to determine when the controller has suspended after one or more suspend bits have been set. This results in the least competition for the PCI bus and thus the shortest time from setting of a suspend bit until completion of the suspend operation.

The suspend bits can be used either to stop the transmitter or receiver while the controller is running or to prevent the transmitter or receiver from operating when the controller starts running. Clearing the RUN bit in CMD0 generates a pulse that clears all the suspend command and status bits (TX\_SPND, RX\_SPND, TX\_FAST\_SPND and RX\_FAST\_SPND in CMD0, TX\_SUSPENDED, and RX\_SUSPENDED in STAT0 and SPNDINT in INT0). To restart the controller with the transmitter disabled, set RUN and TX\_SPND. To restart the controller with the receiver disabled, set RUN, RX\_SPND, and RDMD0. Since the suspend bit is cleared when RUN is cleared, the appropriate suspend bit must be set each time RUN is set. Since the suspend bits and RUN are in the same register (CMD0), the suspend bit can be set at the same time that RUN is set.

#### 3.10.2.4 Descriptor Management

The network controller contains its own DMA controller that automatically transfers network frame data between the network controller and buffers in host system memory. The actions of the DMA controller are controlled by data structures in system memory called descriptors. Each descriptor contains a pointer to a buffer in system memory plus several control and status fields. The Descriptor Management Unit automatically reads control information from descriptors to determine how to manage the data transfers and writes back information about the status of the transfers.

Descriptor management is accomplished through message descriptor entries organized as ring structures in memory. There are five descriptor rings, four for transmit and one for receive. The implementation of four descriptor rings for transmit allows the controller to provide improved support for quality of service. Each descriptor ring is allocated to serve a certain class of traffic, with a clearly

defined priority scheme. The manner in which this scheme relates to standards such as IEEE 802.1 P is described elsewhere in this document. This section simply describes the implementation of the descriptor rings themselves.

Each descriptor describes a single buffer. A frame may occupy one or more buffers. If multiple buffers are used, this is referred to as buffer chaining.

Transmit buffers can be of any size and can start at any byte address. Receive buffers can start at any byte address, but there is a restriction on receive buffer length. Either the length of a receive buffer must be a multiple of 4 bytes, or the length of the receive buffer must be at least as large as the longest frame that the software can accept.

Each descriptor ring must occupy a contiguous area of memory. The user-defined base addresses for the transmit and receive descriptor rings are set up during initialization in the BADR0 and BADX0-3 registers. The number of descriptors in each ring is written into the RCV\_RING0\_LEN and XMT\_RINGx\_LEN registers, where x indicates the transmit ring number.

The descriptor ring base addresses must be aligned to 16-byte boundaries. Each ring entry is organized as four 32-bit message descriptors. Ring lengths can be of any size up to 65535 descriptors.

Each ring entry contains the following information:

- The address of the actual message data buffer in user or host memory
- The length of the message buffer
- Status information indicating the condition of the buffer

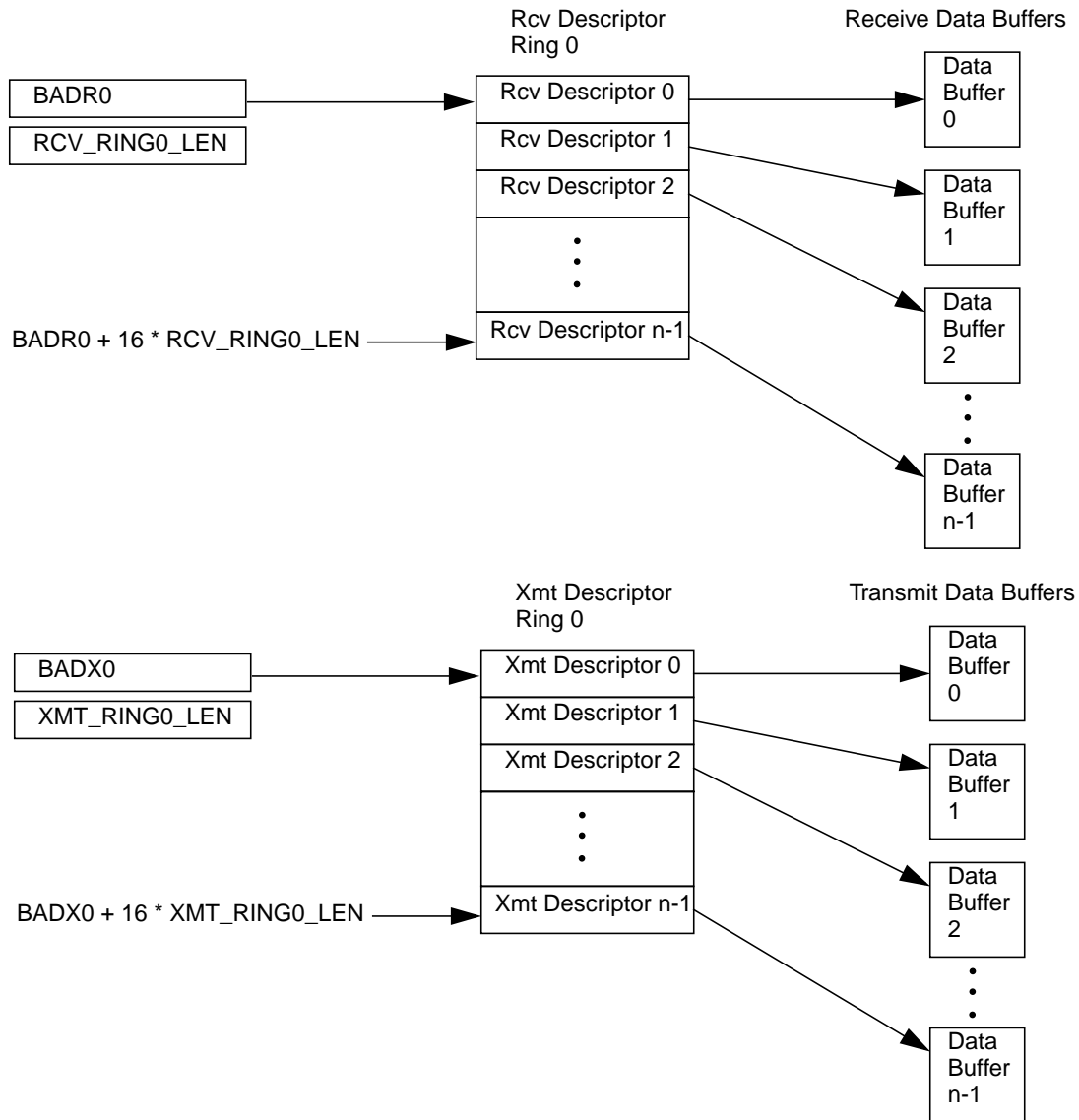
See Figure 32 on page 347 and Figure 34 on page 349 for a detailed description of descriptor formats.

To permit the queuing and de-queuing of message buffers, ownership of each buffer is allocated to either the network controller or the host. The OWN bit within the descriptor status information is used for this purpose.

Setting the OWN to 1 signifies that the network controller currently has ownership of this descriptor and its associated buffer. Only the owner is permitted to relinquish ownership or to write to any field in the descriptor entry. A device that is not the current owner of a descriptor cannot assume ownership or change any field in the entry. A device may, however, read from a descriptor that it does not currently own. Software should always read descriptor entries in sequential order. When software finds that the current descriptor is owned by the network controller, the software must not read ahead to the next descriptor. The software should wait at a descriptor it does not own until the network controller sets OWN to 0 to release ownership to the software.

Figure 17 on page 81 illustrates the relationship between the receive and transmit descriptor ring base addresses, the receive and transmit descriptors, and the receive and transmit data buffers for one receive and one transmit descriptor ring.





**Figure 17. Receive and Transmit Descriptor Rings**

**3.10.2.5 Polling**

Polling is the action that the Descriptor Management Unit (DMU) takes when it reads a descriptor to examine the descriptor’s OWN bit. Polling is initiated either by a command from the host CPU or automatically as part of the frame processing sequence.

When the DMU reads a descriptor from system memory, it reads more than one descriptor in one burst, so that polling can often be done by examining an internal descriptor cache. However, when the

DMU finds that the OWN bit of the next descriptor in the internal descriptor cache is 0, it must access the host system memory again to read in another block of descriptors.

### 3.10.2.6 Transmit Polling

After the host CPU has filled one or more buffers with data to be transmitted and has set the OWN bits of the descriptors that correspond to these buffers, it starts the transmit process by setting the Transmit Demand (TDMDx) bit in CMD0 that corresponds to the appropriate descriptor ring. (The “x” in TDMDx is the number of the descriptor ring.) Setting TDMDx causes the DMU to poll transmit descriptor ring number x if it does not already own a descriptor in that ring.

Responding to the TDMDx bit, the DMU begins the process of reading descriptors, checking OWN bits, retrieving buffer addresses, and copying frame data from transmit buffers to the transmit FIFO. After the controller has finished copying the data from the buffer, it writes to system memory to clear the descriptor’s OWN bit. It then polls the next descriptor in the ring. When the DMU encounters a descriptor whose OWN bit is not set, it stops polling and waits for the host CPU to set a TDMDx bit again.

If the DMU encounters a descriptor whose OWN bit is set, but whose Start of Packet (STP) bit is not set, the controller immediately requests the bus in order to clear the OWN bit of this descriptor. After resetting the OWN bit of this descriptor, the controller immediately polls the next descriptor in the ring.

Similarly, when the DMU encounters a descriptor whose OWN bit is set, but whose byte count field is 0, it also clears the descriptor’s OWN bit and then polls the next descriptor in the ring.

The network controller returns ownership of transmit descriptors to the software when the DMA transfer of data from system memory to the controller's internal FIFO is complete. This is different from older devices in the PCnet family, which do not return the last transmit descriptor of a frame (the one with ENP=1) until transmission of the frame is complete. This controller does not return any status information in the transmit descriptor, it only writes to the OWN bit to clear it.

If underflow occurs due to delays in setting the OWN bits or excessive bus latency, the transmitter appends an inverted FCS field to the frame and increments the XmtUnderrunPkts counter. The frame may be retransmitted (if the REX\_UFLO bit in CMD2 is set) or discarded.

If an error occurs in the transmission that causes the frame to be discarded (late collision, underflow or retry failure with the corresponding retry or retransmit option not enabled) before the entire frame has been transferred or if the current transmit descriptor has its KILL bit set, and if current transmit descriptor does not have its ENP bit set, the controller skips over the rest of the frame that experienced the error. The controller clears the OWN bit for all descriptors with OWN = 1 and STP = 0 and continues in like manner until a descriptor with OWN = 0 (no more transmit frames in the ring) or OWN = 1 and STP = 1 (the first buffer of a new frame) is reached.

At the end of any transmit operation, whether successful or with errors, the controller always performs another polling operation, unless the next transmit descriptor is already known to be owned.

Whenever the DMA controller finishes copying a transmit frame from system memory, it sets the TINTx bit of INT0 that corresponds to the descriptor ring number to indicate that the buffers are no longer needed. This causes an interrupt signal if the INTREN bit of CMD0 the corresponding TINTENx bit of INTEN0 have been set.

### 3.10.2.7 Receive Polling

The receive polling process is also started by the host CPU when it sets the Receive Demand (RDMD) bit in the CMD0 register. Whenever the host CPU releases a receive buffer or group of receive buffers to the controller, it must set the RDMD bit. If the controller does not already own a receive buffer, the DMU polls the receive descriptor ring when RDMD is set. When a frame arrives from the network, the controller copies the frame to the next available receive buffer and then automatically polls the next descriptor.

The network controller provides a means to reduce the number of transmit interrupts by postponing the interrupt to the CPU until a programmable number of interrupt events have occurred or a programmable amount of time has elapsed since the first interrupt event occurred. See Section 3.10.15 on page 123.

When receive activity is present on the channel, the controller waits until 64 bytes have been received. If the frame is accepted based on all active addressing schemes at that time, the DMU is notified that a frame has been received.

As receive buffers become available in system memory, the DMA controller copies frame data from the receive FIFO into system memory. The controller sets the STP bit in the first descriptor of a frame. If the frame length exceeds the length of the current buffer, the controller passes ownership back to the system by writing 0s to the OWN and ENP bits of the descriptor when the first buffer is full. This activity continues until the controller recognizes the completion of the frame (the last byte of this receive message has been removed from the FIFO). The controller subsequently updates the current receive descriptor with the frame status (message byte count, VLAN info, error flags, etc.) and sets the ENP bit to 1. The controller then advances the internal ring pointer to make the next receive descriptor the new current receive descriptor.

If the driver does not provide the network controller with a descriptor in a timely fashion, the receive FIFO will eventually overflow. Subsequent frames are discarded and the RcvMissPkts MIB counter is incremented. Normal receive operation resumes when a descriptor is provided to the controller and sufficient data has been transferred from the network controller's receive FIFO into the system memory.

### 3.10.3 Software Interrupt Timer

The network controller is equipped with a software programmable free-running interrupt timer. The timer is constantly running and generates an interrupt STINT (INT0, bit 4) when STINTEN (INTEN0, bit 4) is set to 1. After generating the interrupt, the software timer loads the value stored in STVAL and restarts. The timer value STVAL is interpreted as an unsigned number with a resolution of 10.24  $\mu$ s. For instance, a value of 98 (62h) corresponds to 1.0 ms. The default value of STVAL is

FFFFh which corresponds to 0.671 seconds. A write to STVAL restarts the timer with the new contents of STVAL.

### 3.10.4 Media Access Control

The Media Access Control (MAC) engine incorporates the essential protocol requirements for operation of an Ethernet/IEEE 802.3-compliant node and provides the interface between the FIFO subsystem and the MII.

This section describes operation of the MAC engine when operating in half-duplex mode. The operation of the device in full-duplex mode is described in the section titled *Full-Duplex Operation*.

The MAC engine is compliant to Section 4 of IEEE standard 802.3, 1998 Edition.

The MAC engine provides programmable enhanced features designed to minimize host supervision, bus utilization, and pre- or post-message processing. These features include the ability to disable retries after a collision, dynamic FCS generation on a frame-by-frame basis, automatic pad field insertion and deletion to enforce minimum frame size attributes, automatic retransmission without reloading the FIFO, and automatic deletion of collision fragments. The MAC also provides a mechanism for automatically inserting, deleting, and modifying IEEE 802.3ac VLAN tags.

The two primary attributes of the MAC engine are:

- Transmit and receive message data encapsulation
  - Framing (frame boundary delimitation, frame synchronization)
  - Addressing (source and destination address handling)
  - Error detection (physical medium transmission errors)
- Media access management
  - Medium allocation (collision avoidance, except in full-duplex operation)
  - Contention resolution (collision handling, except in full-duplex operation)

#### 3.10.4.1 Transmit and Receive Message Data Encapsulation

The MAC engine provides minimum frame size enforcement for transmit and receive frames. When APAD\_XMT (CMD2, bit 6) is set to 1, transmit messages are padded with sufficient bytes (containing 00h) to ensure that the receiving station will observe an information field (destination address, source address, length/type, data, and FCS) of 64 bytes. When ASTRP\_RCV (CMD2, bit 13) is set to 1, the receiver automatically strips pad bytes from the received message by observing the value in the length field and by stripping excess bytes if this value is below the minimum data size (46 bytes). Both features can be independently over-ridden to allow messages shorter than the minimum frame size (less than 64 bytes of frame data) to be transmitted and/or received. The use of this feature reduces bus utilization because the pad bytes are not transferred into or out of main memory.

### 3.10.4.1.1 Framing

The MAC engine autonomously handles the construction of the transmit frame. Once the transmit FIFO has been filled to the predetermined threshold and access to the channel is currently permitted, the MAC engine commences the 7-byte preamble sequence (a repeating 10101010 sequence, where first bit transmitted is a 1). The MAC engine subsequently appends the Start Frame Delimiter (SFD) sequence (10101011) followed by the serialized data from the transmit FIFO. Once the data has been transmitted, the MAC engine appends the FCS (most significant bit first) which was computed on the entire frame, starting with the destination address. The user is responsible for the correct ordering and content of the destination address, source address, length/type, and frame data fields of the frame.

The receiver section of the MAC engine detects the incoming preamble sequence when the RX\_DV signal is activated by the external PHY. The MAC discards the preamble and begins searching for the SFD. (In the case of 100BASE-T4, for which there is no preamble to discard, the SFD is the first two nibbles received.) Once the SFD is detected, all subsequent nibbles are treated as part of the frame. If automatic pad stripping is enabled, the MAC engine inspects the length field so that it can strip unnecessary pad characters and pass the remaining bytes through the receive FIFO to the host. If pad stripping is performed, the MAC engine also strips the received FCS bytes, although normal FCS computation and checking do occur. Note that apart from pad stripping, the frame is passed unmodified to the host. If the length field has a value of 46 or greater, all frame bytes including FCS are passed unmodified to the receive buffer, regardless of the actual frame length.

If the frame terminates or suffers a collision before 64 bytes of information (after SFD) have been received, the MAC engine automatically deletes the frame from the receive FIFO, without host intervention. The network controller has the ability to accept runt packets for diagnostic purposes and proprietary networks.

### 3.10.4.2 Destination Address Handling

The first 6 bytes of information after SFD is interpreted as the destination address field. The MAC engine provides facilities for physical (unicast), logical (multicast), and broadcast address reception.

#### 3.10.4.2.1 Error Detection

The MAC engine provides several facilities which count and recover from errors on the medium. In addition, it protects the network from gross errors due to inability of the host to keep pace with the MAC engine activity.

On completion of transmission, the MAC engine updates various counters that are described in the Statistics Counters section. The host CPU can read these counters at any time for network management purposes.

The MAC engine also attempts to prevent the creation of any network error due to the inability of the host to service the MAC engine. During transmission, if the host fails to keep the transmit FIFO filled sufficiently, causing an underflow, the MAC engine ensures the message is sent with an invalid FCS, which causes the receiver to reject the message.

The MAC engine can be programmed to try to transmit the same frame again after a FIFO underflow. The transmitter backoff logic can also be programmed to treat late collisions just like normal collisions.

The status of each receive message is available in the appropriate Receive Message Descriptor (RMD). All received frames are passed to the host regardless of any error.

During the reception, the FCS is generated on every nibble (including the dribbling bits) coming from the MII, although the internally saved FCS value is only updated on each byte boundary. The MAC engine ignores an extra nibble at the end of a message, which corresponds to dribbling bits on the network medium. A framing or alignment error is reported to the user if an FCS error is detected and there is an extra nibble in the message. If there is an extra nibble but no FCS error, no framing error is reported.

### 3.10.4.3 Media Access Management

A basic requirement for all stations on the network is to provide fair access to the network. The IEEE 802.3/Ethernet protocols define a media access mechanism which permits all stations to access the channel with equality. Any node can attempt to contend for the channel by waiting for a predetermined time (Inter Packet Gap) after the last activity, before transmitting on the media. The channel is a multidrop communications medium (with various topological configurations permitted), which allows a single station to transmit and all other stations to receive. If two nodes simultaneously contend for the channel, their signals interact causing loss of data, defined as a collision. It is the responsibility of the MAC to attempt to avoid and to recover from collisions.

#### 3.10.4.3.1 Medium Allocation

The IEEE standard 802.3, 1998 Edition, requires that the CSMA/CD MAC monitor the medium for traffic by watching for carrier activity. When carrier is detected, the medium is considered busy, and the MAC should defer to the existing message.

The standard allows an optional two-part deferral after a receive message. For details, see the IEEE standard 802.3, 1998 Edition, 4.2.3.2.1.

The MAC engine implements the optional receive two part deferral algorithm, with an InterFrameSpacing-Part1 (IFS1) time of 60 bit times and an InterFrameSpacingPart 2 time of 36 bit times. The Inter Packet Gap (IPG) timer starts timing the 96-bit InterFrameSpacing after the receive carrier is deasserted.

During the first part deferral (IFS1), the MAC engine defers any pending transmit frame and responds to the receive message. If carrier sense or collision is detected during the first part of the gap, the IPG counter is cleared to 0 continuously until carrier sense and collision are both deasserted, at which point the IPG counter resumes the 96-bit time count once again. Once the IPG counter reaches the IFS1 count (60-bit times), the MAC engine does not defer to a receive frame if a transmit frame is pending. Instead, when the IPG count reaches 96-bit times, the transmitter starts transmitting, which causes a collision. The MAC engine completes the preamble (64-bit) and jam (32-bit) sequence before ceasing transmission and invoking the random backoff algorithm.

The MAC engine allows the user to program both the IPG and the first part deferral (IFS1) through the IFS and IFS1 registers. The user can change the IPG value from its default of 96-bit times to compensate for delays through the external PHY device. Changing IFS1 alters the period for which the MAC engine defers to incoming receive frames.

**Note:** *Care must be exercised when altering these parameters. Undesirable network activity could result!*

This transmit two-part deferral algorithm is implemented as an option which can be disabled using the DXMT2PD bit (CMD2, bit 10). When DXMT2PD is set to 1, the IFS1 register is ignored, and the value 0 is used for the InterFrameSpacingPart1 parameter. However, the IPG value is still valid.

When the MAC engine device operates in full-duplex mode, the IPG timer starts counting when TX\_EN is de-asserted. CRS is ignored in full-duplex mode.

#### 3.10.4.3.2 Signal Quality Error (SQE) Test

During the time period immediately after a transmission has been completed, an external transceiver operating in the 10 Mb/s half-duplex mode should generate an SQE Test signal on the COL pin within 0.6  $\mu$ s to 1.6  $\mu$ s after the transmission ceases. Therefore, when the network controller is operating in half-duplex mode, the IPG counter ignores the COL signal during the first 40-bit times of the inter-packet gap. This 40-bit times is the time period in which the SQE Test message is expected.

The SQE Test was originally designed to check the integrity of the Collision Detection mechanism independently of the Transmit and Receive capabilities of the Physical Layer. However, MII-based PHY devices detect collisions by sensing receptions that occur during transmissions, a process that does not require a separate level-sensing collision detection mechanism. Collision detection is therefore dependent on the health of the receive channel. Since the Link Monitor function checks the health of the receive channel, the SQE test is not very useful for MII-based devices. Therefore, the device does not report or count SQE Test failures.

#### 3.10.4.3.3 Collision Handling

Collision detection is performed and reported to the MAC engine through the COL input pin. Since the COL signal is not required to be synchronized with TX\_CLK when the device is operating in half-duplex mode, the COL signal must be asserted for at least three TX\_CLK cycles in order to be detected reliably.

If a collision is detected before the complete preamble/SFD sequence has been transmitted, the MAC engine completes the preamble/SFD before appending the jam sequence. If a collision is detected after the preamble/SFD has been completed, but before 512 bits have been transmitted, the MAC engine aborts the transmission and appends the jam sequence immediately. The jam sequence is a 32-bit all zeros pattern.

The MAC engine attempts to transmit a frame a total of 16 times (initial attempt plus 15 retries) due to normal collisions (those within the slot time). Detection of collision causes the transmission to be rescheduled to a time determined by the random backoff algorithm. If a single retry was required, the

XmtOneCollision counter is incremented. If more than one retry was required, the XmtMultipleCollision counter is incremented. If all 16 attempts experienced collisions, the XmtExcessiveCollision counter is incremented. After an excessive collision error, the transmit message is flushed from the FIFO.

If retries have been disabled by setting the DRTY bit in the CMD2 register, the MAC engine abandons transmission of the frame on detection of the first collision. In this case, XmtExcessiveCollision counter is incremented, and the transmit message is flushed from the FIFO.

If a collision is detected after 512-bit times have been transmitted, the collision is termed a late collision. The MAC engine aborts the transmission, appends the jam sequence, and increments the XmtLateCollision counter. No retry attempt is scheduled on detection of a late collision. The transmit message is flushed from the FIFO.

The IEEE 802.3 Standard requires use of a “truncated binary exponential backoff” algorithm, which provides a controlled pseudo random mechanism to enforce the collision backoff interval, before retransmission is attempted. For details, see IEEE standard 802.3, 1998 Edition, 4.2.3.2.5.

The MAC engine provides an alternative algorithm, which suspends the counting of the slot time/IPG during the time that receive carrier sense is detected. This aids in networks where large numbers of nodes are present, and numerous nodes can be in collision. It effectively accelerates the increase in the backoff time in busy networks and allows nodes not involved in the collision to access the channel, while the colliding nodes await a reduction in channel activity. Once channel activity is reduced, the nodes resolving the collision time-out their slot time counters as normal.

This modified backoff algorithm is enabled when EMBA (CMD2, bit 11) is set to 1.

### 3.10.5 Transmit Operation

The transmit operation and features of the network controller are controlled by programmable options. The controller provides a large transmit FIFO to provide frame buffering for increased system latency, automatic retransmission with no FIFO reload, and automatic transmit padding.

#### 3.10.5.1 Transmit Function Programming

Automatic transmit features such as retry on collision, FCS generation/transmission, and pad field insertion can all be programmed to provide flexibility in the (re-) transmission of messages.

Automatic pad field insertion is controlled by the APAD\_XMT bit in the CMD2 register.

The FCS generation/transmission feature can be programmed as a static feature (DXMTFCS bit in CMD2) or dynamically on a frame-by-frame basis (ADD\_FCS bit in the transmit descriptor).

REX\_UFLO (CMD2, bit 1) can be programmed to cause the transmitter to automatically restart the transmission process instead of discarding a frame that experiences an underflow error. After an underflow error the retransmission does not begin until the entire frame has been loaded into the transmit FIFO.



Transmit Start Point (XMTSP) in CTRL1 sets the point when the transmitter actually attempts to transmit a frame onto the media. A minimum of XMTSP bytes must be written to the transmit FIFO for the current frame before transmission of the current frame will begin. (When automatically padded packets are being sent, it is conceivable that the XMTSP is not reached when all of the data has been transferred to the FIFO. In this case, the transmission begins when all of the frame data has been placed into the transmit FIFO.) The default value of XMTSP is 01b, meaning there has to be 64 bytes in the transmit FIFO to start a transmission.

In order to ensure that collisions occurring within 512 bit times from the start of transmission (including preamble) are automatically retried with no host intervention, the transmit FIFO ensures that data contained within the FIFO is not overwritten until at least 64 bytes (512 bits) of preamble plus address, length, and data fields have been transmitted onto the network without encountering a collision. If the REX\_UFLO bit is set, the transmit data is not overwritten until the frame has been either transmitted or discarded.

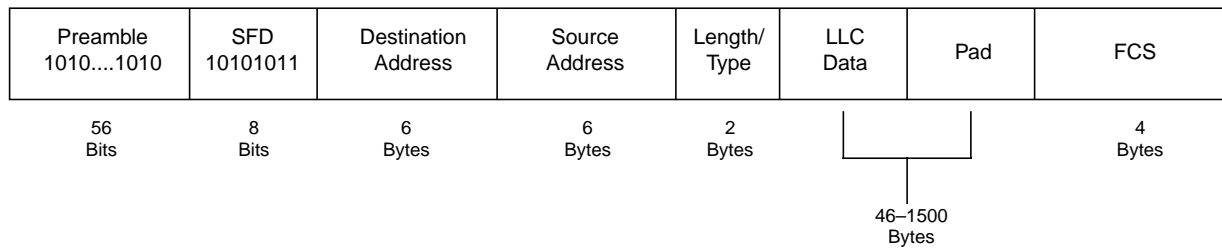
In a (nonstandard) system that allows frames that are larger than the transmit FIFO size, the XMTSP bit should not be set to 01b (full frame), and the REX\_UFLO bit should not be set.

#### **3.10.5.1.1 Automatic Pad Generation**

Transmit frames can be automatically padded to extend them to 64 data bytes (excluding preamble). This allows the minimum frame size of 64 bytes (512 bits) for IEEE 802.3/Ethernet to be ensured with no software intervention from the host/controlling process. Setting the APAD\_XMT bit in CMD2 enables the automatic padding feature. The pad is placed between the LLC data field and FCS field in the IEEE 802.3 frame. FCS is always added if the frame is padded, regardless of the state of DXMTFCS (CMD2, bit 8) or ADD\_FCS (in the transmit descriptor). The transmit frame is padded by bytes with the value of 00H. The default value of APAD\_XMT is 0 after H\_RESET, which disables automatic pad generation.

If automatic pad generation is disabled, the software is responsible for insuring that the minimum frame size requirement is met. The hardware can reliably transmit frames ranging in size from 16 to 65536 octets.

It is the responsibility of upper layer software to correctly define the actual length/type field contained in the message to correspond to the total number of LLC Data bytes encapsulated in the frame (length/type field as defined in the IEEE 802.3 standard). The length value contained in the message is not used by the network controller to compute the actual number of pad bytes to be inserted. The controller appends pad bytes dependent on the actual number of bits transmitted onto the network. Once the last data byte of the frame has completed, prior to appending the FCS, the controller checks to ensure that 544 bits have been transmitted. If not, pad bytes are added to extend the frame size to this value, and the FCS is then added. See Figure 18.



**Figure 18. ISO 8802-3 (IEEE/ANSI 802.3) Data Frame**

The 544 bit count is derived from the following:

- Minimum frame size (excluding preamble/SFD, including FCS)  
64 bytes or 512 bits
- Preamble/SFD size: 8 bytes or 64 bits
- FCS size: 4 bytes or 32 bits

At the point that FCS is to be appended, the transmitted frame should contain:

- Preamble/SFD + (Min Frame Size – FCS):  
 $64 + (512-32) = 544$  bits

A minimum length transmit frame from the network controller, therefore, is 576 bits, after the FCS is appended.

### 3.10.5.2 Transmit FCS Generation

Automatic generation and transmission of FCS for a transmit frame depends on the value of DXMTFCS (CMD2, bit 8). If DXMTFCS is cleared to 0, the transmitter generates and appends the FCS to the transmitted frame. If the transmitter modifies the frame data because of automatic padding or VLAN tag manipulation, it appends the FCS regardless of the state of DXMTFCS or ADD\_FCS (in the transmit descriptor). Note that the calculated FCS is transmitted most significant bit first. The default value of DXMTFCS is 0 after H\_RESET.

When DXMTFCS is set to 1, the ADD\_FCS bit in the transmit descriptor allows the automatic generation and transmission of FCS on a frame-by-frame basis. When DXMTFCS is set to 1, a valid FCS field is appended only to those frames whose TX descriptors have their ADD\_FCS bits set to 1. If a frame is split into more than one buffer, the ADD\_FCS bit is ignored in all descriptors except for the first.

### 3.10.5.3 Transmit Error Conditions

The transmitter detects the following error conditions and increments the appropriate error counters when they occur:

- Loss of carrier
- Late collision

- Transmit FIFO Underflow

Late collision errors can only occur when the device is operating in half-duplex mode. Loss of carrier and transmit FIFO underflow errors are possible when the device is operating in half- or full-duplex mode.

When a late collision or underflow error occurs in the middle of a multi-buffer frame transmission, the appropriate error counter is incremented, and the transmission is aborted with an inverted FCS field appended to the frame. The OWN bits in the current and subsequent descriptors are cleared until the STP bit is found, indicating the start of the next frame.

If REX\_UFLO (CMD2, bit 1) is set, the transmitter does not flush the frame data from the transmit FIFO after a transmit FIFO underflow error occurs. Instead, it waits until the entire frame has been copied into the transmit FIFO, and then it restarts the transmission process.

#### **3.10.5.3.1 Loss of Carrier**

The XmtLossCarrier counter is incremented if transmit is attempted when the LINK\_STAT bit in the STAT0 register is 0.

The LINK\_STAT bit is set automatically during the auto-negotiation process when the external PHY device determines that the link is up. Alternatively the software can bypass the auto-negotiation process by clearing the EN\_PMGR bit in CMD3 and setting FORCE\_LINK\_STATUS, also in CMD3.

#### **3.10.5.3.2 Late Collision**

A late collision is detected when the device is operating in half-duplex mode and a collision condition occurs after one slot time (512 bit times) after the transmit process was initiated (first bit of preamble commenced). When it detects a late collision, the controller increments the XmtLateCollision counter. The controller abandons the transmit process for that frame and processes the next transmit frame in the ring.

#### **3.10.5.3.3 Transmit FIFO Underflow**

An underflow error occurs when the transmitter runs out of data from the transmit FIFO in the middle of a transmission. When this happens, an inverted FCS is appended to the frame so that the intended LAN Ethernet receiver will ignore the frame, and the XmtUnderrunPkts counter is incremented. If REX\_UFLO (CMD2, bit 1) is set to 1, the transmitter then waits until the entire frame has been loaded into the transmit FIFO, and then it restarts the transmission of the same frame. If the REX\_UFLO is cleared to 0, the transmitter does not attempt to retransmit the aborted frame.

The REX\_UFLO bit should not be set in a (nonstandard) system that allows frames that are larger than the transmit FIFO size.

### 3.10.5.4 Conduit Mode

The Conduit Mode of operation allows the LAN Ethernet controller to pass frame data to an external MAC-type device attached to the MII bus. This option eliminates the need for a frame buffer in the external device. In this mode the backoff logic is disabled, and the COL and CRS signals from the MII are used to control transmit retry and frame discard. The internal synchronizers on the COL and CRS inputs are bypassed, and the external device must provide COL and CRS inputs that are synchronized with TX\_CLK.

The CRS signal is used for flow control across the MII bus. The external device controls the local transmit MAC by asserting CRS after the local device asserts TX\_EN. CRS is held on until the external device has finished processing the frame, preventing the local device from starting the transmission of the next frame. The current transmit frame is retained in the FIFO until CRS deasserts. If COL is asserted at any time while CRS is asserted, the frame is requeued and retransmitted.

In Conduit Mode all MIB counters work like they do in half duplex mode, except that the XmtExcessiveCollision counter is incremented when COL is asserted while CRS is deasserted (as explained below), and not when a frame has experienced 16 unsuccessful transmission attempts.

Conduit Mode is enabled by setting the CONDUIT\_MODE bit (CMD2, bit 29).

In Conduit Mode the MII handshake is modified as follows:

1. The CRS and COL inputs must be synchronized with TX\_CLK, changing state at the falling edge of TX\_CLK. In this mode CRS and COL must meet set-up and hold timing requirements with respect to the rising edge of TX\_CLK similar to the timing requirements for the synchronous signals in the MII receive path.
2. If COL is asserted while CRS is also asserted, the frame is requeued for transmission. The frame is not dropped when CRS is deasserted.
3. If COL is not asserted during the time that CRS is asserted, the appropriate MIB counters are incremented to indicate a successful transmission, and the frame is discarded from memory when CRS is deasserted.
4. If COL is asserted during the same TX\_CLK cycle in which CRS is deasserted, the frame is discarded and counted as an excessive collision error. (The XmtExcessiveCollision counter is incremented.) In this case CRS must be held Low for at least 2 TX\_CLK cycles. After 2 TX\_CLK cycles, CRS may be asserted again to delay the transfer of the next frame over the MII.
5. Following deassertion of CRS, normal IPG is counted and next frame transmitted. The external device asserts CRS when it sees TX\_EN asserted.

Figure 19 shows a frame that is requeued and then successfully transmitted. Figure 20 shows a frame that is dropped because the external device reports excessive collisions. In this figure a second frame is transmitted successfully with no collisions.

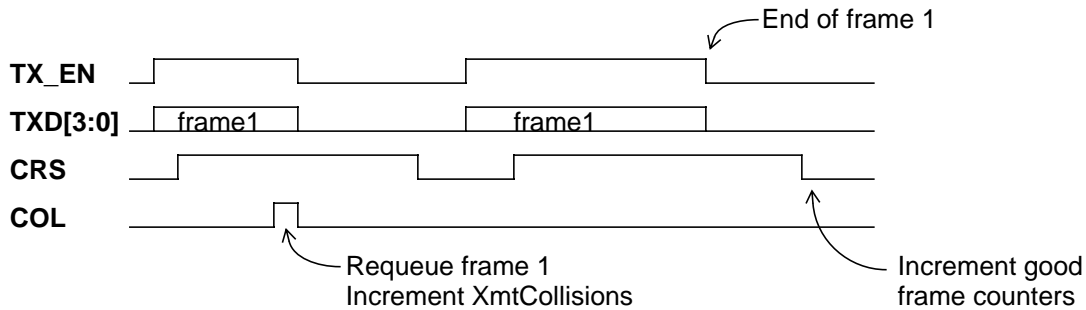


Figure 19. Conduit Mode Transmission with One Collision

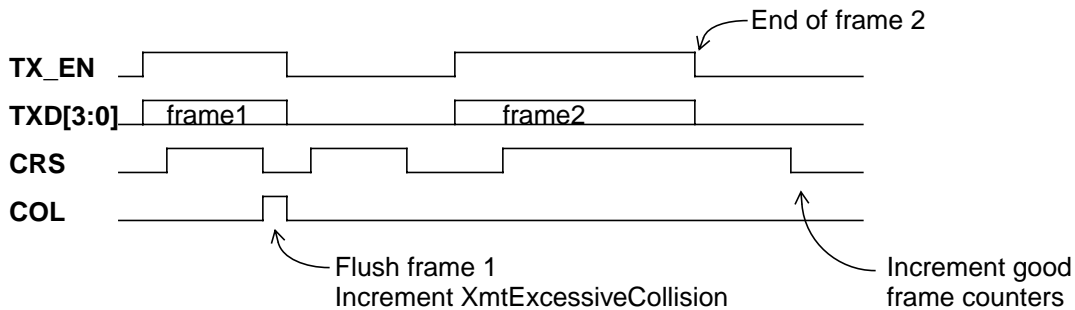


Figure 20. Conduit Mode Excessive Collisions Error

### 3.10.6 Receive Operation

The receive operation and features of the network controller are controlled by programmable options. The controller uses a large receive FIFO to provide frame buffering for increased system latency, automatic flushing of collision fragments (runt packets), automatic receive pad stripping, and a variety of address match options.

#### 3.10.6.1 Receive Function Programming

Automatic pad field stripping is enabled by setting the ASTRP\_RCV bit in CMD2. This can provide flexibility in the reception of messages using the IEEE 802.3 frame format.

The device can be programmed to accept all receive frames regardless of destination address by setting the PROM bit in CMD2. Acceptance of unicast and broadcast frames can be individually turned off by setting the DRCVPA or DRCVBC bits in CMD2. The Physical Address register stores the address that the MAC receiver compares to the destination address of the incoming frame for a unicast address match. The Logical Address Filter register serves as a hash filter for multicast address matching.

For test purposes, the controller can be programmed to accept runt packets of 12 bytes or larger by setting RPA in CMD2.

### 3.10.6.2 Address Matching

MAC addresses are classified as either unicast, multicast, or broadcast. The least significant bit of the first octet of a MAC address (the I/G bit) indicates whether the address is unicast (individual) or multicast (group). If the bit is 0, the address is an individual address and is intended to be received by a single network station. If the bit is 1, the address is a group address and is intended to be received by more than one network station. The special address that consists of all 1s is known as the broadcast address. Frames addressed to the broadcast address are intended to be received by all stations on the local area network.

The network controller contains three address matching mechanisms that determine which incoming frames it will receive:

1. The Physical Address Match Comparator
2. The Broadcast Address Match Comparator
3. The Multicast Hash Filter

In addition if the Promiscuous Mode bit (PROM, CMD2, bit 16) is set, all valid frames are received.

An incoming frame is saved in system memory if any one of these mechanisms signals a match for that frame.

The Physical Address Match Comparator signals a match if the Destination Address (DA) field of the incoming frame exactly matches the contents of the Physical Address Register (PADR). The byte ordering is such that the first byte received from the network (after the SFD) must match the least significant byte of PADR (PADR[7:0]), and the sixth byte received must match the most significant byte of PADR (PADR[47:40]). The Physical Address Match comparator can be disabled by setting the Disable Receive Physical Address Match (DRCVPA) bit in the CMD2 register.

The Broadcast Address Match Comparator signals a match if the Destination Address (DA) field of the incoming frame is all ones. The Broadcast Address Match comparator can be disabled by setting the Disable Receive Broadcast Address Match (DRCVBC) bit in the CMD2 register. If DRCVBC is set and PROM is cleared, no broadcast frames are received even if another address match function signals a match.

#### 3.10.6.2.1 The Logical Address Filter

If the least significant bit of the first octet of a MAC address (as transmitted over the network) is a 1, the address is a multicast (or logical) address that indicates that the frame is meant to be received by more than one network node. The network controller can be programmed to accept frames addressed to any number of multicast addresses. The Logical Address Filter is used to filter out unwanted multicast frames.

The Logical Address Filter block consists of a 64-bit Logical Address Filter Table and associated control logic. The Logical Address Filter logic uses the CRC generator from the Frame Check Sequence block to perform a calculation on the DA field of a frame to determine whether or not to signal a match. The CRC generator calculates a CRC value for the DA field of the frame. If the destination address is a multicast address, the 6 most significant bits of this CRC value are used as a pointer to a bit in the Logical Address Filter Table. If the selected bit is 1, the Logical Address Filter signals a match. See Figure 21 on page 96.

The broadcast address (all ones) is treated as a special case, not as an instance of a multicast address. A frame addressed to the broadcast address does not cause a multicast address match. This means that if the DRCVBC bit is set, broadcast frames are not received even if the bit in the Logical Address Filter that corresponds to the broadcast address is set.

The host CPU can read and write the Logical Address Filter Register directly.

Although the Logical Address Filter rejects the vast majority of addresses that are not intended for this device, it is not a perfect filter. More than one multicast address maps to each bit in the filter table. Therefore the host software must compare the DA field of each frame that was accepted by the Logical Address Filter with a software table of acceptable multicast addresses. If the Logical Address Filter signals a match for an incoming frame, the Logical Address Filter Match (LAFM) bit is set in the receive descriptor that corresponds to the frame. The host CPU can examine this bit to determine whether or not it should look up the destination address in the software address table.

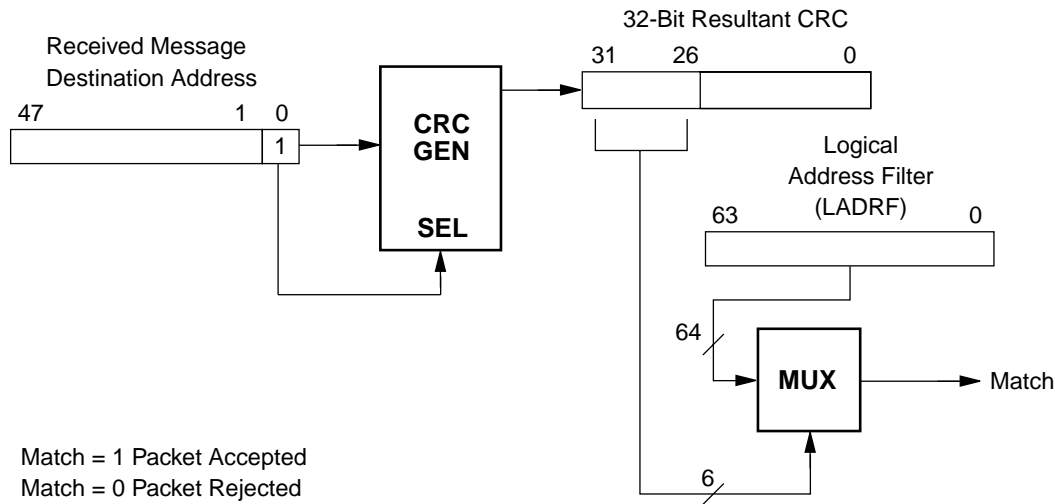
#### **3.10.6.2.2 Address Match Classification**

The receive descriptor associated with a frame includes 3 bits that indicate why the frame was received.

The Physical Address Match (PAM) bit is set when the destination address matches the contents of the PADR.

The Logical Address Filter Match (LAFM) bit is set when the destination address is a multicast address that causes the Multicast Hash Filter to signal a match and the destination address is not the broadcast address.

The Broadcast Address Match (BAM) bit is set when the destination address is the broadcast address.



**Figure 21. Address Match Logic**

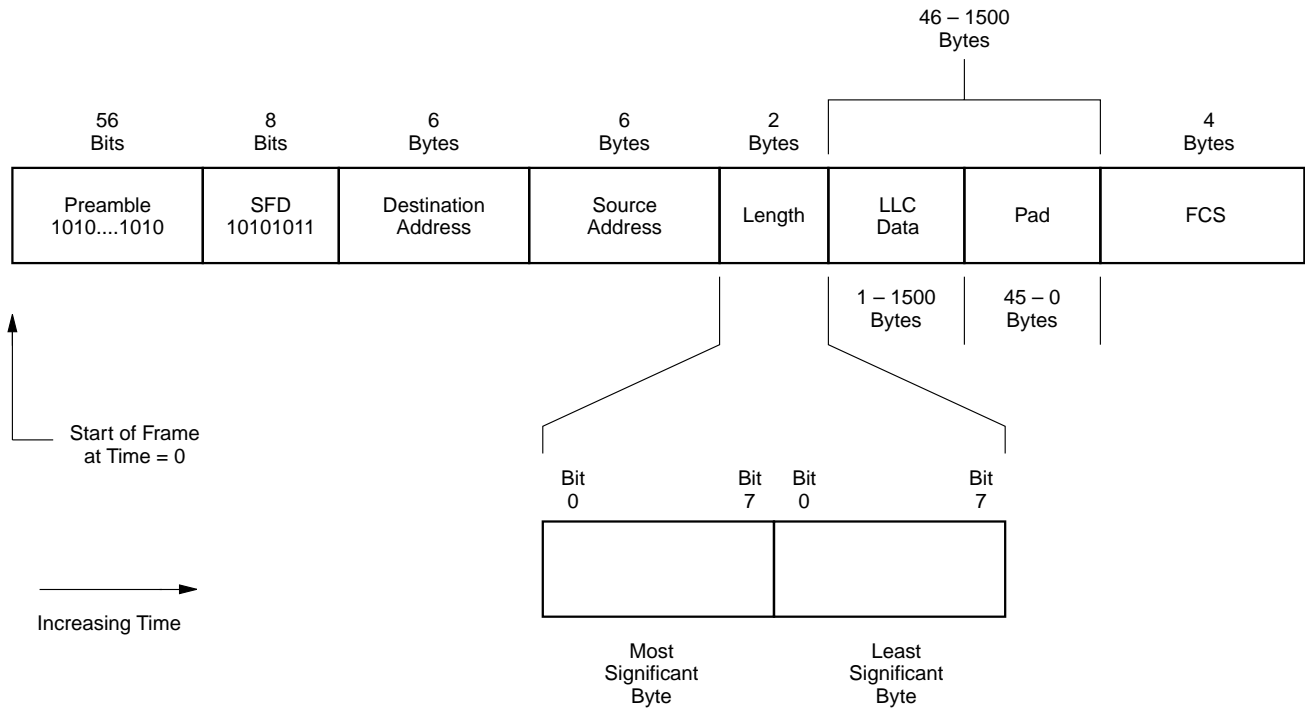
### 3.10.6.3 Automatic Pad Stripping

During reception of an IEEE 802.3 frame, the pad field can be stripped automatically. Setting `ASTRP_RCV` (`CMD2`, bit 13) to 1 enables the automatic pad stripping feature. The pad field is stripped before the frame is passed to the FIFO, thus preserving FIFO space for additional frames. The FCS field is verified, but it is also stripped, because it was computed at the transmitting station based on the data and pad field characters, and is invalid for a receive frame that has had the pad characters stripped.

The number of bytes to be stripped is calculated from the embedded length field (as defined in the IEEE standard 802.3) contained in the frame. The length indicates the actual number of LLC data bytes contained in the message. Any received frame that contains a length field less than 46 bytes has the pad field stripped (if `ASTRP_RCV` is set). Receive frames that have a length field of 46 bytes or greater are passed to the host unmodified.

Figure 22 on page 97 shows the byte/bit ordering of the received length field for an IEEE 802.3-compatible frame format.





**Figure 22. IEEE 802.3 Frame and Length Field Transmission Order**

Since any valid Ethernet Type field value is greater than a valid IEEE 802.3 Length field, the network controller does not attempt to strip valid Ethernet frames.

*Note:* For some network protocols, the value passed in the Ethernet Type and/or IEEE 802.3 Length field may not be compliant with either standard and may cause problems if pad stripping is enabled.

**3.10.6.4 Receive FCS Checking**

Reception and checking of the received FCS is performed automatically by the network controller. Note that if the Automatic Pad Stripping feature is enabled, the FCS for padded frames is verified against the value computed for the incoming bit stream including pad characters, but the FCS value for a padded frame is not passed to the host. If an FCS error is detected in any frame, the error is reported in the CRC bit in the Receive Descriptor.

**3.10.6.5 Receive Exception Conditions**

The network controller detects and collects statistics on the following error conditions:

- FCS errors
- Framing errors
- Receiver overflow events

These error conditions are reported in the corresponding receive descriptors. The RcvFCSErrors, RcvAlignmentErrors, or RcvMissPkts counter is also incremented when one of these events occurs.

### 3.10.7 Statistics Counters

In order to provide network management information with minimum host CPU overhead, the network controller automatically maintains a set of 32-bit controller statistics counters.

The host CPU can access the statistics counters two ways; it can force all counters to be cleared in one operation, or it can read a single 32-bit counter in one operation.

The host CPU can clear all counters by executing the following steps:

1. Poll the MIB\_CMD\_ACTIVE bit in the MIB\_ADDR register until this bit is 0.
2. Set the MIB\_CLEAR bit and clear the MIB\_RD\_CMD bit in the MIB\_ADDR register.

The MIB\_CMD\_ACTIVE bit is automatically cleared to 0 when the counters have all been cleared. The MIB\_CLEAR command takes about 1.3  $\mu$ s.

The host CPU can read any statistics counter indirectly through the MIB\_ADDR and MIB\_DATA registers as described in Section 4.10.1 on page 306. The addresses that should be written to the MIB\_ADDR register are shown in Table 27 on page 99 and Table 28 on page 102. When the host CPU sets the RD\_CMD bit in the MIB\_ADDR register, the controller loads the contents of the selected MIB counter into the MIB\_DATA register between 210 ns and 2  $\mu$ s later.

#### 3.10.7.1 Receive Statistics Counters

The receive statistics counters are defined and the Management Information Base (MIB) objects that they support are listed in Table 26 on page 98.

For these counters, a valid frame is defined as a frame that has a correct FCS value and whose length is between 64 octets and a maximum frame size that depends on the state of the JUMBO and VSIZE bits (CMD3, bits 21 and 20). If both of these bits are 0, the maximum frame size is 1518 octets. Setting the JUMBO bit adds 7500 octets to the maximum frame size, and setting the VSIZE bit adds 4 octets.

The Maximum Valid Frame size is summarized in Table 26 on page 98.

**Table 26. Maximum Valid Frame Size**

JUMBO	VSIZE	Max. Valid Frame Size (Octets)
0	0	1518
0	1	1522
1	0	9018
1	1	9022

Frames longer than 9022 bytes can be received or transmitted successfully, but they are counted as oversized frames. However, frames longer than 65536 bytes might not be handled properly.

In Table 27 on page 99, the Address column gives the value that the host CPU must write to the MIB\_ADDR register in order to read the counter.

**Table 27. Receive Statistics Counters**

Address (hex)	Receive Counter Name	MIB Object Supported	Description of Counter/Comments
00	RcvMissPkts	RMON etherStatsDropEvents RMON etherHistoryDropEvents MIB-II ifInDiscards E-like dot3StatsInternalMacReceiveErrors	The number of times a receive packet was dropped due to lack of resources. This includes the number of times a packet was dropped due to receive FIFO overflow and lack of receive descriptor. This count does not include packets that are counted as undersize, oversize, misaligned or bad FCS packets. When RUN=0 any packet that is not a fragment and that passes the address match tests causes the RcvMissPkts counter to increment.
01	RcvOctets	RMON etherStatsOctets RMON etherHistoryOctets MIB-II IfInOctets	The total number of octets of data received including octets from invalid frames. This does not include the preamble but does include the FCS bits. The RcvOctets counter is incremented whenever the receiver receives an octet.
02	RcvBroadCastPkts	RMON etherStatsBroadcastPkts RMON etherHistoryBroadcastPkts EXT-MIB-II ifInBroadcastPkts	The total number of valid frames received that are addressed to a broadcast address. This counter does not include errored broadcast packets or valid multicast packets.
03	RcvMultiCastPkts	RMON etherStatsMulticastPkts RMON etherHistoryMulticastPkts EXT-MIB-II ifInMulticastPkts	The total number of valid frames received that are addressed to a multicast address. This counter does not include errored multicast packets or valid broadcast packets.
04	RcvUndersizePkts	RMON etherStatsUndersizePkts RMON etherHistoryUndersizePkts	The total number of valid frames received that are less than 64 bytes long (including the FCS) and do not have any error. SFD must be received so that the FCS can be calculated.
05	RcvOversizePkts	RMON etherStatsOversizePkts RMON etherHistoryOversizePkts E-like MIB dot3StatsFrameTooLongs	The total number of packets received that are greater than the maximum valid frame size (including the FCS) and do not have any error. SFD must be received so that the FCS can be calculated.

**Table 27. Receive Statistics Counters (Continued)**

Address (hex)	Receive Counter Name	MIB Object Supported	Description of Counter/Comments
06	RcvFragments	RMON etherStatsFragments RMON etherHistoryFragments	The number of packets received that are less than 64 bytes (not including the preamble or SFD) and have either an FCS error or an alignment error.
07	RcvJabbers	RMON etherStatsJabbers RMON etherHistoryJabbers	The number of packets received that are greater than the maximum valid frame size and have either an FCS error or an alignment error.
08	RcvUnicastPkts	MIB-II ifInUcastPkts	The number of valid frames received that are not addressed to a multicast address or a broadcast address. This counter does not include errored unicast packets.
09	RcvAlignmentErrors	E-like MIB dot3StatsAlignmentErrors	The number of packets received that are between 64 and the maximum valid frame size (excluding preamble/SFD but including FCS), inclusive, and have a bad FCS with non-integral number of bytes.
0a	RcvFCSErrors	E-like MIB dot3StatsFCSErrors	The total number of packets received that are between 64 and the maximum valid frame size (excluding preamble/SFD but including FCS), inclusive, and have a bad FCS with an integral number of bytes. This counter also counts packets with a correct FCS if RX_ER occurs when valid carrier RX_DV is present.
0b	RcvGoodOctets	RMON hostInOctets RMON hostTimeInOctets	The total number of bytes received by a port. Bytes are 8-bit quantities received after the SFD. This does not include preamble or bytes from erroneous packets, but does include the FCS.
0c	RcvMACCtrl	802.3x aMACControlFramesReceived	The total number of valid frames received with a lengthOrType field value equal to 8808h.
0d	RcvFlowCtrl	802.3x aPAUSEMACCtrlFramesReceived	The total number of valid frames received with (1) a lengthOrType field value equal to 8808h and (2) an opcode equal to 1.
0e	RcvPkts64Octets	RMON etherStatsPkts64Octets	The total number of packets received (including error packets) that are 64 bytes long.

**Table 27. Receive Statistics Counters (Continued)**

Addresses (hex)	Receive Counter Name	MIB Object Supported	Description of Counter/Comments
0f	RcvPkts65to127Octets	RMON etherStatsPkts65to127Octets	The total number of packets received (including error packets) that are 65 bytes to 127 bytes long, inclusive.
10	RcvPkts128to255Octets	RMON etherStatsPkts128to255Octets	The total number of packets received (including error packets) that are 128 bytes to 255 bytes long, inclusive.
11	RcvPkts256to511Octets	RMON etherStatsPkts256to511Octets	The total number of packets received (including error packets) that are 256 bytes to 511 bytes long, inclusive.
12	RcvPkts512to1023Octets	RMON etherStatsPkts512to1023Octets	The total number of packets received (including error packets) that are 512 bytes to 1023 bytes long, inclusive.
13	RcvPkts1024to1518Octets	RMON etherStatsPkts1024to1518Octets	The total number of packets received (including error packets) that are 1024 bytes to 1518 (1522 when VLAN set) bytes long, inclusive.
14	RcvUnsupportedOpcodes	802.3x an UnsupportedOpCodesReceived	The total number of valid frames received with (1) a lengthOrType field value equal to 8808h and (2) an opcode not equal to 1.
15	RcvSymbolErrors		The number of times when valid carrier (CRS) was present and there was at least one occurrence of an invalid data symbol (RX_ER). This counter is incremented only once per valid carrier event (once per frame), and if a collision is present, this counter must not be incremented.
16	RcvDropPktsRing0		The number of packets that were dropped because no descriptor was available in Receive Descriptor Ring 0 when it was needed.
17-19	Reserved		
1a	RcvJumboPkts		The total number of packets received (including error packets) that are 1519 bytes to 9018 bytes long, inclusive when VSIZE=0 or 1523 to 9022 when VSIZE=1.
1b-1f	Reserved		

### 3.10.7.2 Transmit Statistics Counters

Table 28 on page 102 describes the statistics counters associated with the transmitter and lists the MIB objects that these counters support.

The Address column gives the value that the host CPU must write to the MIB\_ADDR register in order to read the counter.

**Table 28. Transmit Statistics Counters**

Addresses (hex)	Transmit Counter Name	MIB Object Supported	Description of Counter/Comments
20	XmtUnderrunPkts	RMON etherStatsDropEvents RMON etherHistoryDropEvents MIB-II ifOutDiscards E-like dot3StatsInternalMacTranmsitErrors	The number of times a packet was dropped due to transmit FIFO underrun.
21	XmtOctets	RMON etherStatsOctets RMON etherHistoryOctets RMON hostOutOctets RMON hostTimeOutOctets MIB-II IfOutOctets	The total number of octets of data transmitted. This does not include the preamble but does include the FCS bits. The XmtOctets counter is incremented whenever the transmitter transmits an octet.
22	XmtPackets	RMON etherStatsPkts RMON etherHistoryPkts RMON hostOutPkts RMON hostTimeOutPkts BRIDGE-MIB dot1dTpPortOutFrames	The number of packets transmitted. This does not include packets transmitted with errors (i.e., collision fragments and partial packets due to transmit FIFO under runs).
23	XmtBroadCastPkts	RMON etherStatsBroadcastPkts RMON etherHistoryBroadcastPkts RMON hostOutBroadcastPkts RMON hostTimeOutBroadcastPkts EXT-MIB-II ifOutBroadcastPkts	The number of valid frames transmitted that are addressed to a broadcast address. This counter does not include errored broadcast packets or valid multicast packets.
24	XmtMultiCastPkts	RMON etherStatsMulticastPkts RMON etherHistoryMulticastPkts RMON hostOutMulticastPkts RMON hostTimeOutMulticastPkts EXT-MIB-II ifOutMulticastPkts	The number of valid frames transmitted that are addressed to a multicast address. This counter does not include errored multicast packets or valid broadcast packets.
25	XmtCollisions	RMON etherStatsCollisions RMON etherHistoryCollisions	The number of collisions that occur during transmission attempts. Collisions that occur while the device is not transmitting (i.e., receive collisions) are not identifiable and therefore not counted.

**Table 28. Transmit Statistics Counters (Continued)**

Addresses (hex)	Transmit Counter Name	MIB Object Supported	Description of Counter/Comments
26	XmtUnicastPkts	MIB-II ifOutUcastPkts	The number of valid frames transmitted that are not addressed to a multicast or a broadcast address. This counter does not include errored unicast packets.
27	XmtOneCollision	E-like dot3StatsSingleCollisionFrames	The number of packets successfully transmitted after experiencing one collision.
28	XmtMultipleCollision	E-like dot3StatsMultipleCollisionFrames	The number of packets successfully transmitted after experiencing more than one collision.
29	XmtDeferredTransmit	E-like dot3StatsDeferredTransmissions	The number of packets for which the first transmission attempt on the network is delayed because the medium is busy.
2a	XmtLateCollision	E-like dot3StatsLateCollisions	The number of late collisions that occur. A late collision is defined as a collision that occurs more than 512 bit times after the transmission starts. The 512-bit interval is measured from the start of preamble.
2b	XmtExcessiveDefer		The number of excessive deferrals that occur. An excessive deferral occurs when a transmission is deferred for more than 3036 byte times in normal mode or 3044 byte times in VLAN mode.
2c	XmtLossCarrier		The number of transmit attempts made when the LINK_STAT bit in the STAT0 register is 0.
2d	XmtExcessiveCollision	E-like dot3StatsExcessiveCollisions	The number of packets that are not transmitted because the packet experienced 16 unsuccessful transmission attempts (the first attempt plus 15 retries).
2e	XmtBackPressure		The total number of back pressure collisions generated.
2f	XmtFlowCtrl	PAUSEMACCtrlFramesTransmitted	The total number of PAUSE packets generated and transmitted by the controller hardware.
30	XmtPkts64Octets	RMON etherStatsPkts64Octets	The total number of packets transmitted (excluding error packets) that are 64 bytes long.

**Table 28. Transmit Statistics Counters (Continued)**

Addresses (hex)	Transmit Counter Name	MIB Object Supported	Description of Counter/Comments
31	XmtPkts65to127Octets	RMON etherStatsPkts65to127Octets	The total number of packets transmitted (excluding error packets) that are 65 bytes to 127 bytes long, inclusive.
32	XmtPkts128to255Octets	RMON etherStatsPkts128to255Octets	The total number of packets transmitted (excluding error packets) that are 128 bytes to 255 bytes long, inclusive.
33	XmtPkts256to511Octets	RMON etherStatsPkts256to511Octets	The total number of packets transmitted (excluding error packets) that are 256 bytes to 511 bytes long, inclusive.
34	XmtPkts512to1023Octets	RMON etherStatsPkts512to1023Octets	The total number of packets transmitted (excluding error packets) that are 512 bytes to 1023 bytes long, inclusive.
35	XmtPkts1024to1518Octets	RMON etherStatsPkts1024to1518Octets	The total number of packets transmitted (excluding error packets) that are 1024 bytes to 1518 (1522 when VLAN set) bytes long, inclusive.
36	XmtOversizePkts		The total number of packets transmitted (excluding error packets) that are longer than the maximum valid frame size.
37	XmtJumboPkts		The total number of packets transmitted (including error packets) that are 1519 bytes to 9018 bytes long, inclusive when VSIZE=0 or 1523 to 9022 when VSIZE=1.

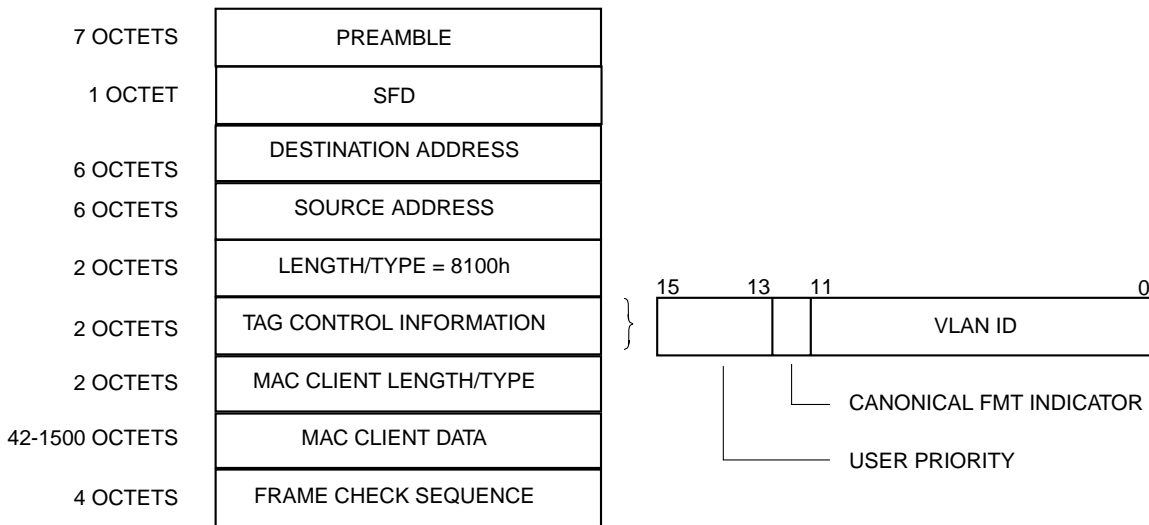
### 3.10.8 VLAN Support

Virtual Bridged Local Area Network (VLAN) tags are defined in IEEE standard 802.3ac-1998. A VLAN tag is a 4-byte quantity that is inserted between the Source Address field and the Length/Type field of a basic 802.3 MAC frame. The VLAN tag consists of a Length/Type field that contains the value 8100h and a 16-bit Tag Control Information (TCI) field. The TCI field is further divided into a 3-bit User Priority field, a 1-bit Canonical Format Indicator (CFI), and a 12-bit VLAN Identifier.

A frame that has no VLAN tag is said to be untagged. A frame with a VLAN tag whose VLAN Identifier field contains the value 0 is said to be priority-tagged. A frame with a VLAN tag with a non-zero VLAN Identifier field is said to be VLAN-tagged.

The format of a VLAN-tagged frame is shown in Figure 23 on page 105.





**Figure 23. VLAN-Tagged Frame Format**

The network controller includes several features that can simplify the processing of IEEE 802.3ac VLAN-tagged frames.

### 3.10.8.1 VLAN Frame Size

Since the VLAN tag contains 4 octets, in networks that allow VLAN tags, the maximum valid frame size is increased by 4 octets from 1518 to 1522 octets. In networks that support 9018-octet jumbo frames, adding VLAN capability increases the maximum valid frame size from 9018 to 9022 octets.

The maximum valid frame size is programmed using the VSIZE and JUMBO bits (CMD3, bits 20 and 21) as shown in Table 26 on page 98.

The maximum valid frame size is used for determining when to increment the XmtOversizePkts, XmtPkts1024to1518Octets, XmtExcessiveDefer, RcvJabbers, RcvAlignmentErrors, RcvFCSErrors, RcvPkts1024to1518Octets, and RcvOversizePkts MIB counters. The maximum frame size does not affect whether or not a frame is actually received.

### 3.10.8.2 Admit Only VLAN Frames Option

The Admit Only VLAN (VLONLY) bit in the Command3 Register can be programmed to reject any frame that is not VLAN-tagged. When VLONLY is set, untagged or priority-tagged frames are flushed from the receive FIFO and are not copied into system memory. Only frames with a Length/Type field equal to 8100h and a non-zero VLAN ID field are received. The VLAN ID field consists of bits [11:0] of the 15th and 16th bytes of the frame.

### 3.10.8.3 VLAN Tags in Descriptors

VLAN tag information can be passed between the host CPU and the network medium through Transmit or Receive Descriptors. The transmitter can be programmed to insert or delete a VLAN tag

or to modify the TCI field of a VLAN tag. This feature allows VLAN software to control the VLAN tag of a frame without modifying data in transmit buffers. The receiver can determine whether a frame is untagged, priority-tagged, or VLAN-tagged, and it can copy the TCI field of the VLAN tag into the Receive Descriptor and remove the tag from the frame.

The Tag Control Command (TCC) is a 2-bit field in the Transmit Descriptor that determines whether the transmitter will insert, delete, or modify a VLAN tag or transmit the data from the transmit buffers unaltered. The encoding of the TCC field is shown in Table 29 on page 106.

If the transmitter adds, deletes, or modifies a VLAN tag, it appends a valid FCS field to the frame, regardless of the state of the Disable Transmit FCS (DXMTFCS) bit (CMD2, bit 8).

The receiver examines each incoming frame and writes the frame's VLAN classification into the Tag Type (TT) field of the Receive Descriptor. If the frame contains a VLAN tag, the receiver copies the TCI field of tag into the TCI field of the Receive Descriptor. The encoding of the TT field is shown in Table 30. If the VL\_TAG\_DEL bit in CMD3 is set, the controller deletes the 4-byte tag from the frame as it copies the frame to host system memory.

**Table 29. VLAN Tag Control Command**

TCC (TMD2[17:16])	Action
00	Transmit data in buffer unaltered
01	Delete Tag Header
10	Insert Tag Header containing TCI field from descriptor.
11	Replace TCI field from buffer with TCI data from descriptor.

**Table 30. VLAN Tag Type**

TT (RMD1[19:18])	Description
00	Reserved
01	Frame is untagged
10	Frame is priority-tagged
11	Frame is VLAN-tagged

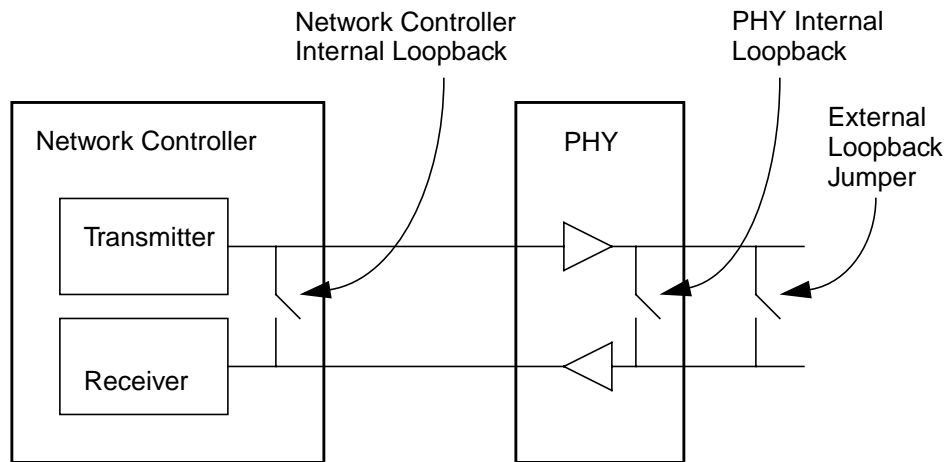
### 3.10.9 Loopback Operation

Loopback is a mode of operation intended for system diagnostics. In this mode, the transmitter and receiver are both operating at the same time so that the controller receives its own transmissions. The controller provides two basic types of loopback. In internal loopback mode, the transmitted data is looped back to the receiver inside the controller without actually transmitting any data to the external network. The receiver moves the received data to the next receive buffer, where it can be examined by

software. Alternatively, in external loopback mode, data can be transmitted to and received from the external network.

All transmit and receive function programming, such as automatic transmit padding and receive pad stripping, operates the same way in loopback as it does in normal operation.

The external loopback through the PHY interface bus requires a two-step operation. First the external PHY must be placed into a loopback mode. Then the network controller must be placed into the external loopback mode by setting EXLOOP (CMD2, bit 3). An external PHY with a standard MII can be placed into loopback mode by writing to the PHY Access Register. Alternatively for external loopback the PHY can be put into full-duplex mode, and an external set of jumpers can be used to connect the network receive path to the transmit path. Figure 24 on page 107 shows three possible loopback paths.



**Figure 24. Internal and External Loopback Paths**

The internal loopback is controlled by INLOOP (CMD2, bit 4). When set to 1, this bit causes the internal transmit data path to loop back to the internal receive data path. The RUN bit (CMD0, bit 0) must be cleared to 0 before the state of the INLOOP bit is changed.

The MII management port (MDC, MDIO) is unaffected by the INLOOP bit.

The internal MII interface is mapped in the following way:

- The TXD[3:0] nibble data path is looped back onto the RXD[3:0] nibble data path
- TX\_CLK and RX\_CLK are driven by the same internal clock so that no external TX\_CLK signal is needed.
- TX\_EN is looped back as RX\_DV.
- TX\_EN is ORed with the CRS pin and is looped back as CRS.
- The COL input is driven Low.

- TX\_ER is not driven by the network controller and therefore not looped back.

Internal loopback should not be used on a live network because transmit data path signals might interfere with network traffic. The PHY should be disconnected from the network or put into the “Isolate” state before internal loopback is used.

### 3.10.10 Full-Duplex Operation

When the network controller is in full-duplex mode, it can receive a frame from the network at the same time that it is transmitting a frame. Full-duplex operation is normally enabled through the auto-negotiation process. Alternatively full-duplex operation can be enabled by clearing the EN\_PMGR bit in CMD3 Register to 0 to disable the Port Manager and then setting the FORCE\_FULL\_DUPLEX bit (CMD3, bit 12).

In full-duplex mode the network controller behaves the same way it does in half-duplex mode with the following exceptions:

- Transmission is not deferred while receive is active.
- The IPG counter which governs transmit deferral during the IPG between back-to-back transmits is started when transmit activity for the first packet ends, instead of when transmit and carrier activity ends.
- The collision indication input to the MAC engine is ignored.

### 3.10.11 External PHY Interface

The network controller supports the Media Independent Interface (MII) as defined by the IEEE 802.3 standard. This Reconciliation Sublayer interface allows a variety of PHYs (100BASE-TX, 100BASE-FX, 100BASE-T4, 100BASE-T2, 10BASE-T, etc.) to be attached to the MAC engine without future upgrade problems. The MII interface is a 4-bit (nibble) wide data path interface that runs at 25 MHz for 100-Mbit/s networks or 2.5 MHz for 10-Mbit/s networks. The interface consists of two independent data paths, receive (RXD(3:0)) and transmit (TXD(3:0)), control signals for each data path (RX\_ER, RX\_DV, TX\_EN), network status signals (COL, CRS), clocks (RX\_CLK, TX\_CLK) for each data path, and a two-wire management interface (MDC and MDIO). See Figure 25 on page 109.

The transmit and receive paths in the controller's MAC are independent. The TX\_CLK and RX\_CLK need not run at the same frequency. TX\_CLK can slow down or stop without affecting receive and vice versa. It is only necessary to respect the minimum clock High and Low time specifications when switching TX\_CLK or RX\_CLK. This facilitates operation with PHYs that use MII signaling but do not adhere to 802.3 MII specifications.

#### 3.10.11.1 MII Transmit Interface

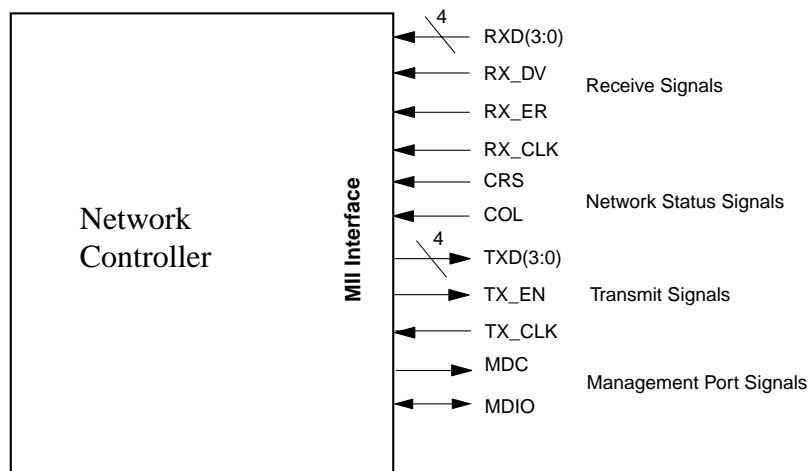
The MII transmit clock is generated by the external PHY and is sent to the network controller on the TX\_CLK input pin. The clock can run at 25 MHz or 2.5 MHz, depending on the speed of the network to which the external PHY is attached. The data is a nibble-wide (4 bits) data path, TXD(3:0), from

the network controller to the external PHY and is synchronous with the rising edge of TX\_CLK. The transmit process starts when the network controller asserts TX\_EN, which indicates to the external PHY that the data on TXD(3:0) is valid.

IEEE standard 802.3 provides a mechanism for signalling unrecoverable errors through the MII to the external PHY with the TX\_ER output pin. The external PHY responds to this error by generating a TX coding error on the current transmitted frame. The network controller does not use this method of signalling errors on the transmit side. Instead, if the network controller detects a transmit error, it inverts the FCS to generate an invalid FCS. Since the network controller does not implement the TX\_ER function, the TX\_ER pin on the external PHY device should be connected to VSS.

### 3.10.11.2 MII Receive Interface

The MII receive clock is also generated by the external PHY and is sent to the network controller on the RX\_CLK input pin. The clock can run at 25 MHz or 2.5 MHz, depending on the speed of the network to which the external PHY is attached.



**Figure 25. Media Independent Interface**

The receive process starts when RX\_DV is asserted. RX\_DV must remain asserted until the end of the receive frame. If the external PHY device detects errors in the currently received frame, it asserts the RX\_ER signal. RX\_ER can be used to signal special conditions *out of band* when RX\_DV is not asserted. Two defined out-of-band conditions for this are the 100BASE-TX signaling of *bad* Start of Frame Delimiter and the 100BASE-T4 indication of illegal code group before the receiver has synchronized with the incoming data. The network controller does not respond to these conditions. All *out of band* conditions are currently treated as NULL events. Certain *in-band* non-IEEE 802.3-compliant flow control sequences may cause erratic behavior for the network controller. Consult the switch/bridge/router/hub manual to disable the *in-band* flow control sequences if they are being used.

### 3.10.11.3 MII Network Status Interface

The MII also provides the CRS (Carrier Sense) and COL (Collision Sense) signals that are required for IEEE 802.3 operation. Carrier Sense is used to detect non-idle activity on the network for the purpose of inter-frame spacing timing in half-duplex mode. Collision Sense is used to indicate that simultaneous transmission has occurred in a half-duplex network.

### 3.10.11.4 MII Management Interface

The MII provides a two-wire management interface so that the network controller can control external PHY devices and receive status from them.

The network controller offers direct hardware support of the external PHY device without software intervention. The device automatically uses the MII Management Interface to read auto-negotiation information from the external PHY device and configures the MAC accordingly. The controller also provides the host CPU indirect access to the external PHY through the PHY Access Register.

With software support the network controller can support up to 31 external PHYs attached to the MII Management Interface.

Two independent state machines use the MII Management Interface to poll external PHY devices: the Network Port Manager and the Auto-poll State Machine. The Network Port Manager coordinates the auto-negotiation process, while the Auto-poll State Machine interrupts the host CPU when it detects changes in user-selected PHY registers.

The Network Port Manager sends a management frame to the default PHY about once every 600 ms to determine auto-negotiation results and the current link status. The Network Port Manager uses the auto-negotiation results to set the MAC's speed, duplex mode, and flow control ability. Changes detected by the Network Port Manager affect the operation of the MAC and MIB counters. For example, if link failure is detected, the transmitter increments the XmtLossCarrier counter each time it attempts to transmit a frame.

The Auto-poll State Machine periodically sends management frames to poll the status register of the default PHY device plus up to 5 user-selected PHY registers and interrupts the host processor if it detects a change in any of these registers. The Auto-poll State Machine does not change the state of the MAC engine.

### 3.10.11.5 MII Management Frames

The format of an MII Management Frame is defined in Clause 22 of IEEE standard 802.3. The start of an MII Management Frame is a preamble of 32 ones that ensures that all of the external PHYs are synchronized on the same interface. (See Figure 26 on page 111.) Loss of synchronization is possible due to the *hot-plugging* capability of the exposed MII. The preamble can be suppressed as described below if the external PHY is designed to accept frames with no preamble.

Preamble 1111....1111	ST 01	OP 10 Rd 01 Wr	PHY Address	Register Address	TA Z0 Rd 10 Wr	Data	Idle Z
32 Bits	2 Bits	2 Bits	5 Bits	5 Bits	2 Bits	16 Bits	1 Bit

**Figure 26. Frame Format at the MII Interface Connection**

The preamble (if present) is followed by a start field (ST) and an operation field (OP). The operation field (OP) indicates whether the network controller is initiating a read or write operation. This field is followed by the external PHY address (PHYAD) and the register address (REGAD). The PHY address of 1Fh is reserved and should not be used.

The register address field is followed by a bus turnaround field. During a read operation, the bus turnaround field is used to determine if the external PHY is responding correctly to the read request or not. The network controller floats the MDIO for both MDC cycles.

During the second cycle of a read operation, if the external PHY is synchronized to the network controller, the external PHY drives a 0. If the external PHY does not drive a 0, the network controller signals a MREINT (INT0, bit 16) interrupt, if MREINTEN (INTEN0, bit 16) is set to a 1. This interrupt indicates that the network controller had an MII management frame read error and that the data read is not valid.

During a write access the network controller drives a 1 for the first bit time of the turnaround field and a 0 for the second bit time.

After the Turn Around field comes the data field. For a write access the network controller fills this field with data to be written to the PHY device. For a read access the external PHY device fills this field with data from the selected register.

The last field of the MII Management Frame is an IDLE field that is necessary to give ample time for drivers to turn off before the next access.

MII management frames transmitted through the MDIO pin are synchronized with the rising edge of the Management Data Clock (MDC). The network controller drives the MDC to 0 and three-states the MDIO any time the MII Management Port is not active.

To help to speed up the reading and writing of the MII management frames to the external PHY, the MDC can be sped up to 10 MHz by setting the FMDC bits in CTRL2. The IEEE 802.3 specification requires use of the 2.5-MHz clock rate, but 5 MHz and 10 MHz are available for the user. The intended applications are that the 10-MHz clock rate can be used for a single external PHY on an adapter card or motherboard. The 5-MHz clock rate can be used for an exposed MII with one external PHY attached. The 2.5-MHz clock rate is intended to be used when multiple external PHYs are connected to the MII Management Port or if compliance to the IEEE 802.3 standard is required.

### 3.10.11.6 Host CPU Access to External PHY

The host CPU can indirectly read and write external PHY registers through the PHY Access Register.

To write to a PHY register the host CPU puts the register data into the PHY\_DATA field of the PHY Access Register, specifies the address of the external PHY device in the PHY\_ADDR field and the PHY register number in the PHY\_REG\_ADDR field, and sets the PHY\_WR\_CMD bit.

To read from a PHY register the host CPU specifies the address of the external PHY device in the PHY\_ADDR field and the PHY register number in the PHY\_REG\_ADDR field of the PHY Access Register and sets the PHY\_RD\_CMD bit. The host CPU can then poll the register until the PHY\_CMD\_ACTIVE bit is 0, or it can wait for the MII Management Command Complete Interrupt (MCCINT in the Int0 Register). When the PHY\_CMD\_ACTIVE bit is 0, the PHY\_DATA field contains the data read from the specified external PHY register. If an error occurs in the read operation, the PHY\_RD\_ERR bit in the PHY Access Register and the MII Management Read Error Interrupt (MREINT) bit in the Interrupt0 Register are set, and if the corresponding enable bit is set (MREINTEN in the Interrupt Enable Register), the host CPU is interrupted.

The host CPU must not attempt a second PHY register access until the first access is complete. When the access is complete, the PHY\_CMD\_ACTIVE bit in the PHY Access Register is cleared and the MII Management Command Complete Interrupt (MCCINT) bit in the Interrupt Register is set to 1, and if the corresponding enable bit is set, the host CPU is interrupted. The host can either wait for this interrupt, or it can use some other method to ensure that it waits for a long enough time. Note that with a 2.5 MHz MDC clock it takes about 27  $\mu$ s to transmit a management frame with a preamble. However, if the Auto-Poll or Port Manager state machines are active, there may be a delay in sending a host generated management frame while other frames are sent. Under these conditions, the host should always check for command completion.

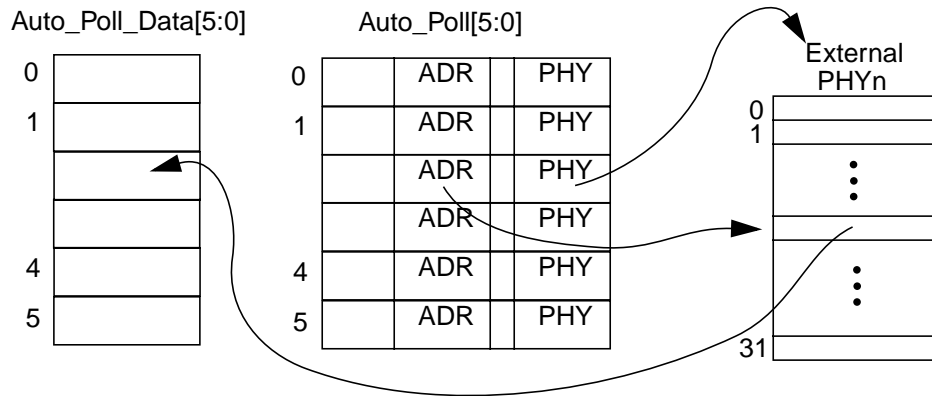
For an MII Management Frame transmitted as the result of a host CPU access to the PHY Access Register, preamble suppression is controlled by the Preamble Suppression bit (PHY\_PRE\_SUP) in the PHY Access Register. If this bit is set to 1 the preamble is suppressed. Otherwise, the frame includes a preamble. The host CPU should only set the Preamble Suppression bit when accessing a register in a PHY device that is known to be able to accept management frames without preambles. For PHY devices that comply with Clause 22 of IEEE standard 802.3, bit 6 of PHY Register 1 is fixed at 1 if the PHY accepts management frames with the preamble suppressed.

### 3.10.11.7 Auto-Poll State Machine

As defined in the IEEE 802.3 standard, the external PHY attached to the network controller's MII has no way of communicating important timely status information back to the network controller. Unless it polls the external PHY's status register, the network controller has no way of knowing that an external PHY has undergone a change in status. Although it is possible for the host CPU to poll registers in external PHY devices, the network controller simplifies this process by implementing an automatic polling function that periodically polls up to 6 user-selected PHY registers and interrupts the host CPU if the contents of any of these registers change.



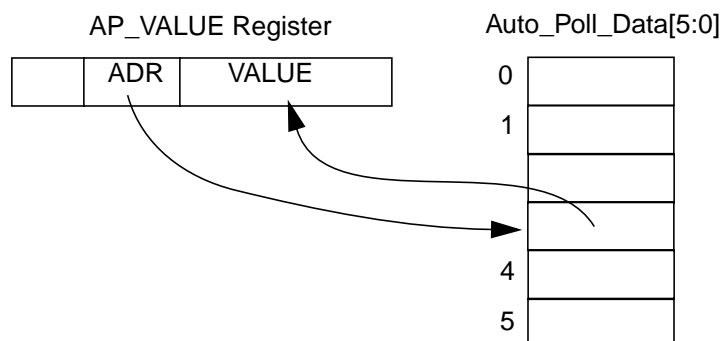
The automatic polling of PHY registers is controlled by six 16-bit Auto-Poll registers, AUTOPOLL0 to AUTOPOLL5 shown in Figure 27. By writing to the Auto-Poll registers, the user can independently define the PHY addresses and register numbers for six external PHY registers. The registers are not restricted to a single PHY device. In the Auto-Poll registers there is an enable bit for each of the selected PHY registers. When the host CPU sets the APEP bit to enable auto-polling, the Auto-Poll logic reads the PHY register corresponding to each enabled Auto-Poll register and stores the result in the corresponding Auto-Poll Data Register. (There is one Auto-Poll Data register for each of the six PHY registers.) Thereafter, at each polling interval, the Auto-Poll logic compares the current contents of the selected PHY register with the corresponding Auto-Poll Data Register. If it detects a change, it sets the MII Management Auto-Poll Interrupt (APINT) in the Interrupt Register, which causes an interrupt to the host CPU (if that interrupt is enabled).



**Figure 27. Auto-Poll and Auto-Poll Data Registers**

When the contents of one of the selected PHY registers changes, the corresponding Auto-Poll Data Register is updated so that another interrupt can occur when the data changes again.

The Auto-Poll Data Registers are accessed indirectly through the AP\_VALUE Register as shown in Figure 28. The host CPU first writes the number (0-5) of the desired Auto-Poll Data Register to the AP\_VAL\_ADDR field and sets the AP\_VAL\_RD\_CMD bit to 1. Setting AP\_VAL\_RD\_CMD to 1 starts the read cycle and automatically sets the AP\_CMD\_ACTIVE bit. The host CPU then polls the AP\_CMD\_ACTIVE bit until this bit is 0. At this time the AP\_VAL field of the AP\_VALUE register contains the contents of the selected Auto-Poll Data Register. The host CPU must not write to the AP\_VALUE Register while the AP\_CMD\_ACTIVE bit has the value 1.



**Figure 28. Indirect Access to Auto-Poll Data Registers**

Auto-Poll Register 0 differs from the other Auto-Poll Registers in several ways. The PHY address (AP\_PHY0\_ADDR) field of this register defines the default PHY address that is used by both the Auto-Poll State Machine and the Network Port Manager. The register number field is fixed at 1 (which corresponds to the external PHY status register), and the register is always enabled. This means that if the Auto-Poll State Machine is enabled, it always polls register 1 of the default PHY and interrupts the host CPU when it detects a change in that register.

In addition to the PHY address, register number, and enable bit, the Auto-Poll Registers contain two other control bits for each of the 5 user-selected registers. These bits are the Preamble Suppression (AP\_PRE\_SUP) and Default PHY (AP\_DFLT\_PHY) bits.

If the Preamble Suppression bit is set, the Auto-Poll sends management frames to the corresponding register with no preamble field. The host CPU should only set the Preamble Suppression bit for registers in PHY devices that are known to be able to accept management frames without preambles. For PHY devices that comply with Clause 22 of IEEE standard 802.3, bit 6 of PHY register 1 is fixed at 1 if the PHY accepts management frames with the preamble suppressed.

If the Default PHY bit (AP\_PHYx\_DFLT) is set, the corresponding Preamble Suppression bit and PHY address field are ignored. In this case the Auto-Poll State Machine uses the default PHY address from the AP\_PHY0\_ADDR field, and suppresses the preamble if the Network Port Manager logic has determined that the default PHY device accepts management frames with no preamble. If the Network Port Manager logic has not determined that the default PHY device accepts management frames with no preamble, the Auto-Poll State Machine does not suppress the preamble when accessing the selected register.

The Auto-Poll State Machine is enabled when the Auto-Poll External PHY (APEP) bit (CMD3, bit 10) is set to 1. If APEP is cleared to 0, the Auto-Poll machine does not poll any PHY registers regardless of the state of the enable bits in the Auto-Poll registers. The APEP bit has no effect on the Network Port Manager, which may poll the default PHY even when the state of the APEP bit is 0.

The host CPU must ensure that the APEP bit is cleared to 0 before it changes the contents of any of the Auto-Poll registers.

The Auto-Poll's frequency of generating MII management frames can be adjusted by setting of the APDW bits (CTRL2, bits 2-0).

### 3.10.12 Network Port Manager

The network controller does not require software intervention to control and configure an external PHY attached to the MII. The controller contains an automatic configuration system called the Network Port Manager.

The Network Port Manager initiates auto-negotiation in the external PHY when necessary and monitors the results. When auto-negotiation is complete, the Network Port Manager sets up the MAC to be consistent with the negotiated configuration. The Network Port Manager auto-negotiation sequence requires that the external PHY respond to the auto-negotiation request within 7.5 seconds. Otherwise, system software is required to properly control and configure the external PHY attached to the MII. After auto negotiation is complete, the Network Port Manager generates MII management frames about once every 600 ms to monitor the status of the external PHY.

The Network Port Manager is enabled when the Enable Port Manager (EN\_PMGR) bit (CMD3, bit 14) is set to 1.

Although the Network Port Manager and the Auto-poll State Machine both poll the external PHY, the two modules have different functions. The Port Manager manages auto-negotiation, while the Auto-poll State Machine monitors user-selected external PHY registers.

### 3.10.13 Auto-Negotiation Support

The external PHY and its link partner may have one or more of the following capabilities: 100BASE-T4, 100BASE-TX Full-/Half-Duplex, 10BASE-T Full-/Half-Duplex, and MAC Control PAUSE frame processing. During the auto-negotiation process the two PHY devices exchange information about their capabilities and then select the best mode of operation that is common to both devices. The modes of operation are prioritized according to the order shown in Table 31 on page 115 (with the highest priority shown at the top of the table).

**Table 31. Auto-Negotiation Capabilities**

Network Speed	Physical Network Type
200 Mbit/s	100BASE-X, Full Duplex
100 Mbit/s	100BASE-T4, Half Duplex
100 Mbit/s	100BASE-X, Half Duplex
20 Mbit/s	10BASE-T, Full Duplex
10 Mbit/s	10BASE-T, Half Duplex

Auto-Negotiation goes further by providing a message-based communication scheme called, Next Pages, before connecting to the Link Partner. The Network Port Manager does not support this feature. However, the host CPU can disable the Network Port Manager and manage Next Pages by accessing the PHY device through the PHY Access Register. The host CPU can disable the Network Port Manager by clearing the Enable Port Manager (EN\_PMGR) bit (CMD3, bit 14) to 0.

To control the auto-negotiation process, the Network Port Manager generates MII Management Frames to execute the procedure described below.

The Network Port Manager is held in the IDLE state while H\_RESET is asserted and while the EN\_PMGR bit is cleared. When none of these conditions are true, the Network Port Manager proceeds through the following steps:

1. If XPHYRST is set, write to the PHY's Control Register (R0) to set the Soft Reset bit and cause the PHY to reset. The Network Port Manager then periodically reads the PHY's Control Register (R0) until the reset is complete.
2. If XPHYRST is not set or after the PHY reset is complete, the PHY's Status Register (R1) is read.
3. If the PHY's Auto-Negotiation Ability bit (R1, bit 3) is 0 or if the XPHYANE bit in the Control2 Register is 0 and if this is the first pass through this procedure since EN\_PMGR was set to 1, write to the PHY's Control Register (R0) to disable auto-negotiation and set the speed and duplex mode to the values specified by the XPHYSP and XPHYFD bits in the Control2 Register ANDed with the appropriate bits from the PHY's Technology Ability Field. Then proceed to step 8.
4. If this is the first pass through this procedure since EN\_PMGR was set to 1, write to the Auto-Negotiation Advertisement Register (R4). Bits A0 to A4 of the Technology Ability field of R4 are taken from bits 15 to 11 in R1. (The Technology Ability field of R4 consists of bits 5 through 12, where bit 5 of R4 corresponds to Technology Ability bit A0.) Bit A5 of the Technology Ability field indicates the MAC's ability to respond to MAC Control Pause frames. This bit is set equal to the value of the Negotiate Pause Ability (NPA) bit in the Flow Control Register. Bit A6 indicates Asymmetric PAUSE operation for full-duplex links. This bit is copied from the Negotiate Asymmetric Pause Ability (NAPA) bit in the Flow Control Register. The Next Page, Acknowledge, and Remote Fault bits are cleared to 0, and the Selector Field is set to 00001 to indicate IEEE standard 802.3.
5. If this is the first pass through this procedure since EN\_PMGR was set to 1, write to the Control Register (R0) to restart Auto-negotiation.
6. Poll R1 at 1.9 second intervals until the Auto-Negotiation Complete bit is set to 1. If auto-negotiation is not complete after four polling intervals, go back to step 1.
7. Read the Auto-Negotiation Link Partner Ability Register (R5). Set the MAC's speed, duplex mode, and pause ability to the highest priority mode that is common to both PHY devices.
8. Poll R1 at 600 ms intervals until the Link Status bit is 1. If Link Status is not found to be 1 after four polling intervals, go back to step 1.
9. Poll R1 at intervals of about 600 ms until the Link Status bit is 0. Go to step 1.

**Table 32. Sources of Auto-negotiation Advertisement Register (R4) Bits**

Bits	1000BASE-T Function	Source of data
4.15	Next Page	R6.2
4.14	Reserved	Cleared to 0

**Table 32. Sources of Auto-negotiation Advertisement Register (R4) Bits**

4.13	Remote Fault	Cleared to 0
4.12	Reserved	Cleared to 0
4.11	ASM_DIR	NAPA
4.10	PAUSE	NPA
4.9	100BASE-T4	R1.15
4.8	100BASE-TX, Full Duplex	R1.14
4.7	100BASE-TX	R1.13
4.6	10BASE-T, Full Duplex	R1.12
4.5	10BASE-T	R1.11
4.4:0	Selector Field	00001b

**Note:** The notation *Rx.y* stands for *PHY Management Register x, bit y*.

When Auto-Negotiation is complete, the Network Port Manager examines the MF Preamble Suppression bit in PHY register 1. If this bit is set, the Network Port Manager suppresses preambles on all frames that it sends until one of the following events occurs:

- A hardware reset occurs.
- The EN\_PMGR bit (CMD3 Register, bit 14) is cleared.
- A management frame read error occurs.
- The external PHY is disconnected.

The Network Port Manager is not disabled when the MDIO pin is held Low when the MII Management Interface is idle. If no PHY is connected and an external pull-down resistor is attached, reads of the external PHY's registers return all zeros, and no read error is reported. If a PHY is disconnected, causing MDIO to go Low, an MIIPD interrupt occurs.

### 3.10.13.1 Auto-Negotiation With Multiple PHY Devices

The MII Management Interface (MDC and MDIO) can be used to manage more than one external PHY device. The external PHY devices may or may not be connected to the network controller's MII bus. For example, two PHY devices can be connected to the network controller's MII bus so that the MAC can communicate over either a twisted-pair cable or a fiber-optic link. Conversely, several network controllers may share a single integrated circuit that contains several PHY devices with separate MII buses but with only one MII Management bus. In this case, the MII Management Interface of one network controller could be used to manage PHY devices connected to different network controllers.

If more than one PHY device is connected to the MII bus, only one PHY device is allowed to be enabled at any one time. Since the Network Port Manager can not detect the presence of more than one PHY on the MII bus, the host CPU is responsible for making sure that only one PHY is enabled. The host CPU can use the PHY Access Register to set the Isolate bit in the Control Register (Register 0, bit 10) of any PHY that needs to be disabled.

### 3.10.13.2 Operation Without MII Management Interface

The Port Manager normally sets up the speed, duplex mode, and flow control (pause) ability of the MAC based on the results of auto-negotiation. However, it is possible to operate the device with no MII Management Interface connection, in which case the Port Manager is not able to start the auto-negotiation process or set up the MAC based on auto-negotiation results. This may happen if the network controller is connected to a multi-PHY device that has only one MII Management Interface that is shared among several PHYs.

If the network controller is operating without a MII Management Interface connection to its external PHY, the host CPU can force the MAC into the desired state by clearing the EN\_PMGR bit in CMD3 Register to 0 to disable the Port Manager, then writing to the following bits:

1. FORCE\_FULL\_DUPLEX (CMD3, bit 12)
2. FORCE\_LINK\_STATUS (CMD3, bit 11)
3. Force Receive Pause Enable (FRPE, FLOW\_CONTROL, bit 21)
4. Force Transmit Pause Enable (FTPE, FLOW\_CONTROL, bit 22)

These bits set up the duplex mode, link status, and flow control ability in the MAC and put the MAC into the Link Pass state.

### 3.10.14 Regulating Network Traffic

The network controller provides two hardware mechanisms for regulating network traffic: 802.3x Flow Control and collision-based back pressure. 802.3x Flow Control applies to full-duplex operation only, while back pressure applies to half-duplex operation only. 802.3x Flow Control works by sending and receiving MAC Control PAUSE frames, which cause the receiving station to postpone transmissions for a time determined by the contents of the PAUSE frame. Back pressure forces collisions to occur when other nodes attempt to transmit, thereby preventing other nodes from transmitting for periods of times determined by the back-off algorithm.

The device includes support for two styles of full-duplex flow control, Fixed Length Pause flow control and Variable Length Pause flow control. In the Variable Length Pause style, which is similar to an XON-XOFF protocol, a pause frame whose request\_operand field (bytes 17 and 18) contains 0FFFFh is sent to prevent the link partner from transmitting. Later, a pause frame whose request\_operand field contains 0 is sent to allow the link partner to resume transmissions. The Variable Length Pause style of flow control is selected by clearing the Fixed Length Pause bit (FIXP, Flow Control Register, bit 18) to 0.

For the Fixed Length Pause style of flow control, a single pause frame is sent to halt transmissions for a predetermined period of time. The contents of the request\_operand field of this frame are taken from the Pause Length register. This style of flow control is selected by setting the Fixed Length Pause bit (FIXP) to 1.

### 3.10.14.1 MAC Control Pause Frames

The format of a MAC Control Pause frame is shown in Table 33.

**Table 33. MAC Control Pause Frame Format**

Octet Numbers	Field Name	Value
1-6	Destination Address	01-80-C2-00-00-01
7-12	Source Address	Sender's physical address
13-14	Length/Type	88-08
15-16	MAC Control Opcode	00-01
17-18	Request_operand	Pause time measured in pause_quanta, which are equal to 512 bit times
19-60	Pad	Zeros
61-64	FCS	FCS

When a network station that supports IEEE 802.3x Flow Control receives a pause frame, it must suspend transmissions after the end of any frame that was being transmitted when the pause frame arrived. The length of time for which the station must suspend transmissions is given in the request\_operand field of the pause frame. This pause time is given in units of pause\_quanta, where, one pause\_quantum is 512 bit times. The request\_operand field is interpreted as Big-Endian data—octet 17 is the most significant byte and octet 18 is the least significant byte.

### 3.10.14.2 Back Pressure

The network controller supports collision-based back pressure for congestion control when the device is operating in half-duplex mode.

When the MAC begins receiving a frame that passes the address matching criteria and if back pressure is enabled, the MAC intentionally causes a collision by transmitting a “phantom” frame. This phantom frame consists of a normal 7-byte preamble and 1-byte SFD followed by 63 bytes of alternating 1 and 0 bits. This frame should be interpreted as a runt frame with FCS error by any standard receiver.

Back pressure does not affect the transmission of a frame. The MAC only forces a collision when it begins to receive a new frame.

Back pressure is enabled when the BKPRS\_EN bit in CMD3 is set, the device is operating in half-duplex mode, and the receiver is congested. Receiver congestion is defined in the next section. The BKPRS\_EN bit allows the software to separate the decision about whether or not to apply back pressure from the decision about when to apply back pressure. If the BKPRS\_EN bit is set, back pressure is applied when the device is operating in half-duplex mode and the receiver is congested. If the BKPRS\_EN bit is 0, back pressure is never applied.

The generation of a Back-Pressure collision causes the XmtBackPressure MIB Counter to increment.

### 3.10.14.3 Enabling Traffic Regulation

MAC Control Pause frame generation or half-duplex back pressure assertion is controlled by the host CPU when it sets or clears the Flow Control Command (FCCMD) bit. When the FCCMD bit is set, the receiver is considered to be in the ‘congested’ state. When the FCCMD bit is cleared, the receiver is not considered to be congested.

If the controller is in the half-duplex mode and the BKPRS\_EN bit is set, back pressure is asserted as long as the receiver is in the congested state. If the controller is in the full-duplex mode, a MAC Control Pause frame is automatically transmitted when the receiver enters the congested state, and optionally a second Pause frame is transmitted when the receiver leaves the congested state. In half-duplex mode the network controller does not respond to received pause frames.

When the device is operating in full-duplex mode, the content of the Pause frames and the number of Pause frames transmitted are controlled by the Fixed Length Pause control bit (FIXP). When FIXP is set to 1, a single Pause frame is transmitted when the receiver enters the congested state. The contents of the request\_operand field of the Pause frame are taken from the Pause Length field of the Flow Control Register (PAUSE\_LEN). This frame causes the link partner to halt transmissions for a predetermined length of time that corresponds to the value read from the PAUSE\_LEN field.

If FIXP is 0, when the receiver enters the congested state, a Pause frame is transmitted with the request\_operand field set to FFFFh, so that the link partner stops transmitting for a very long time. When the receiver exits the congested state, a second Pause frame is transmitted with the request\_operand field cleared to 0000h, so that the link partner resumes transmissions immediately.

Traffic regulation is affected by the following:

- Duplex mode
- Flow Control Command (FCCMD) bit
- Fixed Length Pause (FIXP) bit
- 16-bit PAUSE Length field (PAUSE\_LEN)
- Negotiate Pause Ability (NPA) bit
- Negotiate Asymmetric Pause Ability (NAPA) bit
- Force Receive Pause Enable (FRPE) bit
- Force Transmit Pause Enable (FTPE) bit

### 3.10.14.4 Software Control of Traffic Regulation

The host CPU uses the Flow Control Command bit (FCCMD) to cause the controller to transmit flow control frames automatically or to enable back pressure.

In half-duplex mode, back pressure is enabled when FCCMD is set to 1, and it is disabled when FCCMD is cleared to 0.

In full-duplex mode, the act of setting FCCMD to 1 causes a pause frame to be sent. The contents of the request\_operand field of the frame depend on the state of the FIXP bit. If FIXP is 1, the contents



of the request\_operand field are copied from the PAUSE\_LEN field of the Flow Control register. If FIXP is 0, the contents of the request\_operand field are set to 0FFFFh.

In full-duplex mode, if FIXP is 0, the act of clearing FCCMD to 0 causes a pause frame to be sent with its request\_operand field cleared to 0.

If FIXP is set to 1, the FCCMD bit is self-clearing—the CPU does not have to write to the network controller to clear the FCCMD bit. This allows the CPU to use a single write access to cause a pause frame to be sent with a predetermined request\_operand field.

The effects of the FCCMD bit are summarized in Table 34.

**Table 34. FCCMD Bit Functions**

FCCMD Transition	FIXP	Duplex Mode	Action
0 to 1	X	Half	Enable back pressure
1 to 0	X	Half	Disable back pressure
0 to 1	1	Full	Send pause frame with request operand equal to the contents of the Pause Length register. Automatically clear FCCMD to 0.
1 to 0	1	Full	No action. (FCCMD is cleared automatically when FIXP = 1.)
0 to 1	0	Full	Send pause frame with request operand equal to 0FFFFh.
1 to 0	0	Full	Send pause frame with request operand equal to 0000h.

### 3.10.14.5 Programming the Pause Length

Before the host CPU changes the contents of the Pause Length field of the Flow Control register, it must make sure that a sufficient amount of time (about 50  $\mu$ s) has elapsed since the last time the register was updated. When the host CPU changes the value of this field it must also write a 1 to the PAUSE\_LEN\_CHG bit (FLOW\_CONTROL, bit 30). This bit is used internally to synchronize the change in Pause Length value with the transmission of flow control frames. The bit is automatically cleared to 0 after the update process has finished. After the internal logic has cleared this bit, it is safe to write a new value to the Pause Length field.

### 3.10.14.6 Enabling Receive Pause

The ability to respond to received pause frames, or receive pause ability, is controlled independently from the transmission of pause frames. When receive pause ability is enabled, the receipt of a pause frame causes the device to stop transmitting for a time period that is determined by the contents of the pause frame.

Receive pause ability is enabled either by auto-negotiation or by the Force Receive Pause Enable bit (FRPE, FLOW\_CONTROL, bit 21). If the FRPE bit is set, receive pause ability is enabled regardless of the results of auto-negotiation. If FRPE is not set, the pause ability is determined by auto-negotiation.

Clause 37 of IEEE standard 802.3, 1998 Edition defines two types of pause configurations, symmetric PAUSE and asymmetric PAUSE. In a symmetric configuration both link partners transmit and receive

PAUSE frames. In an asymmetric configuration only one link partner transmits PAUSE frames and the other one receives them. During the auto-negotiation process each link partner indicates its configuration preference in the form of PAUSE and ASM\_DIR bits that are sent in MII Management Frames. The encoding of these bits is shown in Table 35.

**Table 35. Pause Encoding**

PAUSE	ASM_DIR	Capability
0	0	No PAUSE
0	1	Asymmetric PAUSE toward link partner
1	0	Symmetric PAUSE
1	1	Both Symmetric PAUSE and Asymmetric PAUSE toward local device

During the auto-negotiation process the PHY device attached to the LAN Ethernet controller controller exchanges configuration information with the PHY device on the other end of the network link segment as described in IEEE standard 802.3, 1998 Edition, Clause 28. The value of the PAUSE bit that is sent in MII Management frames is copied from the Negotiate Pause Ability bit (NPA, FLOW\_CONTROL Register, bit 19), and the value of the ASM\_DIR bit is copied from the Negotiate Asymmetric Pause Ability bit (NAPA, FLOW\_CONTROL Register, bit 20).

After auto-negotiation has finished, receive pause is enabled if either

1. The NPA bit is set and link partner's PAUSE bit is also set or
2. The NPA and NAPA bits are both set and the link partner's ASM\_DIR bit is set.

After auto-negotiation has finished, transmit pause is enabled if either

1. The NPA bit is set and link partner's PAUSE bit is also set or
2. The NAPA bit is set and the link partner's PAUSE and ASM\_DIR bits are set.

### 3.10.14.7 PAUSE Frame Reception

A MAC Control PAUSE frame is any valid frame with the following:

- A destination address field value equal to the MAC's physical address or equal to the reserved multicast address 01-80-C2-00-00-01,
- A Length/Type field value equal to 88-08, and
- A MAC Control Opcode field value equal to 0001.

If such a frame is received while pause ability is enabled, the MAC device waits until the end of the frame currently being transmitted (if any) and then stops transmitting for a time equal to the value of the request\_operand field (octets 17 and 18) multiplied by 512-bit times.

If another MAC Control PAUSE frame is received before the Pause timer has timed out, the Pause timer is reloaded from the request\_operand field of the new frame so that the new frame overrides the earlier one.

Received MAC Control PAUSE frames are handled completely by the hardware. They are not passed on to the host computer. However, MAC Control frames with opcodes not equal to 0001h are treated as normal frames, except that their reception causes the Unsupported Opcodes counter to be incremented.

Since the host computer does not receive MAC Control PAUSE frames, 32-bit MIB counters have been added to record the following:

- MAC Control Frames received
- Unsupported Opcodes received
- PAUSE frames received

### 3.10.15 Delayed Interrupts

To reduce the host CPU interrupt service overhead the network controller can be programmed to postpone the interrupt to the host CPU until either a programmable number of receive or transmit interrupt events have occurred or a programmable amount of time has elapsed since the first interrupt event occurred. The use of the Delayed Interrupt Registers allows the interrupt service routine to process several events at one time without having to return control back to the operating system between events.

A receive interrupt event occurs when receive interrupts are enabled, and the network controller has completed the reception of a frame and has updated the frame's descriptors. A receive interrupt event causes the Receive Interrupt (RINT0) bit in the INT0 Register to be set if it is not already set. Similarly a transmit interrupt event occurs when transmit interrupts are enabled, and the network controller has copied a transmit frame's data to the transmit FIFO and has updated the frame's descriptors. A transmit interrupt event causes the appropriate Transmit Interrupt (TINTx) bit in the INT0 Register to be set if it is not already set. Note that frame receptions or transmissions affect the interrupt event counter only when the corresponding receive or transmit interrupts are enabled.

The network controller contains two Delayed Interrupt Registers that can be programmed to delay two groups of interrupts. Corresponding to each Delayed Interrupt Register is an Interrupt Group Register that selects which interrupt events are included in each group.

Each Interrupt Group Register has five Include bits, DLY\_INT\_x\_T[3:0] and DLY\_INT\_x\_R0, that correspond to transmit interrupts (TINT[3:0]) for each of 4 priority levels and the receive interrupt. The x in DLY\_INT\_x stands for A for group A interrupts and for B in group B interrupts. For each Include bit that is set in a particular Interrupt Group Register, the corresponding interrupt event is included in the interrupt group. Any interrupt event that is included in an interrupt group is delayed until the conditions programmed into the corresponding Delayed Interrupt Register are met.

Each Delayed Interrupt Register contains a 5-bit Event Count field and a 11-bit Maximum Delay Time field.

Each time the host CPU clears a RINT or TINT bit that corresponds to an interrupt event that is included in an interrupt group, the contents of the Event Count field of the corresponding Delayed Interrupt Register are loaded into an internal interrupt event counter, the contents of the Maximum

Delay Time field are loaded into an internal interrupt event timer, and the interrupt event timer is disabled. Each time a receive or transmit interrupt event that is included in that group occurs, the interrupt event counter is decremented by 1 and the interrupt event timer is enabled, or if it has already been enabled, it continues to count down. Once the interrupt event timer has been enabled, it decrements by 1 every 10 microseconds.

When either the interrupt event counter or the interrupt event timer reaches zero, the PIRQA\_L pin is asserted.

### 3.10.16 Power Management Support

The network controller supports power management as defined in the PCI Bus Power Management Interface Specification V1.1 and Network Device Class Power Management Reference Specification V1.0. These specifications define the network device power states, PCI power management interface including the Capabilities Data Structure and power management registers block definitions, power management events, and OnNow network Wake-up events.

The general scheme for the network controller power management is that when a wake-up event is detected, a signal (PME\_L) is generated to cause hardware external to the network controller to put the computer into the working (S0) mode. The controller supports three types of wake-up events:

- Magic Packet Frame Detect
- Link State Change
- Pattern Match Detect

All three wake-up events can cause wake-up from any power state including D3<sub>cold</sub> (PCI bus power off and clock stopped).

#### 3.10.16.1 OnNow Wake-Up Sequence

The system software enables the PME\_L signal by setting the PME\_EN bit in the PMCSR register (PCI configuration registers, offset 44h, bit 8) to 1. When a Wake-up event is detected, the network controller sets the PME\_STATUS bit in the PMCSR register (PCI configuration registers, offset 44h, bit 15). Setting this bit causes the PME\_L signal to be asserted.

Assertion of the PME\_L signal causes external hardware to wake up the CPU. The system software then reads the PMCSR register of every PCI device in the system to determine which device asserted the PME\_L signal.

When the software determines that the signal came from the network controller, it writes to the device's PMCSR to put the device into power state D0. The software then writes a 1 to the PME\_STATUS bit to clear the bit and turn off the PME\_L signal, and it calls the device's software driver to tell it that the device is now in state D0. The system software can clear the PME\_STATUS bit either before, after, or at the same time that it puts the device back into the D0 state.

The type of wake-up is configured by software using the bits LCMODE\_SW, PMAT\_MODE, and MPEN\_SW in the CMD7 register. These bits are only reset by the power-on reset (POR) so that they maintain their values across PCI bus resets.

### 3.10.16.2 Link Change Detect

Link change detect is one of the Wake-up events that is defined by the OnNow specification. Link Change Detect mode is set when the LCMODE\_SW bit (CMD7, bit 0) is set.

When this bit is set, any change in the Link status causes the LC\_DET bit (STAT0, bit 10) to be set. When the LC\_DET bit is set, the PME\_STATUS bit (PMCSR register, bit 15) is set. If the PME\_EN bit (PMCSR, bit 8) is set, then the PME\_L signal is also asserted.

### 3.10.16.3 Magic Packet™ Technology Mode

A Magic Packet frame is a frame that is addressed to the network controller and contains a data sequence made up of 16 consecutive copies of the device's physical address (PADR[47:0]) anywhere in its data field. The frame must also cause an address match. By default, it must be a physical address match, but if the MPPLBA bit (CMD3, bit 9) is set, logical and broadcast address matches are also accepted. Regardless of the setting of MPPLBA, the sequence in the data field of the frame must be 16 repetitions of the device's physical address (PADR[47:0]).

Magic Packet technology mode is enabled by setting the MPEN\_SW bit (CMD7, bit 1). If the RUN bit is set, the RX\_SPND bit must also be set to disable the normal receive mode. Magic Packet technology mode is disabled by clearing the enable bit.

When the network controller detects a Magic Packet frame, it sets the MP\_DET bit (STAT0, bit 11), the MPINT bit (INT0, bit 13), and the PME\_STATUS bit (PMCSR, bit 15). If the PME\_EN bit is set, the PME\_L signal is asserted as well. If INTREN (CMD0, bit 1) and MPINTEN (INTEN0, bit 13) are set to 1, PIRQA\_L is asserted.

The PCI bus interface clock (PCI\_CLK) is not required to be running while the device is operating in Magic Packet technology mode. Either of the PIRQA\_L, or the PME\_L signal may be used to indicate the receipt of a Magic Packet frame when the CLK is stopped.

#### 3.10.16.3.1 OnNow Pattern Match Mode

In the OnNow Pattern Match Mode, the network controller compares the incoming packets with up to eight patterns stored in the Pattern Match RAM (PMR). The stored patterns can be compared with part or all of incoming packets, depending on the pattern length and the way the PMR is programmed. When a pattern match has been detected, the PMAT\_DET bit (STAT0, bit 12) is set. This causes the PME\_STATUS bit (PMCSR, bit 15) to be set, which in turn asserts the PME\_L signal if the PME\_EN bit (PMCSR, bit 8) is set.

Pattern Match mode is enabled by setting the PMAT\_MODE bit (CMD7, bit 3). The RUN bit (CMD0, bit 0) and either the RX\_SPND bit (CMD0, bit 3) or the RX\_FAST\_SPND bit (CMD0, bit 5) must also be set before PMAT\_MODE is set.

The RUN or suspend bits must not be reset while Pattern Match mode is enabled.

The controller can be programmed to save the wake-up frame so that the software can respond to the frame that caused the CPU to wake up. This feature is enabled by setting the PMAT\_SAVE\_MATCH bit (CMD7, bit 4) together with the PMAT\_MODE bit.

The following procedure can be used to enter Pattern Match mode with PMAT\_SAVE\_MATCH enabled:

1. Set TX\_SPND and either RX\_SPND or RX\_FAST\_SPND.
2. Wait until RX\_SUSPENDED and TX\_SUSPENDED are set.
3. Clear RUN.
4. Set RX\_SPND, TX\_SPND, and RUN.
5. Set PMAT\_MODE and PMAT\_SAVE\_MATCH.

When the CPU wakes up, it can use the following procedure to restart the controller and retrieve the wake-up frame.

1. Read STAT0 to determine what caused the wake-up.
2. Clear PMAT\_MODE.
3. If wake-up was caused by a pattern match, set up a receive descriptor for the wake-up frame.
4. Clear RX\_SPND and TX\_SPND.
5. Set RDMD0.

### 3.10.16.3.2 Pattern Match RAM (PMR)

The PMR is organized as an array of 64 words by 40 bits as shown in Figure 29. The PMR is programmed indirectly through the PMAT0 and PMAT1 registers. Pattern Match mode must be disabled (PMAT\_MODE bit cleared) to allow reading or writing the PMR.

To write a 40-bit word to the PMR the host CPU executes the following steps:

1. Write bits 31:0 of the data to PMAT1.
2. Write to PMAT0. Set the PMR\_WR\_CMD bit. Put PMR data bits 39:32 into the PMR\_B4 field. Put the address to be accessed into the PMR\_ADDR field.

The sequence for reading a 40-bit word from the PMR is shown below:

1. Write to PMAT0. Set the PMR\_RD\_CMD bit. Put the address to be accessed into the PMR\_ADDR field.
2. Read PMAT0. The PMR\_B4 field contains the high-order byte of data that was read from the PMR.
3. Read data bits 31:0 from PMAT1.

The first two 40-bit words in the PMR serve as pointers and contain enable bits for the eight possible match patterns. The format of the first two words is shown in Table 36 on page 127.

**Table 36. Format of PMR Pointer Words**

Word	Bits	Description
0	0	Pattern 0 Enable. When this bit is 1, pattern 0 is compared with incoming data. When this bit is 0, pattern 0 is ignored.
	1	Pattern 1 Enable. When this bit is 1, pattern 1 is compared with incoming data. When this bit is 0, pattern 1 is ignored.
	2	Pattern 2 Enable. When this bit is 1, pattern 2 is compared with incoming data. When this bit is 0, pattern 2 is ignored.
	3	Pattern 3 Enable. When this bit is 1, pattern 3 is compared with incoming data. When this bit is 0, pattern 3 is ignored.
	4-7	Reserved. This field is ignored.
	8-15	Pattern 0 Pointer. This field contains the PMR address of the first word of pattern 0.
	16-23	Pattern 1 Pointer. This field contains the PMR address of the first word of pattern 1.
	24-31	Pattern 2 Pointer. This field contains the PMR address of the first word of pattern 2.
	32-39	Pattern 3 Pointer. This field contains the PMR address of the first word of pattern 3.
1	0	Pattern 4 Enable. When this bit is 1, pattern 4 is compared with incoming data. When this bit is 0, pattern 4 is ignored.
	1	Pattern 5 Enable. When this bit is 1, pattern 5 is compared with incoming data. When this bit is 0, pattern 5 is ignored.
	2	Pattern 6 Enable. When this bit is 1, pattern 6 is compared with incoming data. When this bit is 0, pattern 6 is ignored.
	3	Pattern 7 Enable. When this bit is 1, pattern 7 is compared with incoming data. When this bit is 0, pattern 7 is ignored.
	4-7	Reserved. This field is ignored.
	8-15	Pattern 4 Pointer. This field contains the PMR address of the first word of pattern 4.
	16-23	Pattern 5 Pointer. This field contains the PMR address of the first word of pattern 5.
	24-31	Pattern 6 Pointer. This field contains the PMR address of the first word of pattern 6.
	32-39	Pattern 7 Pointer. This field contains the PMR address of the first word of pattern 7.

The remainder of the RAM contains the match patterns and associated match pattern control bits. The format of words 2-63 is shown in Table 37.

**Table 37. Format of PMR Data Words**

Bits	Description
0-3	Mask. This bit mask determines which of the 4 bytes in this word are compared with frame data. Bits 0 to 3 correspond to bytes 1 to 4, respectively. A 1 in a bit position means that the corresponding byte is compared with frame data. A 0 in a bit position means that the corresponding byte is ignored in the comparison. For example, if the value of the mask field is 5, bytes 1 and 3 are compared with frame data and bytes 2 and 4 are ignored.

**Table 37. Format of PMR Data Words**

Bits	Description
4-6	Skip. The value of the SKIP field indicates the number of doublewords of frame data that are skipped before the pattern in this PMR word is compared with data from the incoming frame. A maximum of seven doublewords may be skipped.
7	End of Pattern (EOP). If this bit is set to 1, this word contains the last byte of the pattern.
8-39	Pattern Data. The order is such that if bits 8-15 correspond to byte n of the frame, then bits 32-39 correspond to byte n+3.

The contents of the PMR are not affected by any reset. The contents are undefined after a power-up reset (POR).

Future AMD Ethernet controllers may contain two Pattern Match RAMs, one for patterns 0-3 and one for patterns 4-7. To anticipate compatibility with these controllers, it is recommended that the software follow these rules.

1. Word 0 should be written with the PMR\_BANK bit (PMAT0, bit 28) cleared to 0. (In this controller the PMR\_BANK bit has no effect, but in other controllers it selects one of two Pattern Match RAMs.) Word 1 should be written with PMR\_BANK set to 1.
2. Data for patterns 0-3 should be written with PMR\_BANK cleared to 0; data for patterns 4-7 should be written with PMR\_BANK set to 1.

Figure 29 on page 129 shows the layout of the data in the Pattern Match RAM.





Directed IP packet (note that this excludes any other directed packets to our MAC address):

**Table 39. Directed IP Packet Example**

Offset (decimal)	Pattern (hex)	Comments
0	08003e304770	Destination MAC Address (Station Address)
12	0800	Protocol type (0800 = IP)
30	9d37c748	IP Address (157.55.199.72)

NBT Name Query/Registration for computer name <00>, <03>, <20>:

**Table 40. NBT Name Query/Registration Example**

Offset (decimal)	Pattern (hex)	Comments
12	0800	Protocol type (0800 = IP)
23	11	Protocol (11= UDP)
34	00890089	Port Number (NETBIOIS Name Service)
45	10	NetBIOS Flags (10 = Query OR Registration)
54	20	Name scope—NULL (limited to 32 bytes)
55	46 48 45 42 45 4C 45 46 46 43 43 41 43 41 43 41 43 41 43 41 43 41 43 41 43 41 43 41 43 41	Computer name—coded in half-ASCII ('WAKER')

The table below shows how these 3 patterns could be stored in the pattern RAM. In this example the third pattern was arbitrarily stored as pattern number 4, and its data was arbitrarily placed in the locations starting at PMR word 40.

Note that in the 4 columns containing the frame data, the byte in the leftmost column is closer to the start of the frame than the byte in the rightmost column, while the RAM data is shown with the most significant byte on the left.

**Table 41. PMR Programming Example**

Frame Offset	Offset + 0	Offset + 1	Offset + 2	Offset + 3	EOP	Skip	Mask	PMR Addr	PMR Data (Hex)
								0	00 00 06 02 03
								1	00 00 00 28 01
0	xx	xx	xx	xx					
4	xx	xx	xx	xx					
8	xx	xx	xx	xx					
12	08	06	xx	xx	0	3	0011	2	xx xx 06 08 33
16	xx	xx	xx	xx					
20	xx	01	xx	xx	0	1	0010	3	xx xx 01 xx 12
24	xx	xx	xx	xx					
28	xx	xx	xx	xx					

**Table 41. PMR Programming Example**

32	xx	xx	xx	xx					
36	xx	xx	9d	37	0	3	1100	4	37 9d xx xx 3c
40	c7	48	xx	xx	1	0	0011	5	xx xx 48 c7 83
0	08	00	3e	30	0	0	1111	6	30 3e 00 08 0f
4	47	70	xx	xx	0	0	0011	7	xx xx 70 47 03
8	xx	xx	xx	xx					
12	08	00	xx	xx	0	1	0011	8	xx xx 00 08 13
16	xx	xx	xx	xx					
20	xx	xx	xx	xx					
24	xx	xx	xx	xx					
28	xx	xx	9d	37	0	3	1100	9	37 9d xx xx 3c
32	c7	48	xx	xx	1	0	0011	10	xx xx 48 c7 83
0	xx	xx	xx	xx					
4	xx	xx	xx	xx					
8	xx	xx	xx	xx					
12	08	00	xx	xx	0	3	0011	40	xx xx 06 08 33
16	xx	xx	xx	xx					
20	xx	xx	xx	11	0	1	1000	41	11 xx xx xx 18
24	xx	xx	xx	xx					
28	xx	xx	xx	xx					
32	xx	xx	00	89	0	2	1100	42	89 00 xx xx 2c
36	00	89	xx	xx	0	0	0011	43	xx xx 89 00 03
40	xx	xx	xx	xx					
44	xx	10	xx	xx	0	1	0010	43	xx xx 10 xx021
48	xx	xx	xx	xx					
52	xx	xx	20	46	0	1	1100	44	46 20 xx xx 1c
56	48	45	42	45	0	0	1111	45	45 42 45 48 0f
60	4c	45	46	46	0	0	1111	46	46 46 45 4c 0f
64	43	43	41	43	0	0	1111	47	43 41 43 41 0f
68	41	43	41	43	0	0	1111	48	43 41 43 41 0f
72	41	43	41	43	0	0	1111	49	43 41 43 41 0f
76	41	43	41	43	0	0	1111	50	43 41 43 41 0f
80	41	43	41	43	0	0	1111	51	43 41 43 41 0f
84	41	xx	xx	xx	1	0	0001	52	xx xx xx 41 81

### 3.10.17 Reset

There are four different types of reset operations that may be performed on the network controller, H\_RESET, S\_RESET, Power-Up Reset, and External PHY Reset. The following is a description of each type of reset operation.

### 3.10.17.1 H\_RESET

Hardware Reset (H\_RESET) occurs when the RESET\_L signal is asserted. When the minimum pulse width timing as specified in the RESET\_L signal description has been satisfied, an internal reset operation is performed.

H\_RESET programs most of the internal registers to their default values. Note that there are several register fields that are undefined after H\_RESET. See the descriptions of the individual registers for details.

H\_RESET clears most of the registers in the PCI configuration space. H\_RESET resets the internal state machines.

To allow the LAN Ethernet controller to report wake-up events while operating in the D3cold state, the internal H\_RESET signal is blocked and no internal reset occurs when Dev1:0x44[PWRSTAT] is 3h. This allows wake-up logic to function correctly when RESET\_L is asserted in S3-5. An internal H\_RESET signal is automatically generated and a reset operation occurs when Dev1:0x44[PWRSTAT] is changed from 3h to any other value.

### 3.10.17.2 RUN Reset

A RUN reset is generated when the value of the RUN bit (CMD0, bit 0) is changed from 1 to 0.

RUN reset resets most of the bits in CMD0 and INT0.

RUN reset terminates all network activity abruptly and resets the internal state machines. The host can use the suspend mode (TX\_SPND and RX\_SPND in CMD0) to terminate all network activity in an orderly sequence before clearing the RUN bit.

### 3.10.17.3 Power Up Reset

Power Up reset is generated when power is first applied to the network controller or when clock to the controller is enabled (see DevB:3x64). The assertion of this signal generates a hardware reset (H\_RESET). In addition, it clears certain power management bits in PCI\_PMCSR, CMD7, and STAT0 that H\_RESET does not affect.

### 3.10.17.4 External PHY Reset

The PHY\_RST pin is intended to be connected to the reset input of an external PHY that does not have its own power on reset capability. The polarity of PHY\_RST is determined by the PHY\_RST\_POL bit in CMD3.

The host CPU can cause the PHY\_RST pin to be asserted by setting the RESET\_PHY or the RESET\_PHY\_PULSE pin in CMD3. The PHY\_RST pin remains asserted as long the RESET\_PHY bit has the value 1. If the RESET\_PHY\_PULSE bit is set, the PHY\_RST pin remains asserted for a time determined by the contents of the RESET\_PHY\_WIDTH field in CTRL1 (bits 23:16). RESET\_PHY\_PULSE is a read-only bit that does not have to be cleared. The RESET\_PHY and the RESET\_PHY\_PULSE bits should not be set at the same time.





## Chapter 4 Registers

---

### 4.1 Register Overview

The IC includes several sets of registers accessed through a variety of address spaces. I/O address space refers to register addresses that are accessed by x86 I/O instructions such as IN and OUT. PCI configuration space is typically accessed by PCI-defined I/O cycles to CF8h and CFCh in the host. There is also memory space and indexed address space in the IC.

#### 4.1.1 Configuration Space

The address space for PCI configuration registers is broken up into *buses*, *devices*, *functions*, and *offsets*, as defined by the PCI specification. The IC can be configured to include configuration space on two buses, the primary and secondary. The configuration space on the primary (host) bus is accessed by type 0 configuration cycles (as defined by the PCI and HyperTransport™ technology specifications). In HyperTransport technology mode, the device number is mapped into bits[15:11] of the configuration address. The function number is mapped into bits[10:8] of the configuration address. The offset is mapped to bits[7:2] of the configuration address. Type 1 configuration cycles on the host bus that are targeted to internal or external devices on the secondary PCI bus are converted to type 0 configuration cycles.

The IC is viewed by software as two PCI bus devices that sit on the primary bus, typically bus 0. The first device number—which matches the assigned base HyperTransport link UnitID number for the part—is the PCI bridge and the second is the LPC bridge. Since the USB and Ethernet controllers reside on the secondary PCI bus, their device numbers are hard wired to 0 through 1 on that bus. The remaining device numbers through 15 are available on the external bus, per the PCI-to-PCI bridge specification. Configuration cycles to device numbers 16 through 31 on the secondary PCI bus are generated with address bits[31:11] all zero so no IDSEL lines are set. The following diagram shows where all the functions reside in configuration space.

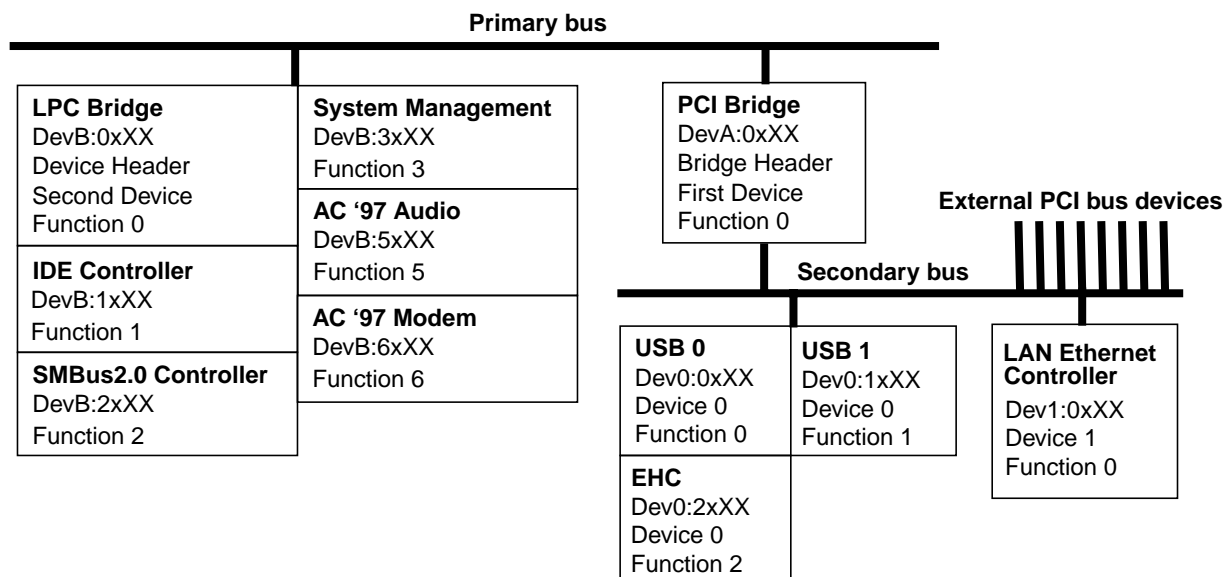


Figure 30. Configuration Space (Primary and Secondary Bus)

Configuration accesses to non-existent functions or functions disabled by DevB:0x48[PRIENS] within device A and B are not claimed by the IC. Configuration writes to non-existent registers within each enabled function are ignored and reads return all zeros.

### 4.1.2 Register Naming and Description Conventions

Each register location has an assigned mnemonic that specifies the address space and offset. These mnemonics start with two or three characters that identify the space followed by characters that identify the offset within the space. Register fields within register locations are also identified with a name or bit group in brackets following the register location mnemonic. For example, the ACPI sleep type register field, which is located at offset 04h of PMxx space, bits 10, 11, and 12, is referenced as PM04[SLP\_TYP] or PM04[12:10].

PCI configuration spaces are referenced with mnemonics that take the form of Dev[A|B|0|1|2|3]:[7:0]x[FF:0], where the first bracket contains the device number identifier, the second bracket contains the function number, and the last bracket contains the offset. Device numbers on the primary and secondary bus are assigned by the platform designer by programming DevA:0xC0[BUID].

Table 43. PCI Configuration Spaces

Bus	Device	Function	Mnemonic	Functionality
Primary	BUID	0	DevA:0xXX	PCI bridge
Primary	BUID+1	0	DevB:0xXX	LPC bridge, legacy circuitry
Primary	BUID+1	1	DevB:1xXX	Enhanced IDE controller



**Table 43. PCI Configuration Spaces**

Primary	BUID+1	2	DevB:2xXX	SMBus 2.0 controller
Primary	BUID+1	3	DevB:3xXX	System management registers
Primary	BUID+1	5	DevB:5xXX	AC '97 soft audio controller
Primary	BUID+1	6	DevB:6xXX	AC '97 soft modem controller
Secondary	0	0	Dev0:0xXX	OHCI-based USB controller 0
Secondary	0	1	Dev0:1xXX	OHCI-based USB controller 1
Secondary	0	2	Dev0:2xXX	EHCI-based USB controller
Secondary	1	0	Dev1:0xXX	LAN Ethernet controller

**Table 44. Fixed Address Spaces**

Port(s)	Mnemonic	Type	Function
00-0F	PORTxx	I/O mapped	Slave DMA controller
20-21	PORTxx	I/O mapped	Master interrupt controller
40-43	PORTxx	I/O mapped	Programmable interval timer
60	PORT60	I/O mapped	USB keyboard emulation address
61	PORT61	I/O mapped	AT compatibility Register
64	PORT64	I/O mapped	USB keyboard emulation address
70-73	RTCxx	I/O mapped	Real-time clock and CMOS RAM
80-8F	PORTxx	I/O mapped	DMA page registers
92	PORT92	I/O mapped	System control register
A0-A1	PORTxx	I/O mapped	Slave interrupt controller
C0-DF	PORTxx	I/O mapped	Master DMA controller
F0-F1	PORTxx	I/O mapped	Floating-point error control
170-177, 376	PORTxx	I/O mapped	Secondary IDE drives (not used when in native mode)
1F0-1F7, 3F6	PORTxx	I/O mapped	Primary IDE drives (not used when in native mode)
4D0-4D1	PORT4D0	I/O mapped	EISA-defined level-triggered interrupt control registers
CF9	PORTCF9	I/O mapped	Reset register
FEC0_00xx	IOAxx	Memory mapped	IOAPIC register set

**Table 45. Relocatable Address Spaces**

Address Specified By Configuration Register	Size (Bytes)	Type	Mnemonic	Function
DevB:0x74	256	I/O mapped	None	Pointer to 256 bytes of non volatile RAM
DevB:0xA0	1024	Memory mapped	HPETxx	High Precision Event Timer control registers
DevB:0xA8	32	Memory mapped	WDTxx	Watchdog Timer control registers
DevB:1x10 <sup>1</sup>	8	I/O mapped	None	Pointer to primary port IDE command space
DevB:1x14 <sup>1</sup>	4	I/O mapped	None	Pointer to primary port IDE control space
DevB:1x18 <sup>1</sup>	8	I/O mapped	None	Pointer to secondary port IDE command space
DevB:1x1C <sup>1</sup>	4	I/O mapped	None	Pointer to secondary port IDE control space
DevB:1x20	16	I/O mapped	IBMx	IDE controller bus-master control registers
DevB:2x10	32	I/O mapped	SCxx	SMBus controller command register space
DevB:3x58	256	I/O mapped	PMxx	System management I/O register space
DevB:5x10	256	I/O mapped	None	Pointer to AC '97 audio mixer space
DevB:5x14	64	I/O mapped	ACxx	AC '97 audio bus master control registers
DevB:6x10	256	I/O mapped	None	Pointer to AC '97 modem mixer space
DevB:6x14	64	I/O mapped	MCxx	AC '97 modem bus master control registers
Dev0:0x10	4K	Memory mapped	None	USB OHC control registers
Dev0:1x10	4K	Memory mapped	None	USB OHC control registers
Dev0:2x10	256	Memory mapped	ECAP	USB EHC capability registers
Dev0:2x14	256	Memory mapped	DBG	USB EHC debug port registers
Dev1:0x10	4K	Memory mapped	ENC	Ethernet controller memory space

**Notes:**

1. DevB:1x10, DevB:1x14, DevB:1x18, and DevB:1x1C are only used when the IDE controller is in native mode as specified by DevB:1x08.

The following are register behaviors found in the register descriptions.

**Table 46. Register Behavior Types (Read, Write, Etc.)**

Type	Description
Read or read-only	Capable of being read by software. Read-only implies that the register cannot be written to by software.
Write	Capable of being written by software.
Set by hardware	Register bit is set High by hardware.
Write 1 to clear	Software must write a 1b to the bit in order to clear it. Writing a 0b to these bits has no effect.
Write 1 only	Software can set the bit High by writing a 1b to it. However subsequent writes of 0b have no effect. RESET_L must be asserted in order to clear the bit.
Write once	After RESET_L, these registers may be written to once. After they are written, they become read-only until the next RESET_L assertion.

### 4.1.3 Positively- and Subtractively-Decoded Spaces

The IC positively decodes all address ranges described above as well as address windows for the secondary PCI bus, specified by the PCI bridge header. All transactions received that are not positively decoded are passed either directly to the LPC bridge or provided onto the secondary PCI

bus before being sent to the LPC bus (see DevB:0x40[SUBDEC]). Thus, the LPC bridge is the subtractive decode path for all unclaimed cycles.

## 4.2 PCI Bridge Configuration Registers (DevA:0xXX)

These registers are located in PCI configuration space on the primary PCI interface, in the first device (device A), function 0. See Section 4.1.2 on page 136 for a description of the register naming convention.

### PCI Bridge Vendor And Device ID Register

DevA:0x00

Default: 7460 1022h

Attribute: Read-only

Bits	Description
31:16	PCI Bridge Device ID
15:0	Vendor ID

### PCI Bridge Status And Command Register

DevA:0x04

Default: 0230 0000h.

Attribute: See below.

Bits	Description
31	<b>DPE. Detected Parity Error.</b> This bit is fixed in the Low state.
30	<b>SSE. Signaled System Error.</b> Read; set by hardware; write 1 to clear (however, software cannot get to this register until after reset). 1=A system error was signaled to the host (the outgoing HyperTransport™ link was flooded with sync packets) as a result of: <ul style="list-style-type: none"> <li>• A CRC error (see DevA:0xC4[CRCFEN, CRCERR]) or</li> <li>• A discard timer error (see DevA:0x3C[DTSERREN, DTSTAT]) or</li> <li>• An address parity error of a transaction targetted to the PCI bridge (see DevA:0x3C[SERREN]) or</li> <li>• SERR_L assertion (see DevA:0x3C[SERREN]).</li> </ul> Note: This bit is cleared by PWROK reset but not by RESET_L.
29	<b>RMA. Received Master Abort.</b> Read; set by hardware; write 1 to clear. 1=A request sent to the host bus received a master abort (an NXA error response). Note: this bit is cleared by PWROK reset but not by RESET_L.
28	<b>RTA. Received Target Abort.</b> Read; set by hardware; write 1 to clear. 1=A request sent to the host bus received a target abort (a non-NXA error response). Note: this bit is cleared by PWROK reset but not by RESET_L.
27	<b>Signaled Target Abort.</b> Read-only. This bit is fixed in the Low state.
26:25	<b>DEVSEL Timing.</b> Read-only. These bits are fixed at STATUS[10:9] = 01b. This specifies “medium” timing as defined by the PCI specification.
24	<b>Data Parity Detected.</b> Read-only. This bit is fixed in the Low state.
23	<b>Fast Back-to-Back Enable.</b> Read-only. This bit is fixed in the Low state.
22	<b>User Definable Features.</b> Read-only. This bit is fixed in the Low state.

Bits	Description (Continued)
21	<b>66 MHz Capable.</b> Read-only. This bit is fixed in the High state.
20	<b>Capabilities Pointer.</b> Read-only. This bit is fixed in the High state.
19:9	Reserved
8	<b>SERREN. SERR_L Enable.</b> Read-write. See DevA:0x3C[SERREN].
7	Reserved.
6	<b>PERSP. Parity Error Response.</b> Read-write. This bit controls no hardware.
5	Reserved.
4	<b>MWIEN. Memory Write and Invalidate Enable.</b> Read-write. This bit does not control any internal hardware.
3	<b>Special Cycle Enable.</b> Read-only. This bit is hardwired Low.
2	<b>MASEN. PCI Master Enable.</b> Read-write. 1=Enables internal and external PCI secondary bus masters to initiate cycles to the host.
1	<b>MEMEN. Memory Enable.</b> Read-write. 1=Enables access to the secondary PCI bus memory space.
0	<b>IOPEN. I/O Enable.</b> Read-write. 1=Enables access to the secondary PCI bus I/O space.

**PCI Bridge Revision And Class Code Register****DevA:0x08**

Default: 0604 00XXh see below.

Attribute: Read-only.

Bits	Description
31:8	<b>CLASSCODE.</b> Provides the bridge class code as defined in the PCI specification. This field is write accessible through DevA:0x60.
7:0	<b>REVISION.</b> PCI bridge silicon revision. The value of this register is revision-dependent.

**PCI Bridge BIST-Header-Latency-Cache Register****DevA:0x0C**

Default: 0001 0000h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. These bits fixed at their default values.
23:16	<b>HEADER.</b> Read-only. These bits fixed at their default values.
15:8	<b>LATENCY.</b> Read-write. These bits control no hardware.
7:0	<b>CACHE.</b> Read-only. These bits fixed at their default values.

**PCI Bridge Bus Numbers and Secondary Latency Register****DevA:0x18**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:24	<b>SECLAT.</b> Secondary latency timer.
23:16	<b>SUBBUS.</b> Subordinate bus number.
15:8	<b>SECBUS.</b> Secondary bus number.
7:0	<b>PRIBUS.</b> Primary bus number.

**PCI Bridge Memory Base-Limit Registers****DevA:0x1C, DevA:0x20 and DevA:0x24**

These registers specify the I/O-space (DevA:0x1C), non-prefetchable memory-space (DevA:0x20), and prefetchable memory-space (DevA:0x24) address windows for transactions that are mapped to the secondary PCI bus as follows:

```

PCI I/O window =
    {16'h0000, DevA:0x1C_IOLIM, 12'hFFF} >= address >=
    {16'h0000, DevA:0x1C_IOBASE, 12'h000};
PCI non-prefetchable memory window =
    {DevA:0x20_MEMLIM, 20'hF_FFFF} >= address >=
    {DevA:0x20_MEMBASE, 20'h0_0000};
PCI prefetchable memory window =
    {DevA:0x24_PMEMLIM, 20'hF_FFFF} >= address >=
    {DevA:0x24_PMEMBASE, 20'h0_0000};

```

These windows may also be altered by DevA:0x3C[VGAEN, ISAEN]. When the address (from either the host or from a secondary PCI bus master) is inside one of the windows, then the transaction is assumed to be intended for a target that sits on the secondary PCI bus. Therefore, the following transactions are possible:

- Host-initiated transactions inside the windows are sent to the PCI bus.
- Secondary PCI-initiated transactions inside the windows are not claimed by the IC.
- Host initiated transactions outside the windows that are not claimed by any other functions within the IC are passed to the LPC bus.
- Secondary PCI initiated transactions outside the windows are claimed by the IC using medium decoding and passed to the host.
- If IOBASE > IOLIM, MEMBASE > MEMLIM, and PMEMBASE > PMEMLIM, then *no* host-initiated transactions are forwarded by the secondary PCI bus and *all* secondary-PCI-bus-initiated memory and I/O (not configuration) transactions are forwarded to the host.

**DevA:0x1C**

Default: 0200 00F0h.

Attribute: See below.

Bits	Description
31	<b>DPE. Detected parity error.</b> Read; set by hardware; write 1 to clear. 1=The IC detected an address parity error as the target of a secondary PCI bus cycle or a data parity error as the target of a secondary PCI bus write cycle or a data parity error as the master of a secondary PCI bus read cycle.
30	<b>RSE. Received system error.</b> Read; set by hardware; write 1 to clear. 1=The IC detected assertion of SERR_L. Note: this bit is cleared by PWROK reset but not by RESET_L.
29	<b>RMA. Received master abort.</b> Read; set by hardware; write 1 to clear. 1=The IC received a master abort as a master on the secondary PCI bus. Note: this bit is cleared by PWROK reset but not by RESET_L.
28	<b>RTA. Received target abort.</b> Read; set by hardware; write 1 to clear. 1=The IC received a target abort as a master on the secondary PCI bus. Note: this bit is cleared by PWROK reset but not by RESET_L.

27	<b>STA. Signaled target abort.</b> Read; set by hardware; write 1 to clear. 1=The IC generated a target abort as a target on the secondary PCI bus. The IC generates a target abort if it receives a target abort (a non-NXA error) response from the host to a secondary PCI master transaction request. Note: this bit is cleared by PWROK reset but not by RESET_L.
26:25	<b>Device select timing.</b> Read-only. These bits are hard wired to indicate medium decoding.
24	<b>MDPE. Master data parity error.</b> Read; set by hardware; write 1 to clear. 1=The IC detected a parity error during a data phase of a read or detected PERR_L asserted during a write as a master on the secondary PCI bus and DevA:0x3C[PEREN] is set. When this bit is set, an NMI is generated; see PM48[NMI2SMI_EN] for information on how NMI interrupts may be controlled.
23:16	Reserved.
15:12	<b>IOLIM. I/O limit address bits.</b> [15:12]. See DevA:0x[24:1C] above.
11:8	Reserved.
7:4	<b>IOBASE. I/O base address bits.</b> [15:12]. See DevA:0x[24:1C] above.
3:0	Reserved.

**DevA:0x20**

Default: 0000 FFF0h.

Attribute: Read-write.

Bits	Description
31:20	<b>MEMLIM. Non-prefetchable memory limit address bits.</b> [31:20]. See DevA:0x[24:1C] above.
19:16	Reserved.
15:4	<b>MEMBASE. Non-prefetchable memory base address bits.</b> [31:20]. See DevA:0x[24:1C] above.
3:0	Reserved.

**DevA:0x24**

Default: 0000 FFF0h.

Attribute: Read-write.

Bits	Description
31:20	<b>PMEMLIM. Prefetchable memory limit address bits.</b> [31:20]. See DevA:0x[24:1C] above.
19:16	Reserved.
15:4	<b>PMEMBASE. Prefetchable memory base address bits.</b> [31:20]. See DevA:0x[24:1C] above.
3:0	Reserved.

**PCI Bridge Subsystem ID and Subsystem Vendor ID Register****DevA:0x2C**

Default: 0000 0000h.

Attribute: Read-only.

Bits	Description
31:16	<b>SSID. Subsystem ID.</b> This field is write accessible through DevA:0x70.
15:0	<b>SSVENDORID. Subsystem vendor ID.</b> This field is write accessible through DevA:0x70.

**HyperTransport™ Technology Capabilities Pointer Register****DevA:0x34**

Default: 0000 00C0h.

Attribute: Read-only.

Bits	Description
31:8	Reserved.
7:0	<b>CAPABILITIES_PTR.</b> Specifies the offset to standard HyperTransport technology registers.

**PCI Bridge Interrupt and Bridge Control Register****DevA:0x3C**

Default: 0000 00FFh.

Attribute: See below.

Bits	Description
31:28	Reserved.
27	<b>DTSERREN. Discard timer SERR_L enable.</b> Read-write. This bit specifies if a system error is issued on discard timer expiration. 1= Enable system error. 0= No error is flagged on discard timer expiration. Note: A system error is only flagged if DevA:0x04[SERREN] is set. This IC signals a system error by a sync flood on the HyperTransport™ link which also sets DevA:0x04[SSE].
26	<b>DTSTAT. Discard timer status.</b> Read; set by hardware; write 1 to clear. 1= The IC detected the expiration of the secondary discard timer. Note: this bit is cleared by PWROK reset but not by RESET_L.
25	<b>SECDTV. Secondary discard timer value.</b> Read-write. This bit specifies after how many clocks a received response for a delayed request gets discarded. 1= The discard timer counts 2 <sup>10</sup> PCI clock cycles. 0= The discard timer counts 2 <sup>15</sup> PCI clock cycles.
24:23	Reserved.
22	Read-write. This bit controls no hardware.
21	<b>MARSP. Master abort response.</b> Read-write. 1=The response to non-posted requests that come from the host bus or secondary PCI bus that results in a master abort indicates a target abort (through PCI bus protocol or HyperTransport link protocol). 0=Master aborts result in normal responses; read responses are sent with the appropriate amount of data, which are all 1's and writes are ignored.
20	Reserved.
19	<b>VGAEN. VGA decoding enable.</b> Read-write. 1=Route host-initiated commands targeting VGA-compatible address ranges to the secondary PCI bus. These include memory accesses from A0000h to BFFFFh, I/O accesses in which address bits[9:0] range from 3B0h to 3BBh or 3C0h to 3DFh (address bits[15:10] are not decoded, regardless of bit[18], ISA enable). 0=PCI does not decode VGA-compatible address ranges.
18	<b>ISAEN. ISA decoding enable.</b> Read-write. 1=The I/O address window specified by DevA:0x1C[15:0] is limited to the first 256 bytes of each 1K byte block specified. 0=The PCI I/O window is the whole range specified by DevA:0x1C[15:0].

Bits	Description (Continued)
17	<b>SERREN. System error enable.</b> Read-write. If DevA:0x04[SERREN] and DevA:0x3C[SERREN] are both High and if: <ul style="list-style-type: none"> <li>• An address parity error of a transactions targetted to the PCI bridge is detected, or</li> <li>• SERR_L is detected asserted (DevA:0x1C[RSE] = 1),</li> </ul> Then the IC responds by flooding the outgoing link with sync packets and setting DevA:0x04[SSE]. If either DevA:0x04[SERREN] or DevA:0x3C[SERREN] are Low, then neither of the two events above stops HyperTransport link operation or causes DevA:0x04[SSE] to be set.
16	<b>PEREN. Parity error response enable.</b> Read-write. 1=Enable parity error detection on secondary PCI interface (see DevA:0x1C[MDPE]); PERR_L signal enabled to set status bit or be driven. 0=DevA:0x1C[MDPE] cannot be set; PERR_L signal is ignored and it is not driven by the IC.
15:8	<b>INTERRUPT_PIN.</b> Read-only. These bits fixed in their default state.
7:0	<b>INTERRUPT_LINE.</b> Read-write. These bits control no internal logic.

**System Management Class Code Write Register****DevA:0x60**

Default: 0604 0000h.

Attribute: Read-write.

Bits	Description
31:8	<b>CCWRITE.</b> The value placed in this register is visible in DevA:0x08.
7:0	Reserved.

**Device and Subsystem ID Read-Write Register****DevA:0x70**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>SSID. Subsystem ID.</b> The value placed in this register is visible DevA:0x2C[31:16].
15:0	<b>SSVENDORID. Subsystem vendor ID.</b> The value placed in this register is visible DevA:0x2C[15:0].

**HyperTransport™ Technology Command Register****DevA:0xC0**

Default: 0080 F008h.

Attribute: See below.

Bits	Description
31:29	<b>Slave/primary interface type.</b> Read-only.
28	<b>Drop on Unitialized Link.</b> Read-Write. This bit controls no internal hardware. Note: this bit is cleared by PWROK reset but not by RESET_L.
27	<b>Default direction.</b> Read-only. This bit is hardwired Low to indicate that requests are sent to the master HyperTransport host bridge.
26	<b>Master host.</b> Read-only. This bit is hardwired Low to indicate that the master is connected to link 0.



25:21	<b>Unit ID count.</b> Read-only. Specifies the number of unit IDs used by the IC.
20:16	<b>BUID. Base UnitID.</b> Read-write. This specifies the HyperTransport™ protocol base unit ID. The IC's logic uses this value to determine the unit IDs for HyperTransport request and response packets.
15:8	<b>Capabilities Pointer.</b> Read-only. This register contains the pointer to the HyperTransport Interrupt Discovery and Configuration Capability.
7:0	<b>Capabilities ID.</b> Read-only. Specifies the capabilities ID for HyperTransport technology configuration space.

**HyperTransport™ Link Control Register****DevA:0xC4**

Default: ??00 0020h; see below for the default of bits[31:24].

Attribute: See below.

Bits	Description
31	Reserved.
30:28	<b>LWO. Link width out.</b> Read-write. Specifies the operating width of the outgoing HyperTransport™ link. Legal values are 000b (8 bits), 100b (2 bits), and 101b (4 bits). Writes of invalid values do not change the register content and are ignored. Note: this field is cleared by PWROK reset but not by RESET_L; the default value of this field depends on the widths of the HyperTransport links of the connecting device, as specified by the HyperTransport specification. Note: the link width does not change until either RESET_L is asserted or a HyperTransport link disconnect sequence occurs.
27	Reserved.
26:24	<b>LWI. Link width in.</b> Read-write. Specifies the operating width of the incoming HyperTransport link. Legal values are 000b (8 bits), 100b (2 bits), and 101b (4 bits). Writes of invalid values do not change the register content and are ignored. Note: this field is cleared by PWROK reset but not by RESET_L; the default value of this field depends on the widths of the HyperTransport links of the connecting device, as specified by the HyperTransport specification. Note: the link width does not change until either RESET_L is asserted or a HyperTransport link disconnect sequence occurs.
23	Reserved.
22:20	<b>Max link width out.</b> Read-only. This specifies the width of the outgoing HyperTransport link to be 8 bits wide.
19	Reserved.
18:16	<b>Max link width in.</b> Read-only. This specifies the width of the incoming HyperTransport link to be 8 bits wide.
15	Reserved.
14	<b>EXTCTL. Extended control time.</b> Read-write. This bit specifies the time in which LTXCTL_H/L is held asserted during the initialization sequence that follows a LDTSTOP_L deassertion, after LRXCTL_H/L is sampled asserted. 1= About 50 microseconds. 0= At least 16 bit times (for a 8 bit link). Note: this bit is cleared by PWROK reset but not by RESET_L.
13	<b>LDT3SEN. HyperTransport link three-state enable.</b> Read-write. 1=During the LDTSTOP_L disconnect sequence, the HyperTransport transmitter signals are placed into the high impedance state and the receivers are prepared for the high impedance mode. 0=During the LDTSTOP_L disconnect sequence, the HyperTransport transmitter signals are driven, but in an undefined state, the HyperTransport clock is toggling, and the HyperTransport receiver signals are assumed to be driven. Note: this bit is cleared by PWROK reset but not by RESET_L.
12:9	Reserved.
8	<b>CRCERR. CRC Error.</b> Read; set by hardware; write 1 to clear. 1=The hardware detected a CRC error on the incoming HyperTransport link. Note: this bit is cleared by PWROK reset but not by RESET_L.
7	<b>TXOFF. Transmitter off.</b> Read-only (not implemented).

Bits	Description (Continued)
6	<b>ENDOCH. End of chain.</b> Read-only (not implemented).
5	<b>INITCPLT. Initialization complete.</b> Read-only; set by hardware; cleared by RESET_L. This bit is set by hardware when low-level link initialization has successfully completed. If there is no device on the other end of the link, or if that device is unable to properly perform link initialization, the bit is not set.
4	<b>LKFAIL. HyperTransport link failure.</b> Read; set by hardware; write 1 to clear. This bit is set High by the hardware when a CRC error is detected on the link (if enabled by DevA:0xC4[CRCFEN]). Note: this bit is cleared by PWROK reset but not by RESET_L.
3	<b>CRCERRCMD. CRC error command.</b> Read-write. 1=The HyperTransport transmission logic generates erroneous CRC values. 0=Transmitted CRC values match the values calculated per the HyperTransport specification. This bit is intended to be used to check the CRC failure detection logic of the device on the other side of the link.
2	Reserved.
1	<b>CRCFEN. CRC flood enable.</b> Read-write. 1=CRC errors result in sync packets to the outgoing HyperTransport link and set the DevA:0xC4[LKFAIL] bit. 0=CRC errors do not result in sync packets and do not set the link fail bit.
0	Reserved.

**HyperTransport™ Non-Existent Link Control Register****DevA:0xC8**

Default: 0000 00D0h.

Attribute: Read-only.

Bits	Description
31:8	Reserved.
7	<b>Transmitter off.</b> This bit is hardwired High to indicate that there is no subordinate HyperTransport™ link.
6	<b>End of HyperTransport chain.</b> This bit is hardwired High to indicate that there is no subordinate HyperTransport link.
5	Reserved.
4	<b>Link failure.</b> This bit is hardwired High to indicate that there is no subordinate HyperTransport link.
3:0	Reserved.

**HyperTransport™ Technology Revision ID Register****DevA:0xCC**

Default: 0001 0022h.

Attribute: See below.

Bits	Description
31:16	<b>Link Frequency Capability.</b> Read-only. These bits indicate that the IC supports 200 MHz HyperTransport™ clock.
15:12	<b>Link Error.</b> Read-only. These bits are hard wired to the Low state.
11:8	<b>LINKFREQ. Link Frequency.</b> These bits control the frequency of the HyperTransport link. Legal value is 0000b (= 200 MHz). Writes of invalid values does not change the register content and are ignored. Note: this bit is cleared by PWROK reset, not by RESET_L. Note: after this field is updated, the link frequency does not change until either RESET_L is asserted or a disconnect sequence occurs.
7:0	<b>REVISION.</b> Read-only. The IC is designed to version 1.02 of the HyperTransport technology specification. Exceptions to the 1.02 HyperTransport technology specification are: - No CRC test support. - No support for memory accesses over 32 bits.

**HyperTransport™ Technology Feature Register****DevA:0xD0**

Default: 0000 0002h (see below).

Attribute: Read-only.

Bits	Description
31:2	Reserved.
1	<b>LSSUP. LDTSTOP_L supported.</b> This bit is High to indicate that the IC supports the LDTSTOP_L signal.
0	Reserved.

**HyperTransport™ Technology Enumeration Scratchpad Register****DevA:0xD4**

Default: 0000 0000h.

Attribute: See below.

Bits	Description
31:16	Reserved.
15:0	<b>ESP. Enumeration scratchpad.</b> Read-write. This field controls no hardware within the IC. Note: these bits are cleared by PWROK reset, not by RESET_L.

**HyperTransport™ Link PHY Compensation Control Registers****DevA:0x[E8, E4, E0]**

The HyperTransport™ PHY circuitry includes automatic compensation that is used to adjust the electrical characteristics for the HyperTransport transmitters and receivers. There is one compensation circuit for the receivers and one for each polarity of the transmitters. These registers provide visibility into the calculated output of the compensation circuits, the ability to override the calculated value with software-controlled values, and the ability to offset the calculated values with a fixed difference. These registers specify the compensation parameters as follows:

- DevA:0xE0: transmitter rising edge (P) drive strength compensation.
- DevA:0xE4: transmitter falling edge (N) drive strength compensation.
- DevA:0xE8: receiver impedance compensation.

For DevA:0x[E4, E0], higher values represent higher drive strength; the values range from 00h to 13h (20 steps). For DevA:0xE8, higher values represent lower impedance; the values range from 00h to 1Fh (32 steps).

**Note:** The default state of these registers is set by PWROK reset; assertion of RESET\_L does not alter any of the fields.

Default: See below.

Attribute: See below.

Bits	Description										
31	Must be Low. Read-write. This bit is required to be Low at all times; setting it High results in undefined behavior.										
30:21	Reserved.										
20:16	<b>CALCCOMP. Calculated compensation value.</b> Read-only. This provides the calculated value from the auto compensation circuitry. The default value of this field is not predictable.										
15:7	Reserved.										
6:5	<b>CTL. HyperTransport™ PHY control value.</b> Read-write. These two bits combine to specify the PHY compensation value as follows: <table border="0"> <tr> <td>CTL</td> <td>Description</td> </tr> <tr> <td>00b</td> <td>Apply CALCCOMP directly as the compensation value.</td> </tr> <tr> <td>01b</td> <td>Apply DATA directly as the compensation value.</td> </tr> <tr> <td>10b</td> <td>Apply the sum of CALCCOMP and DATA as the compensation value. In DevA:0x[E4, E0], if the sum exceeds 13h, then 13h is applied. In DevA:0x[E8], if the sum exceeds 1Fh, then 1Fh is applied.</td> </tr> <tr> <td>11b</td> <td>Apply the difference of CALCCOMP minus DATA as the compensation value. If the difference is less than 00h, then 00h is applied.</td> </tr> </table> <p>The default value of this field (from PWROK reset) is controlled by DevB:3x48[CMPOVR]. If CMPOVR = 0, the default is 00b. If CMPOVR = 1, the default is 01b.</p>	CTL	Description	00b	Apply CALCCOMP directly as the compensation value.	01b	Apply DATA directly as the compensation value.	10b	Apply the sum of CALCCOMP and DATA as the compensation value. In DevA:0x[E4, E0], if the sum exceeds 13h, then 13h is applied. In DevA:0x[E8], if the sum exceeds 1Fh, then 1Fh is applied.	11b	Apply the difference of CALCCOMP minus DATA as the compensation value. If the difference is less than 00h, then 00h is applied.
CTL	Description										
00b	Apply CALCCOMP directly as the compensation value.										
01b	Apply DATA directly as the compensation value.										
10b	Apply the sum of CALCCOMP and DATA as the compensation value. In DevA:0x[E4, E0], if the sum exceeds 13h, then 13h is applied. In DevA:0x[E8], if the sum exceeds 1Fh, then 1Fh is applied.										
11b	Apply the difference of CALCCOMP minus DATA as the compensation value. If the difference is less than 00h, then 00h is applied.										
4:0	<b>DATA. HyperTransport PHY data value.</b> Read-write. This value is applied to the HyperTransport PHY compensation as described in CTL. The default for DevA:0x[E4, E0] is 08h. The default for DevA:0xE8 is 0Fh.										

**Read-Write Register****DevA:0xEC**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	<b>RW. Read-write.</b> These bits are read-write accessible through software; they control no hardware.

**HyperTransport™ Technology Interrupt Discovery and Configuration Capability Register****DevA:0xF0**

Default: 8000 0008h

Attribute: See below.

Bits	Description
31:29	<b>CAPTYPE. Capability type.</b> Read-only. This register is hardwired to 100b to indicate this is the interrupt discovery and configuration capability.
28:24	Reserved.
23:16	<b>IRIDX. Interrupt Register Index.</b> Read-write. This register select the register which is accessed through the interrupt register dataport.
15:8	<b>CAPPTR. Capability Pointer.</b> Read-only. This register is hardwired to 0.
7:0	<b>CAPID. Capabilities ID.</b> Read-only. Specifies the capabilities ID for HyperTransport™ technology configuration space.

**HyperTransport™ Technology Interrupt Register Dataport****DevA:0xF4**

Default: 0000\_0000h.

Attribute: Read-write.

Bits	Description
31:0	<b>IRDP. Interrupt register dataport.</b> Read-write. This register provides access to the interrupt register which is specified by the index register IRIDX. Only doubleword access to this register is supported.

## 4.3 LPC Bridge Configuration Registers (DevB:0xXX)

These registers are located in PCI configuration space on the primary PCI interface, in the second device (device B), function 0. See Section 4.1.2 on page 136 for a description of the register naming convention.

**LPC Bridge Vendor And Device ID Register****DevB:0x00**

Default: 7468 1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>LPC bridge device ID</b>
15:0	<b>Vendor ID</b>

**LPC Bridge Status And Command Register****DevB:0x04**

Default: 0220 000Fh.

Attribute: See below.

Bits	Description
31:4	Hardwired to default values.
3	<b>SPCYCEN. Special cycle enable.</b> Read-write. 1=The IC responds to shutdown special cycles by using INIT to reset the processor. 0=The IC ignores shutdown special cycles.
2:0	<b>I/O, memory, and master enable.</b> Read-only. Hardwired in the enabled state.

**LPC Bridge Revision And Class Code Register****DevB:0x08**

Default: 0601 00XXh see below.

Attribute: Read-only.

Bits	Description
31:8	<b>CLASSCODE.</b> Provides the ISA bridge class code as defined in the PCI specification.
7:0	<b>REVISION. LPC bridge silicon revision.</b> The value of this register is revision-dependent.

**LPC Bridge BIST-Header-Latency-Cache Register****DevB:0x0C**

Default: 0080 0000h.

Attribute: Read-only.

Bits	Description
31:24	<b>BIST.</b> These bits fixed at their default values.
23:16	<b>HEADER.</b> These bits fixed at their default values.
15:8	<b>LATENCY.</b> These bits fixed at their default values.
7:0	<b>CACHE.</b> These bits fixed at their default values.

**LPC Bridge Subsystem ID and Subsystem Vendor ID Register****DevB:0x2C**

Default: 0000 0000h.

Attribute: Read-only.

Bits	Description
31:16	<b>SSID. Subsystem ID.</b> This field is write accessible through DevB:0x70.
15:0	<b>SSVENDORID. Subsystem vendor ID.</b> This field is write accessible through DevB:0x70.

**I/O Control 1 Register****DevB:0x40**

Default: 00h.

Attribute: See below.

Bits	Description
7	<b>NMIONERR. Generate an NMI on error.</b> Read-write. 1=An NMI is generated when one of the error status bits specified by Section 3.1.2 on page 37 is set. Note: see PM48[NMI2SMI_EN] for information on how NMI interrupts may be controlled.
6	<b>LPCERR. LPC transaction error status.</b> Read; set by hardware; write 1 to clear. The bit is set High by hardware when an LPC sync error occurs.
5	<b>SUBDEC. Subtractive decoding off of the secondary PCI bus.</b> Read-write. 1=All memory mapped and I/O mapped transactions received by the host that are not destined for any internally specified devices or buses are sent to the secondary PCI bus; if DEVSEL_L is not asserted for these PCI bus cycles, then the IC asserts DEVSEL_L during the subtractive window, and asserts STOP_L to complete the cycle; the cycle is then retransmitted to the LPC bus; if, during the PCI bus cycle, DEVSEL_L is asserted by an external component before the subtractive window, then the cycle is assumed to be for the secondary PCI bus and allowed to complete. 0=All memory mapped and I/O mapped transactions received by the host that are not destined for any internally specified devices or buses are sent directly to the LPC bus.
4	<b>LPC_IOR. LPC I/O recovery.</b> Read-write. 1= I/O recovery delay (specified by IORT) enforced for both LPC and legacy I/O cycles. 0=I/O recovery delay only enforced for legacy I/O cycles (cycles to the DMA controller, legacy PIC, programmable interval timer, and real-time clock).
3	<b>IORT. I/O recovery time.</b> Read-write. This bit specifies the amount of time enforced between internal legacy I/O cycles (cycles to the DMA controller, legacy PIC, programmable interval timer, and real-time clock) and, if enabled by LPC_IOR, LPC cycles. 0=There are a minimum of 22 PCLK cycles between the end of each I/O cycle and the beginning of the next I/O cycle. 1=There are a minimum of 54 PCLK cycles between I/O cycles. This bit does not affect memory cycles (only I/O cycles).
2	<b>BLE. BIOS lock enable.</b> Read; write 1 only. 1=Setting DevB:0x40[RWR] from 0 to 1 sets PM44[IBIOS_STS] and generates an SMI. 0=Setting DevB:0x40[RWR] from 0 to 1 does not set PM44[IBIOS_STS] and does not generate an SMI. Once BLE is set, it can only be cleared by RESET_L.
1	<b>PW2LPC. Discarded posted request targeting LPC.</b> Read; set by hardware; write 1 to clear. This bit is set by hardware if a posted request that targets a device on the LPC gets discarded. This can only be set if DevB:0x41[DPW2LPC] is programmed High. Setting this bit can result in a NMI if enabled with DevB:0x40[NMIONERR].
0	<b>RWR. LPC ROM write.</b> Read-write. 1=Memory writes to BIOS ROM address space, as defined by DevB:0x43, are allowed to pass onto the LPC bus. 0=Memory writes to BIOS ROM address space are dropped.

**I/O Control 2 Register****DevB:0x41**

Default: 00010?00b. (Pin-strap: see below)

Attribute: See below.

Bits	Description
7	Reserved.
6	Reserved.
5	<b>P92FR. Port 92 fast reset.</b> Read-write. 1=Writes that attempt to set I/O PORT92[0]—the fast processor reset bit—are enabled. 0=Writes to PORT92[0] do not generate a processor reset pulse using the HyperTransport™ technology INIT interrupt.

4	<b>DPW2LPC. Discard posted writes targeting LPC.</b> Read-write. 1=Posted writes targeting the LPC while a LPC bus master or DMA cycle is in progress get discarded. 0=Posted writes are passed to the LPC.
3	<p><b>SPECEN. Special cycle enable onto secondary PCI bus.</b> Read-write. 1=Special cycles received from the host are passed to the secondary PCI bus. 0=No special cycles are passed to the secondary PCI bus. The supported special cycles and the associated field of the PCI data phase are:</p> <p>Shutdown: 0000_0000h.  Halt: 0000_0001h.  Writeback invalidate 0001_0002h.  Invalidate 0002_0002h.  Enter SMM: 0005_0002h.  Exit SMM: 0006_0002h.  VID/FID change 0007_0002h.  Stop Grant: 0012_0002h.  Thermal Trip Point Crossed 0014_0002h.  FERR_L asserted 0021_0002h.  FERR_L deasserted 0022_0002h.</p>
2	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times; setting it High results in undefined behavior.
1	<b>NMIDIS. NMI disable.</b> Read-only. This provides read access to RTC70[NMIDIS].
0	<b>SHEN. Shadow register access enable.</b> Read-write. 1=Shadowed I/O access to legacy write-only registers is enabled. 0=Normal access of legacy registers. The following table shows all registers affected by this bit.



**Table 47. Registers Affected by DevB:0x41[SHEN]**

I/O Port	R/W	Normal Mode	Shadow Mode
DMA: 00h, 02h, 04h, 06h, C0h, C4h, C8h, CCh	W	Base address for DMA channel	Current address for DMA channel
	R	Current address for DMA channel	Base address for DMA channel
DMA: 01h, 03h, 05h, 07h, C2h, C6h, CAh, CEh	W	Base byte count for DMA channel	Current byte count for DMA channel
	R	Current byte count for DMA channel	Base byte count for DMA channel
DMA: 08h/D0h	W	Command Register DMA CH[3:0]/[7:4]	Status Register DMA CH[3:0]/[7:4]
	R	Status Register DMA CH[3:0]/[7:4]	First read: Command reg DMA CH[3:0]/[7:4] Second read: Request reg DMA CH[3:0]/[7:4] Third read: Mode register DMA CH0/4 Fourth read: Mode register DMA CH1/5 Fifth read: Mode register DMA CH2/6 Sixth read: Mode register DMA CH3/7
DMA: 09h/D2h, 0Ah/D4h, 0Bh/D6h	W	See DMA controller	Reserved
DMA: 0Ch/D8h, 0Dh/DAh, 0Eh/DCh	W	See DMA controller	Same as normal mode
DMA: 0Fh/DEh	W	Write all masks [3:0]/[7:4]	Write all masks [3:0]/[7:4]
	R	Reserved	Read all masks [3:0]/[7:4]
PIT: 40h	R	Status byte counter 0	First read: Status byte counter 0 Second read: CRL for counter 0 Third read: CRM for counter 0 Fourth read: CRL for counter 1 Fifth read: CRM for counter 1 Sixth read: CRL for counter 2 Seventh read: CRM for counter 2
	R	Status byte counter 1	Status byte counter 1
	R	Status byte counter 2	Status byte counter 2
	R	Status byte counter 1	Status byte counter 1
	R	Status byte counter 2	Status byte counter 2
	R	Status byte counter 1	Status byte counter 1
	R	Status byte counter 2	Status byte counter 2

**Table 47. Registers Affected by DevB:0x41[SHEN] (Continued)**

PIC: 20h	R	Interrupt request register for PIC 1	First read: ICW1 for controller 1 Second read: ICW2 for controller 1 Third read: ICW3 for controller 1 Fourth read: ICW4 for controller 1 Fifth read: OCW1 for controller 1 Sixth read: OCW2 for controller 1 Seventh read: OCW3 for controller 1 Eighth read: ICW1 for controller 2 Ninth read: ICW2 for controller 2 Tenth read: ICW3 for controller 2 Eleventh read: ICW4 for controller 2 Twelfth read: OCW1 for controller 2 Thirteenth read: OCW2 for controller 2 Fourteenth read: OCW3 for controller 2
PIC: 21h	R	In service register for PIC 1	In service register for PIC 1
PIC: A0h	R	Interrupt request register for PIC 2	Interrupt request register for PIC 2
PIC: A1h	R	In service register for PIC 2	In service register for PIC 2

**Legacy Blocks Control Register****DevB:0x42**

Default: 07h.

Attribute: Read-write.

Bits	Description
7:3	Reserved.
2	<b>DMAEN. Internal DMAC (Direct Memory Access Controller) Enable.</b> 1=Enable internal 8237-based DMAC and associated logic; accesses to I/O ports 00h through 0Fh, 80h through 8Fh, and C0h through DFh are routed to the internal DMAC. 0=Disable DMAC and associated logic; DMAC accesses are routed to the LPC bus.
1	<b>PITEN. Internal PIT (Programmable Interval Timer) Enable.</b> 1=Enable internal 8254-based PIT and associated logic; accesses to I/O ports 40h through 43h and 61h are routed to the internal PIT. 0=Disable PIT and associated logic; PIT accesses are routed to the LPC bus.
0	<b>PICEN. Internal PIC (Programmable Interrupt Controller) Enable.</b> 1=Enable internal 8259-based PIC and associated logic; accesses to I/O ports 20h, 21h, 92h, A0h, A1h, F0h, F1h, 4D0h, 4D1h as well as interrupt acknowledge cycles are routed to the internal PIC and associated logic. 0=Disable PIC and associated logic; PIC accesses are routed to the LPC bus.

**ROM Decode Control Register****DevB:0x43**

Default: 00h.

Attribute: Read-write.

**SEGEN. ROM Segment Enables.** This register specifies the address space mapped to the BIOS ROM on the LPC or PCI bus (see DevB:3x48[PCIBIOS]). Each bit specifies if the LPC or PCI bus is enabled for BIOS. For each of these bits: 1=Enables an address range as a BIOS ROM access. 0=The corresponding address range is not decoded as a BIOS ROM access and not forwarded to the LPC or PCI bus. Writes to disabled segments are ignored and reads return all 0xF. The bits control the following address ranges (the last column shows the translated LPC bus addresses):

Bits	Size	Host Address Range[31:0]	Address translation for LPC bus
7	4 megabytes	FFC0_0000h–FFFF_FFFFh	FFC0_0000h–FFFF_FFFFh
6	1 megabyte	FFB0_0000h–FFBF_FFFFh	FFB0_0000h–FFBF_FFFFh
5	32K bytes	000E_8000h–000E_FFFFh	FFFE_8000h–FFFE_FFFFh
4	32K bytes	000E_0000h–000E_7FFFh	FFFE_0000h–FFFE_7FFFh
3	32K bytes	000D_8000h–000D_FFFFh	FFFD_8000h–FFFD_FFFFh
2	32K bytes	000D_0000h–000D_7FFFh	FFFD_0000h–FFFD_7FFFh
1	32K bytes	000C_8000h–000C_FFFFh	FFFC_8000h–FFFC_FFFFh
0	32K bytes	000C_0000h–000C_7FFFh	FFFC_0000h–FFFC_7FFFh

*Note:* The following ranges are always specified as BIOS address ranges. See DevB:0x80 for more information about how access to BIOS spaces may be controlled.

Size	Host Address Range[31:0]	Address translation for LPC bus
64K bytes	FFFF_0000h–FFFF_FFFFh	FFFF_0000h–FFFF_FFFFh
64K bytes	000F_0000h–000F_FFFFh	FFFF_0000h–FFFF_FFFFh

**Prefetchable Memory Control Register****DevB:0x44**

Default: 0001h.

Attribute: Read-write.

Bits	Description
15:4	<b>TOM[31:20]. Top of memory bits[31:20].</b> This specifies the top of system memory. System memory space is treated as prefetchable by the PCI bridge. It is defined as follows: System_memory = (PCI_address[31:20] <= TOM[31:20]); Note: If ALLPF is set High, then TOM is ignored.
3:1	Reserved.
0	<b>ALLPF. All of memory space (4 gigabytes) is prefetchable.</b> 1=All 4 gigabytes of memory space is prefetched by the PCI bridge; when memory read transactions with PCI command encoding of 6h are initiated by PCI bus masters, then the data is prefetched, regardless of the address. 0=Only memory read accesses with PCI command encoding of Ch or Eh, or memory accesses with PCI command encoding 6h below the top of memory specified in the TOM field of this register are prefetched.

**Miscellaneous Control Register****DevB:0x47**

Default: 00h.

Attribute: See below.

Bits	Description
7	<b>EHCDIS. Enhanced host controller configuration space disable bit.</b> Read-write. 1= The EHC configuration space (Dev0:2xXX) is disabled. Reads return all 0xF (and an NXA error response) and writes are ignored (and return an NXA error response). 0= The EHC configuration space is enabled. Note: If the EHC is disabled the IC is not capable to handle USB 2.0 traffic.
6	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times; if it is High then undefined behavior results.
5	<b>ALLTOD. All internal PCI interrupts mapped to PIRQD_L.</b> Read-write. 1=Internal PCI interrupts—including the two USB interrupts, the AC '97 interrupts, the Ethernet controller interrupt, the SMBus 2.0 controller interrupt, the primary and secondary port (when these ports are in native mode) IDE controller interrupts, and the GPIO interrupts as specified by DevB:0x4B[MPIRQ]—are mapped to assert PIRQD_L when they become active; i.e., all these interrupts are shared on PIRQD_L. 0=Internal PCI interrupts are distributed across all four PIRQ[A,B,C,D]_L pins as specified in Section 3.4.2.1 on page 41 and by DevB:0x4B[MPIRQ].
4	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times; if it is High then undefined behavior results.
3	<b>CMLK_B8. CMOS RAM offsets B8h through BFh lock.</b> Read; write 1 only. 0=Accesses to the eight bytes of CMOS RAM (powered by the VDD_COREAL plane) addressed from B8h to BFh are read-write accessible. 1=Writes to these bytes are ignored and read always return FFh (regardless as to which of the I/O ports from 70h to 73h are used for the access). After this bit is set High, it cannot be cleared again by software; it can only be cleared by PWROK reset.
2	<b>CMLK_38. CMOS RAM offsets 38h through 3Fh lock.</b> Read; write 1 only. 0=Accesses to the eight bytes of CMOS RAM (powered by the VDD_COREAL plane) addressed from 38h to 3Fh are read-write accessible. 1=Writes to these bytes are ignored and read always return FFh (regardless as to which of the I/O ports from 70h to 73h are used for the access). After this bit is set High, it cannot be cleared again by software; it can only be cleared by PWROK reset.
1	<b>RSTONLE. Reset on HyperTransport™ link error.</b> Read-write. 1=Assert the RESET_L and LDTRST_L pins when either (1) a CRC error is detected on the incoming HyperTransport link (DevA:0xC4[CRCERR]) or (2) the incoming HyperTransport link is flooded with sync packets. 0=HyperTransport link errors do not result in resets.
0	<b>SWRST. Software reset.</b> Write-only. When this bit is written with 1, a reset pulse is generated over RESET_L and LDTRST_L. This bit always reads as 0.

**Function/Device Enable Register****DevB:0x48**

Default: FFFFh.

Attribute: Read-write.

Bits	Description
15:8	<b>SECENS[7:0]. Secondary PCI bus device enables.</b> Each of these bits apply to the first 8 internal devices on the secondary PCI bus. Bit[0] applies to device 0, etc. Bits that apply to device numbers that are not implemented internally are ignored. 1=The device's configuration space is enabled. 0=The device's configuration space is invisible; accesses to the space are master aborted; reads return all ones.
7:0	<b>PRIENS[7:0]. Primary PCI bus function enables.</b> Each of these bits apply to Device B functions inside the IC (on the primary bus). Bit[1] applies to function 1, bit[3] to function 3, etc. However, bit[0] is ignored since function 0 cannot be disabled. Bits that apply to internal functions that do not exist in the IC are required to be Low at all times; setting may result in undefined behaviour. 1=The function's configuration space is enabled. 0=The function's configuration space is disabled; accesses to the space are master aborted and reads return all ones.

**IOAPIC Configuration Register 0****DevB:0x4A**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:2	Reserved.
1	<b>LINTEN_NMI. Local Interrupt Enable NMI.</b> 1 = MT[3] is set in HyperTransport™ link NMI interrupt request packets not coming from the IOAPIC.
0	<b>LINTEN_INTR. Local Interrupt Enable INTR.</b> 1 = MT[3] is set in HyperTransport link INTR interrupt request packets not coming from the IOAPIC.

**IOAPIC Configuration Register 1****DevB:0x4B**

Default: 00h.

Attribute: Read-write.

Bits	Description
7	<b>MPIRQ. Multi-processor IRQ mode.</b> This bit is combined with the mask bits of IOAPIC redirection table entries 23 through 20 and used to specify if the GPIO[31:28] inputs are mapped to drive the PIRQ[A,B,C,D]_L pins Low, respectively (GPIO28 to PIRQA_L, etc.). For each of these four pins:  <pre> if (MPIRQ &amp; MASK[23:20] &amp; ~GPIO[31:28])     then PIRQ[A,B,C,D]_L = 0;     else PIRQ[A,B,C,D]_L = Z; // high impedance </pre> The polarity of GPIO[31:28], in the above equation, is as seen on the external pins; the polarity is not altered the programming of DevB:3x[DF:DC]. The four GPIO pins may all be mapped to PIRQD_L, as specified by DevB:0x47[ALLTOD].
6	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
5	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
4:3	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
2	Reserved.

Bits	Description (Continued)
1	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
0	<b>APICEN. IOAPIC enable.</b> 0=Accesses to the IOAPIC memory mapped register space are ignored; also, the dual 8259 PIC may generate interrupts through HyperTransport™ messages. 1=The IOAPIC is enabled; the PIC does not directly generate interrupt requests through HyperTransport messages (although INTR goes through the IOAPIC redirection table, which may result in an interrupt request HyperTransport message).

**Miscellaneous Register****DevB:0x4C**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:13	Reserved.
12	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
11	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
10	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
9	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
8	<b>Must be Low.</b> This bit is required to be Low at all times; if it is High then undefined behavior results.
7:0	<b>Must be Low.</b> These bits are required to be Low at all times; if they are High then undefined behavior results.

**Miscellaneous Register****DevB:0x4E**

Default: C000h.

Attribute: Read-write.

Bits	Description
15:0	<b>Must be Low.</b> These bits are required to be set Low during initialization; if any bits are High then undefined behavior results.

**PCI Prefetching Control 0****DevB:0x50**

Default: 0000 0000h.

Attribute: see below.

Bits	Description
31:24	Reserved.
23:21	<b>[DPDM7, DPDH7, PFEN7_L].</b> These bits apply to the PREQ_L/PGNT_L pair. See bits 2:0.
20:18	<b>[DPDM6, DPDH6, PFEN6_L].</b> These bits apply to the REQ_L[6]/GNT_L[6] pair. See bits 2:0.
17:15	<b>[DPDM5, DPDH5, PFEN5_L].</b> These bits apply to the REQ_L[5]/GNT_L[5] pair. See bits 2:0.
14:12	<b>[DPDM4, DPDH4, PFEN4_L].</b> These bits apply to the REQ_L[4]/GNT_L[4] pair. See bits 2:0.
11:9	<b>[DPDM3, DPDH3, PFEN3_L].</b> These bits apply to the REQ_L[3]/GNT_L[3] pair. See bits 2:0.
8:6	<b>[DPDM2, DPDH2, PFEN2_L].</b> These bits apply to the REQ_L[2]/GNT_L[2] pair. See bits 2:0.

5:3	<b>[DPDM1, DPDH1, PFEN1_L].</b> These bits apply to the REQ_L[1]/GNT_L[1] pair. See bits 2:0.
2	<p><b>DPDM0. Discard prefetch data upon upstream or peer to peer transaction.</b> Read-write. This bit applies to the REQ_L[0]/GNT_L[0] pair.</p> <p>1 = When there is a transaction from any PREQ_L/REQ_L[6:1] input which targets another's device on the PCI (peer-to-peer), or a transaction from the same master (request from REQ_L[0]) targetting a different location (either to the host or peer-to-peer), no further prefetching occurs and any prefetch data not yet transferred is discarded, except for one doubleword (the next one to transfer). Discarding only happens if this transaction is not finished with a master abort.</p> <p>0 = Master requests from PREQ_L/REQ_L[6:0] inputs do not affect prefetching.</p>
1	<p><b>DPDH0. Discard prefetch data upon host request.</b> Read-write. This bit applies to the REQ_L[0]/GNT_L[0] pair.</p> <p>1 = When there is a host request to the PCI bridge which is executed on the PCI bus, no further prefetching occurs and any prefetch data not yet transferred is discarded, except for one doubleword (the next one to transfer).</p> <p>0 = Host requests to the PCI bridge do not effect prefetching.</p> <p>Note: Programming of this bit may vary based on platform requirements. DPDH is typically programmed low by system BIOS. The system BIOS may set this bit to protect against stale prefetch-data scenarios, as described in the PCI specification, revision 2.3, section 3.10, point 6; scenarios similar to this have been observed, albeit rarely. However, if the PCI bus includes a device that is accessed frequently as a target, then setting this bit may result in reduced memory read bandwidth.</p>
0	<p><b>PFEN0_L. Prefetch enable (active Low) 0 for busmaster burst read requests.</b> Read-write. This bit applies to the REQ_L[0]/GNT_L[0] pair.</p> <p>1 = Prefetching is not enabled. When prefetching is not enabled and a non-burst read request occurs, then the IC requests one double-word of data from the host. When prefetching is not enabled and a burst read request occurs, then the IC requests two double-words of data from the host.</p> <p>0 = Prefetching is enabled. When prefetching is enabled and a non-burst read request occurs, then the IC requests one double-word of data from the host. When prefetching is enabled and a burst read request occurs, then the IC requests prefetch data as controlled by DevB:0x50.</p>

### PCI Prefetching Control 1

DevB:0x54

When prefetching is enabled by DevB:0x50[PFEN<sub>x</sub>\_L] (x=0...7) and a burst read request occurs, then the IC requests data starting from the request address up to the end of the cacheline, when the request address is smaller or equal to DevB:0x44[TOM] or when DevB:0x44[ALLPF] = 1. When prefetching is disabled and the request address is smaller or equal to DevB:0x44[TOM] or when DevB:0x44[ALLPF] = 1 then the IC requests two double-words of data. Otherwise it requests one double-word of data (with the correct byte enables).

Prefetching of a second, third or fourth cacheline as enabled by IPF\_MR, IPF\_MRL, IPF\_MRM only occurs when all of the following conditions are valid.

1. More than 1 internal PCI response buffer is available.
2. At least 1 internal HyperTransport™ command buffer is available.
3. The request address is smaller or equal to DevB:0x44[TOM] or DevB:0x44[ALLPF] = 1.

Prefetching of the first, second, third or fourth cacheline as enabled by CPFEN\_MR, CPFEN\_MRL, CPFEN\_MRM only occurs when all of the following conditions are valid.

1. At least 1 internal PCI response buffer is available.
2. At least 1 internal HyperTransport command buffer is available.

The request address is smaller or equal to DevB:0x44[TOM] or DevB:0x44[ALLPF] = 1.

Default: 0000 0185h.

Attribute: see below.

Bits	Description
31:16	Reserved.
17	Reserved.
16	Reserved.
15:12	<b>PDDTV. Prefetch data discard timer value.</b> Read-write. The discard timer for prefetch data expires after $PDDTV * 2^6 + 64$ clock cycles. With the expiration of this timer all prefetch data is discarded. (Revision C1 and later)
11	Reserved.
10:9	<b>BLLIMIT. Burst length limit.</b> .Read-write. These bits specify how many cachelines are allowed to be continuously transferred to/from the external master. These bits apply to read and write bursts. 00 = no limitation 01 = 4 cachelines 10 = 8 cachelines 11 = 12 cachelines  If the burst length counter exceed these limits then the IC disconnects the transfer (even if in the read case the next cacheline is available). Any prefetch operation stays unaffected.
8:6	<b>[CPFEN_MRM, IPF_MRM].</b> These bits apply to memory read multiple requests. See bits 2:0.
5:3	<b>[CPFEN_MRL, IPF_MRL].</b> These bits apply to memory read line requests. See bits 2:0.
2	<b>CPFEN_MR. Continuous prefetch enable for memory read request.</b> Read-write. 1 = One or more cachelines (up to three) of prefetch data is requested when a cacheline starts to be transferred to the external master on the PCI bus. More cachelines are only requested if the IPF count is not yet satisfied. 0 = No cacheline of prefetch data is requested when a cacheline is transferred to the master.
1:0	<b>IPF_MR. Initial prefetch for memory read request.</b> Read-write. These bits specify how many cachelines of prefetch data are requested upon an initial burst read request. This does not apply to any burst read request that is retried because of a previous target disconnect. 00 = 1 cacheline 01 = 2 cachelines 10 = 3 cachelines 11 = 4 cachelines



**Debug Control 1****DevB:0x58**

Default: 0000 0000h.

Attribute: see below.

Bits	Description
31	<b>Must be Low.</b> Read-write. These bits are required to be Low at all times. Setting this bit can result in an undefined behaviour.
30:8	Reserved. Read-only. Reading these bits provides undefined values.
7:0	Reserved. Read-only. Reading these bits provides undefined values.

**Debug Control 2****DevB:0x5C**

Default: 0000 0000h.

Attribute: see below.

Bits	Description
31:3	Reserved.
2	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times. Setting this bit may result in an undefined behaviour.
1	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times. Setting this bit may result in an undefined behaviour.
0	<b>Must be Low.</b> Read-write. This bit is required to be Low at all times. Setting this bit may result in an undefined behaviour.

**Device and Subsystem ID Read-Write Register****DevB:0x70**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>SSID. Subsystem ID.</b> The value placed in this register is visible DevB:0x2C[31:16].
15:0	<b>SSVENDORID. Subsystem vendor ID.</b> The value placed in this register is visible DevB:0x2C[15:0].

**Non Volatile RAM Base Address Register****DevB:0x74**

Default: DE00h.

Attribute: see below.

Bits	Description
15:8	<b>IOBASE. I/O space base Address.</b> Read-write. This field specifies bits[15:8] the I/O address space from which the 256-byte non-volatile RAM may be accessed. The non-volatile RAM is powered by the VDD_COREAL plane; if VDD_COREAL becomes invalid, then the contents of the non volatile RAM become invalid.
7:1	Read-only. These bits are fixed to 7'b 000_0000.
0	<b>NVRAMEN. Non-volatile RAM enable.</b> Read-write. 1=Accesses to the 256 byte I/O space specified DevB:0x74[IOBASE] is claimed by non-volatile RAM. 0= Nonvolatile RAM doesn't claim cycles.

**BIOS Access Control Registers****DevB:0x80, DevB:0x84, and DevB:0x88**

Default: 0000 0000h (for each).

Attribute: Read-write.

These registers consists of 24 4-bit registers called OAR (open at reset) locks. Each 4-bit register applies to a sector of the BIOS in the 5 megabyte BIOS range at the top of the 4-gigabyte address space as follows:

- DevB:0x84 and DevB:0x80 contain 16 four-bit lock registers, called OAR<sub>x</sub> where x ranges across [F:0]; each four-bit register controls a 64Kbyte address range at the top megabyte of memory as follows: [FFF<sub>x</sub>\_FFFh:FFF<sub>x</sub>\_0000h].
- DevB:0x88 contains 8 four-bit lock registers, called OAR<sub>x</sub> where x ranges as [E, C, A, 8, 6, 4, 2, 0]; each four-bit register controls an 8Kbyte address range as follows: [FFBF\_(x+1)FFFh:FFBF\_x000h].

Accesses to BIOS space in the low megabyte (between 000C\_0000h and 000F\_FFFFh) are mapped to the top megabyte (between FFFC\_0000h and FFFF\_FFFFh) on the LPC bus; the OAR locks for these apply to these accesses based on the remapped address at the top megabyte. Note: there is an additional OAR lock specified in DevB:0x8C. Note: OAR locks only apply to BIOS address space; if there is an access to an OAR lock address range that is not in BIOS address space as defined by DevB:0x43, then the OAR lock register is ignored. Note: the OAR locks only apply to the BIOS address space on the LPC bus; if DevB:3x48[PCIBIOS] = 1, then they are ignored.

As defined below, access to BIOS space can be limited to when the host is in system management mode (SMM). HyperTransport technology system management messages are sent that specify when the host is in SMM.

**Table 48. OAR<sub>x</sub> Lock Locations**

Register	Bits 31:28	Bits 27:24	Bits 23:20	Bits 19:16	Bits 15:12	Bits 11:8	Bits 7:4	Bits 3:0
DevB:0x88:	OARE	OARC	OARA	OAR8	OAR6	OAR4	OAR2	OAR0
DevB:0x84:	OARF	OARE	OARD	OARC	OARB	OARA	OAR9	OAR8
DevB:0x80:	OAR7	OAR6	OAR5	OAR4	OAR3	OAR2	OAR1	OAR0

**Table 49. OAR<sub>x</sub> Lock Bit Descriptions**

Bits	Description
OAR <sub>x</sub> [3]	<b>FLLOCK. Full access to RD/WRLOCK lock.</b> Read; write 1 only. This bit can only be set High by software; it is cleared by RESET_L. 0=Read-write access to RDLOCK and WRLOCK enabled. 1=Write access to RDLOCK and WRLOCK disabled (whether the system is in SMM mode or not).
OAR <sub>x</sub> [2]	<b>SLLOCK. SMM access to RD/WRLOCK lock.</b> Read; write 1 only. This bit can only be set High by software; it is cleared by RESET_L. 0=Read-write access to RDLOCK and WRLOCK enabled (if FLLOCK=0). 1=Write access to RDLOCK and WRLOCK only enabled in SMM mode (if FLLOCK=0).
OAR <sub>x</sub> [1]	<b>WRLOCK. BIOS sector x write lock.</b> Read; write if enabled by SLLOCK and FLLOCK. 0=Write access to BIOS sector x enabled (if DevB:0x40[RWR]=1). 1=Write access to BIOS sector x disabled.
OAR <sub>x</sub> [0]	<b>RDLOCK. BIOS sector x read lock.</b> Read; write if enabled by SLLOCK and FLLOCK. 0=Read access to BIOS sector x enabled. 1=Read access to BIOS sector x disabled.

**OAR Control Register****DevB:0x8C**

Default: 00h.

Attribute: See below.

Bits	Description
7:6	Reserved.
5	<b>SMIACK. System in locked state.</b> Read-only. 1=The last SMI mode special cycle received by the IC indicated that the host is in system management mode. 0=The last SMI mode special cycle received by the IC indicated that the host is in not system management mode.
4	<b>LKLOCK. SMM access to the ROM access registers lock.</b> Read, write 1 only. This bit can only be set High by software; it is cleared by RESET_L. 0=Write access to DevB:0x80, DevB:084, DevB:0x88, DevB:0x8C and DevB:0x43 enabled. 1=Write access to DevB:0x80, DevB:084, DevB:0x88, DevB:0x8C and DevB:0x43 only enabled in SMM mode (see DevB:0x80 or the definition of when the system is in SMM mode).
3:0	<b>OAR_ROB. OAR locks for rest of BIOS space.</b> Read-write. These four bits are defined identically to the OAR registers in DevB:0x80 and DevB:0x84. They apply to the BIOS ROM space across [FFEF_FFFFh:FFC0_0000h] (if the space is specified by DevB:0x43 to be BIOS).

**Read-Write Register****DevB:0x90**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	Read-write. These bits are read-write accessible through software; they control no hardware.

**High Precision Event Timer Base Address Register****DevB:0xA0**

Default: 0000 0000h.

Attribute: See below.

Bits	Description
31:10	<b>HPETBAR. High Precision Event Timer Base Address.</b> Read-write. These bits are used in the memory space for decoding the memory mapped High Precision Event Timer Registers. There is a maximum of 1 Kbyte for this base address.
9:1	Read-only. These bits are fixed to 0.
0	<b>HPETEN. High Precision Event Timer Enable.</b> Read-write. If set it means that the HPET address decoding through DevB:0xA0 is enabled.

**Watchdog Timer Base Address Register****DevB:0xA8**

Default: 0000 0002h.

Attribute: See below.

Bits	Description
31:5	<b>WDTBAR. Watchdog Timer Base Address.</b> Read-write. These bits are used in the memory space for decoding the memory mapped Watchdog Timer Registers. There is a maximum of 32 bytes for this base address.
4:3	Read-only. These bits are fixed to 0.
2	<b>WDTSILENT. Watchdog Timer Silent Mode.</b> Read-write. When set the watchdog timer does not cause any action as specified by WDT00[WACT] upon expiring. When cleared the watchdog timer silent mode is disabled.
1	<b>WDTHALT. Watchdog Timer Halt.</b> Read-write. When set the watchdog timer goes into the idle state and its functionality is disabled. When cleared the watchdog timer functionality is enabled.
0	<b>WDTEN. Watchdog Timer Enable.</b> Read-write. When set the watchdog timer address decoding as controlled by DevB:0xA8[WDTBAR] is enabled.

## 4.4 Legacy Registers

### 4.4.1 Miscellaneous Fixed I/O Space Registers

These registers are in I/O space, at fixed addresses. See Section 4.1.2 on page 136 for a description of the register naming convention.

**AT Compatibility Register****PORT61**

Fixed I/O space; offset: 61h.

Default: 00h.

Attribute: See below.

Bits	Description
7	<b>SERR. SERR_L latch.</b> Read-only. This bit is set High when SERR_L is asserted and stays High until cleared by PORT61[CLRSERR]. The state of this bit is combined with RTC70[NMIDIS] and PORT61[IOCHK] to generate NMI interrupts.
6	<b>IOCHK. IOCHK_L latch.</b> Read-only. This bit is set High when the serial IRQ signal IOCHK_L is asserted or if there is a <i>sync with error</i> message received on the LPC bus; it stays High until cleared by PORT61[CLRIOCHK]. The state of this bit is combined with RTC70[NMIDIS] and PORT61[SERR] to generate NMI interrupts.
5	<b>TMR2. Programmable interval timer, timer number 2 output.</b> Read-only. This bit provides the current state of the output signal from legacy PIT timer number 2.
4	<b>RSHCLK. Refresh clock.</b> Read-only. This bit toggles state at intervals specified by PIT timer 1 (normally, every 15 microseconds).

3	<b>CLRIOCHK. Clear PORT61[IOCHK].</b> Read-write. 1=Bit[6] of this register, IOCHK, is asynchronously cleared. 0=PORT61[IOCHK] can be set High.
2	<b>CLRSERR. Clear PORT61[SERR].</b> Read-write. 1=Bit[7] of this register, SERR, is asynchronously cleared. 0=PORT61[SERR] can be set High.
1	<b>SPKREN. Speaker enable.</b> Read-write. 1=The output of PIT timer number 2 drives the SPKR pin. 0=SPKR is held Low.
0	<b>TMR2EN. Programmable interval timer, timer number 2 enable.</b> Read-write. 1=PIT timer 2 is enabled to count. 0=PIT timer 2 is halted.

**System Control Register****PORT92**

Fixed I/O space; offset: 92h.

Default: 00h.

Attribute: See below.

Bits	Description
7:2	Reserved.
1	<b>A20EN. Processor address bit 20 enable.</b> Write-only; reads provide the current state of the virtual A20M_L HyperTransport™ link wire rather than the state of the bit. The value written to this bit is ORed with the KA20G bit from the keyboard controller before being sent as a HyperTransport technology message. In order for this register to control A20M_L, KA20G must be Low.
0	<b>INITCPU. Generate processor initialization command.</b> Read-write. When this bit is Low and then written to a High, the IC sends an INIT message. This bit must be written to a Low again before another INIT message can be sent. Note: use of this bit is enabled by DevB:0x41[P92FR].

**Fixed I/O Ports F0 and F1 and the FERR\_L and IGNNE\_L Logic****PORTF0 and PORTF1**

Fixed I/O space; offset: F0h and F1h.

Default: 00h.

Attribute: See below.

FERR\_L is used to control IGNNE\_L and generate IRQ13 to the PIC and IOAPIC. The following diagram shows the logic. FERR\_CLR\_L is asserted by (1) an I/O write to F0h, (2) an I/O write to F1h, (3) any processor reset command, and (4) PWROK reset; when any of these are active, FERR\_CLR\_L goes Low.

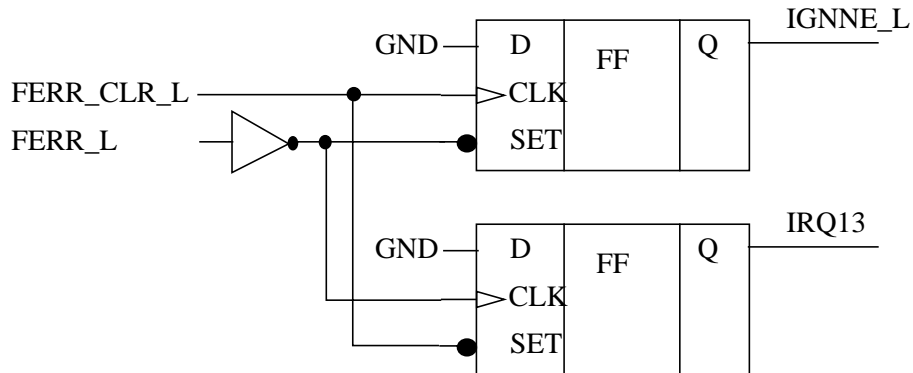


Figure 31. FERR\_L and IGNNE\_L Logic

Level Sensitive IRQ Select Register

PORT4D0

Fixed I/O space; offset: 4D0h and 4D1h.  
 Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	<b>LIRQ. Level sensitive IRQs.</b> Each of these 16 bits controls whether a corresponding IRQ line that enters the legacy PIC is edge sensitive (if the bit is Low) or level sensitive (if it is High). Edge sensitive interrupts must enter the PIC such that the rising edge generates the interrupt and level sensitive interrupts must enter the PIC as active Low; see Section 3.4.2.1 on page 41 for details about how the interrupts are mapped to the PIC. The bit numbers correspond directly to the IRQ numbers (e.g., bit[12] controls IRQ12). Bits[0 and 2] are reserved (IRQ0 is always edge sensitive and IRQ2 does not exist).

**System Reset Register****PORTCF9****Notes:**

1. This register is enabled by DevB:3x41[PCF9EN].
2. This register may be accessed only as a byte operation; 16- or 32-bit accesses to port CF8h are ignored by this register.

Fixed I/O space; offset: CF9h.

Default: 00h.

Attribute: See below.

Bits	Description
7:4	Reserved.
3	<b>FULLRST. Full reset.</b> Read-write. 1=Full resets require the IC to place the system in the SOFF state for 3 to 5 seconds; full resets occurs whenever (1) RSTCMD and SYSRST are both written High, (2) an AC power fail is detected (PWROK goes Low without the appropriate command), or (3) when PM46[2NDTO_STS] is set while DevB:3x48[NO_REBOOT]=0. 0=Full resets do not transition the system to SOFF; only the reset signals are asserted.
2	<b>RSTCMD. Reset command.</b> Write-only; always reads as a zero. When this bit is written with a 1, a reset is generated as specified by bits[3,1] of this register (bits[3,1] are observed in their state when RSTCMD is written to a 1; their previous value does not matter).
1	<b>SYSRST. System reset.</b> Read-write. This bit specifies whether a full system reset or a processor INIT is generated when PORTCF9[RSTCMD] is written to a 1. 1=Full system reset with RESET_L and LDTRST_L asserted for about 1.8 milliseconds. 0=INIT HyperTransport™ message is sent.
0	Reserved.

**4.4.2 Legacy DMA Controller (DMAC) Registers**

The legacy DMA controller (DMAC) in the IC supports the features required by the LPC I/F Specification Revision 1.0, which are a subset of legacy DMA Controllers. Single, demand, verify, and increment modes are supported. Block, decrement, cascade modes are not supported. Also, memory-to-memory transfers and external EOPs (end of process) are not supported.

There are 7 supported DMA channels. Channels 0-3 support 8-bit transfers and channels 5-7 support 16-bit transfers. There is no support for 32-bit DMA transfers. LPC Master device requests are made using channel 4.

Although not all registers in legacy DMA controllers are supported, the I/O address locations for the unsupported registers is consistent with legacy logic. The implemented DMAC registers are listed in the following table.

**Table 50. DMA Controller Register Summary**

Name	Size	Number	Comments
Base Address Registers	16 bits	8	1 for each channel (0-7) (see note)
Base Word Count Registers	16 bits	8	1 for each channel (0-7) (see note)
Current Address Registers	16 bits	8	1 for each channel (0-7) (see note)
Current Word Count Registers	16 bits	8	1 for each channel (0-7) (see note)
Status Registers	8 bits	2	1 for Master and 1 for Slave DMAC
Command Registers	1 bit	2	1 for Master and 1 for Slave DMAC
Mode Registers	5 bits	8	1 for each channel (0-7) (see note)
Mask Registers	4 bits	2	1 for Master and 1 for Slave DMAC
<b>Note:</b> Although channel 4 base and current registers exist for compatibility, they are not used.			

Note that not all bits in the command and mode registers of legacy DMA controllers are used in the IC's DMA controller. The bit usage for these registers are as follows.

**Table 51. DMA Command Register Bits (Master and Slave DMAC)**

Bit	8237 Function	DMAC Function of the IC
7	DACK sense	Obsolete
6	DREQ sense	Obsolete
5	Late/Extended write	Obsolete
4	Fixed/Rotating priority	Obsolete (always fixed priority)
3	Normal/Compressed timing	Obsolete
2	Controller enable/disable	Controller enable/disable
1	Ch0 address hold enable/disable	Obsolete
0	Memory-to-memory enable/disable	Obsolete

**Table 52. DMA Mode Register Bits (Master and Slave DMAC)**

Bit	8237 Function	DMAC Function of the IC
7:6	00b Demand mode select	00b Demand mode select
	01b Single mode select	01b Single mode select
	10b Block mode select	10b Obsolete
	11b Cascade mode select	11b Obsolete
5	Address increment/decrement select	Obsolete (always increment)
4	Auto initialization enable/disable	Auto initialization enable/disable
3:2	00b Verify transfer	00b Verify transfer
	01b Write transfer	01b Write transfer
	10b Read transfer	10b Read transfer
	11b Illegal	11b Illegal
1:0	Channel select	Channel select

**Note:** DMA channel 4 is hardwired into cascade mode; however is obsolete for all other channels.

For debug purposes writes to the DMA page registers at 0x80 to 0x87 show up on PCI and LPC. It is expected that no external device claims these cycles nor responds to these cycles.



### 4.4.3 Legacy Programmable Interval Timer (PIT) Registers

These timers are halted from counting, if enabled to do so in DevB:3x4C[PIT\_DIS], when PRDY is asserted.

Here are the ports used to access the legacy PIT:

**Table 53. PIT Register Summary**

Offset	Access	Port
40h	Write	Counter 0 write access port
	Read	Counter 0 read access port
41h	Write	Counter 1 access port
	Read	Counter 1 read access port
42h	Write	Counter 2 access port
	Read	Counter 2 read access port
43h	Write	Control byte
	Read	Not supported

**PIT Control Byte Register****PORT43**

Fixed I/O space; offset: 43h.

Default: 00h.

Attribute: write-only.

Bits	Description
7:6	<b>SC[1:0]. Select counter.</b> Specifies the counter that the command applies to as follows: 00b Counter 0. 01b Counter 1. 10b Counter 2. 11b Read back command.
5:4	<b>RW[1:0]. Read-write command.</b> Specifies the read-write command as follows: 00b Counter latch command. 01b Read-write least significant byte only. 10b Read-write most significant byte only. 11b Read-write least significant byte followed by most significant byte.
3:1	<b>M[2:0]. Counter mode.</b> Specifies the mode in which the counter selected by SC[1:0] operates as follows: 000b Interrupt on terminal count. 001b Hardware retriggerable one-shot (not supported). 010b Rate generator. 011b Square wave mode. 100b Software triggered strobe. 101b Hardware triggered (retriggerable) strobe (not supported). 110b When this value is written, 010b is stored in the register, rate generator mode. 111b When this value is written, 011b is stored in the register, square wave mode.
0	<b>BCD. Binary coded decimal.</b> 1=Counter specified by SC[1:0] operates in binary coded decimal. 0=Counter specified by SC[1:0] operates in 16-bit binary mode.

**4.4.4 Legacy Programmable Interrupt Controller (PIC)**

The legacy dual-8259 programmable interrupt controller (PIC) includes a master, which is accessed through ports 20h and 21h and controls IRQ[7:0], and a slave, which is accessed through ports A0h and A1h and controls IRQ[15:8].

Here are all the PIC registers.

**Table 54. PIC Register Summary**

Offset	Access type	Register
20h (master), A0h (slave)	Write-only; D[4]=1b	Initialization command word 1 (ICW1)
	Write-only; D[4:3]=00b	Operation command word 2 (OCW2)
	Read-write; D[4:3]=01b	Operation command word 3 (OCW3)
21h (master), A1h (slave)	Write-only	Initialization command word 2 (ICW2)
	Write-only	Initialization command word 3 (ICW3)
	Write-only	Initialization command word 4 (ICW4)
	Read-write	Operation command word 1 (OCW1)

D[4:3] above refers to bits[4:3] of the associated 8-bit data field. Normally, once ICW1 is sent, ICW2, ICW3, and ICW4 are sent in that order before any OCW registers are accessed.

**Initialization Command Word 1 Register****ICW1**

Fixed I/O space; offset: 20h for master and A0h for slave; data bit[4] must be High. Attribute: write-only.

Bits	Description
7:5	<b>A[7:5]. Interrupt vector address.</b> These bits are not implemented.
4	This should always be High.
3	<b>LTIM. Level triggered mode.</b> This bit is not implemented; PORT4D0 controls this function instead.
2	<b>ADI. Call address interval.</b> This bit is not implemented.
1	<b>SNGL. Single mode.</b> This bit must be programmed Low to indicate cascade mode.
0	<b>IC4. ICW4 needed.</b> This bit must be programmed High.

**Initialization Command Word 2 Register****ICW2**

Fixed I/O space; offset: 21h for master and A1h for slave.

Attribute: Write-only.

Bits	Description
7:3	<b>T[7:3]. Interrupt vector table base address bits[7:3].</b>
2:0	<b>A[10:8].</b> Obsolete. These bits must be programmed Low.

**Initialization Command Word 3 for Master Register****ICW3M**

Fixed I/O space; offset: 21h.

Attribute: Write-only.

Bits	Description
7:0	<b>SLAVES[7:0].</b> These bits must always be programmed to 04h.

**Initialization Command Word 3 for Slave Register****ICW3S**

Fixed I/O space; offset: A1h.

Attribute: Write-only.

Bits	Description
7:3	Reserved (must be programmed to all zeros).
2:0	<b>ID[2:0]</b> . These bits must always be programmed to 02h.

**Initialization Command Word 4 Register****ICW4**

Fixed I/O space; offset: 21h for master and A1h for slave.

Attribute: Write-only.

Bits	Description
7:5	Reserved (must be programmed all zeros)
4	<b>SFNM. Special fully nested mode.</b> This bit is normally programmed Low.
3:2	<b>BUFF and MS.</b> These two are normally programmed to 00b for non-buffered mode.
1	<b>AEOI. Auto EOI.</b> This bit is ignored; the IC only operates in normal EOI mode (this bit Low).
0	<b>UPM. x86 mode.</b> This bit is ignored; the IC only operates in x86 mode (this bit High).

**Operation Command Word 1 Register****OCW1**

Fixed I/O space; offset: 21h for master and A1h for slave.

Attribute: Write-only.

Bits	Description
7:0	<b>MASK[7:0]. Interrupt mask.</b> 1=Interrupt is masked. Masking IRQ2 on the master interrupt controller masks all slave-controller interrupts.

**Operation Command Word 2 Register****OCW2**

Fixed I/O space; offset: 20h for master and A0h for slave; data bits[4:3] must be 00b. Attribute: Write-only.

Bits	Description																				
7:5	<p><b>R (bit 7), SL (bit 6), and EOI (bit 5).</b> These are decoded as:</p> <table border="1"> <thead> <tr> <th>R, SL, EOI</th> <th>Function</th> <th>R, SL, EOI</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>000b<sup>1</sup></td> <td>Rotate in auto EOI mode clear.</td> <td>100b<sup>1</sup></td> <td>Rotate in auto EOI mode set</td> </tr> <tr> <td>001b</td> <td>Non-specific EOI mode.</td> <td>101b</td> <td>Rotate on non-specific EOI</td> </tr> <tr> <td>010b</td> <td>No operation.</td> <td>110b<sup>2</sup></td> <td>Set priority command</td> </tr> <tr> <td>011b</td> <td>Specific EOI command.</td> <td>111b<sup>2</sup></td> <td>Rotate on specific EOI command</td> </tr> </tbody> </table> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>Values 000b and 100b are not supported.</li> <li>Values 110b and 111b use the IRLEVEL field.</li> </ol>	R, SL, EOI	Function	R, SL, EOI	Function	000b <sup>1</sup>	Rotate in auto EOI mode clear.	100b <sup>1</sup>	Rotate in auto EOI mode set	001b	Non-specific EOI mode.	101b	Rotate on non-specific EOI	010b	No operation.	110b <sup>2</sup>	Set priority command	011b	Specific EOI command.	111b <sup>2</sup>	Rotate on specific EOI command
R, SL, EOI	Function	R, SL, EOI	Function																		
000b <sup>1</sup>	Rotate in auto EOI mode clear.	100b <sup>1</sup>	Rotate in auto EOI mode set																		
001b	Non-specific EOI mode.	101b	Rotate on non-specific EOI																		
010b	No operation.	110b <sup>2</sup>	Set priority command																		
011b	Specific EOI command.	111b <sup>2</sup>	Rotate on specific EOI command																		
4:3	Reserved (must be programmed all zeros)																				
2:0	<b>IRLEVEL. Interrupt request level.</b> Specifies the interrupt request level to be acted upon.																				

**Operation Command Word 3 Register****OCW3**

Fixed I/O space; offset: 20h for master and A0h for slave; data bits[4:3] must be 01b. Attribute: Write-only.

Bits	Description
7	Must be programmed Low.
6:5	<b>ESMM (bit 6) and SMM (bit 5). Special mask mode.</b> These are decoded as: [ESMM, SMM] = 0XbNo action. [ESMM, SMM] = 10bReset special mask mode. [ESMM, SMM] = 11bSet special mask mode.
4:3	01b
2	<b>P. Poll command.</b> 1=Poll enabled; next I/O read of the interrupt controller treated like an interrupt acknowledge cycle.
1:0	<b>RR (bit 1) and RIS (bit 0). Read register command.</b> These are decoded as: [RR,RIS] = 0Xb No action. [RR,RIS] = 10b Read in-request (IR) register. [RR,RIS] = 11b Read IS register.

**4.4.5 IOAPIC Registers**

The IOAPIC register set for the 24 IOAPIC interrupts supported by the IC is indexed through two fixed-location, memory-mapped ports: FEC0\_0000h, which provides the 8-bit index register, and FEC0\_0010h, which provides the 32-bit data port. Writes to the 32-bit data port at FEC0\_0010h must be 32-bit, aligned accesses; other than 32-bit writes result in undefined behavior. Reads provide all four bytes regardless of the byte enables. The access to these memory locations can be disabled using DevB:0x4B[APICEN].

The index register selects one of the following:

Index	Description	Attribute	Default
00h	APIC ID register. The ID is in bits[27:24]. All other bits are reserved.	Read-write	0000 0000h
01h	IOAPIC version register.	Read-only	0017 0011h
02h	IOAPIC arbitration ID register. The ID is in bits[27:24]. All other bits are reserved.	Read-only	0000 0000h
10h-3Fh	Redirection registers. Each of the 24 redirection registers utilizes two of these indexes. Bits[63:32] are accessed through the odd indexes and bits[31:0] are accessed through the even indexes.	Read-write	0000 0000 0001 0000h
40h-FFh	Reserved.		

The redirection registers are defined as follows:

Bits	Description
63:56	<b>Destination.</b> In physical mode, bits[59:56] specify the APIC ID of the target processor. In logical mode bits[63:56] specify a set of processors.
55:17	Reserved.
16	<b>Interrupt mask.</b> 1=Interrupt is masked.
15	<b>Trigger mode.</b> 0=Edge sensitive. 1=Level sensitive. Note: this bit is ignored for delivery modes of SMI, NMI, Init, and ExtINT, which are always treated as edge sensitive.
14	<b>IRR. Interrupt request receipt.</b> Set by hardware; cleared by hardware. This bit is not defined for edge-triggered interrupts. For level-triggered interrupts, this bit is set by the hardware after an interrupt is detected. It is cleared by receipt of EOI with the vector specified in bits[7:0].
13	<b>Polarity.</b> 0=Active High. 1=Active Low.
12	<b>Delivery status.</b> 0=Idle. 1=Interrupt message pending.
11	<b>Destination mode.</b> 0=Physical mode. 1=Logical mode.
10:8	<b>Delivery mode.</b> 000b=fixed. 001b=Lowest priority. 010b=SMI. 011b=Reserved. 100b=NMI. 101=Init. 110b=Reserved. 111b=ExtINT.
7:0	<b>Interrupt vector.</b>

Normally all level triggered interrupts are programmed active Low and all edge triggered interrupts are programmed active High. Normally redirection register 0 (INTR) is programmed to be active High, edge triggered.

The IC also provides alternative means for configuring the IOAPIC as required by the HyperTransport specification. This access method is realized by using DevA:0xF0 which contains the index register and DevA:0xF4 which is the dataport. The access to these configuration registers is always possible independent of the state of the DevB:0x4B[APICEN] bit.

The index register selects one of the following:

Index	Description	Attribute	Default
00h	Reserved.	Read-only	0000 0000h
01h	<b>Last Interrupt.</b> The number of the last interrupt is stored in bits[23:16]. All other bits are reserved.	Read-only	0017 0000h
02h-09h	Reserved.	Read-only	0000 0000h
10h-3Fh	<b>Interrupt definition registers.</b> Each of the 24 registers utilizes two of these indexes. Bits[63:32] are accessed through the odd indexes and bits[31:0] are accessed through the even indexes.	Read-write	0000 0000 F800 0001h
40h-FFh	Reserved.	Read-only	0000 0000h

The interrupt definition registers are defined as follows:

Bits	Description
63	<b>Wait for EOI.</b> Read-write. This bit is set by hardware when an interrupt request is sent and cleared by hardware when the EOI is returned. Software may write a 1 to clear this register without an EOI so that the device can send another interrupt. Writing 0 has no effect. This bit is not defined if RQEOI is clear.
62	<b>PassPW.</b> Read-write. 1= the HyperTransport™ interrupt message will have the pass PW bit set and is allowed to pass other posted requests. 0= the pass PW bit in the interrupt message will be clear.
61:56	Reserved.
55:32	<b>IntrInfo[55:32].</b> Read-write. These bits contain the interrupt information bits 55:32 which are send in the interrupt message.
31:24	<b>IntrInfo[31:24].</b> Read-write. These bits contain the interrupt information bits 31:24 which are send in the interrupt message. Should be programmed to F8h for compliance with earlier HyperTransport technology implementations.
23:6	<b>IntrInfo[23:6].</b> Read-write. These bits contain the interrupt information bits 23:6 which are send in the interrupt message.
5	<b>RQEOI. Request EOI.</b> Read-write. When set, after each interrupt request is sent the device waits for an EOI (or software clears the wait for EOI bit) before sending another interrupt.
4:2	<b>Message Type.</b> Read-write. Specifies the type of interrupt. See HyperTransport technology specification 1.02 for valid settings.
1	<b>Polarity.</b> Read-write. This bit is only valid for external interrupt sources (e.g., pins). 1= The interrupt signal is active-Low. 0= The interrupt signal is active-High.
0	<b>Mask.</b> Read-write. 1= Interrupt masked, no interrupt is sent from this source (default). 0= Interrupt can be sent.

#### 4.4.6 Watchdog Timer Registers

The Watchdog Timer Registers are non-enumerable memory mapped I/O registers. The base address is controlled by DevB:0xA8. All register can be accessed by 32-bit memory accesses.

**Table 55. Watchdog Timer Registers**

Address	Mnemonic	Name	Default
00h	WDT00	Watchdog Timer Control/Status Register	0000_0008h
08h	WDT08	Watchdog Timer Count Register	0000_0000h

**Watchdog Timer Control/Status Register****WDT00**

Default: 0000 0008h.

Attribute: see below.

Bits	Description
31:8	Read-only.
7	<b>WTRIG. Watchdog Trigger.</b> Read-write. Setting this bit triggers the watchdog timer to start a new count interval, counting down from the value that was last written to the Watchdog Count Register. This bit always reads as zero. Setting this bit has no effect when DevB:0xA8[WDTHALT] is set or RSTOP is cleared.
6:4	Read-only.
3	<b>WDA_ALIAS. Watchdog Disable Alias.</b> Read-only. This bit reflects the state of the watchdog timer hardware and is an alias of DevB:0xA8[WDTHALT]. 1 = Watchdog timer functionality disabled. 0=Watchdog timer functionality enabled.
2	<b>WACT. Watchdog Action.</b> Read-write. This bit determines the action to be taken when the watchdog timer expires. 0 = system reset; 1 = system power off.
1	<b>WFIR. When set, the watchdog timer expired and caused the current restart.</b> Read-write. This bit is cleared for any restart that is not caused by the watchdog timer firing. If the Watchdog Action bit is set to 1 (system power off) and the watchdog timer fires, forcing a shutdown, the bit will be cleared on the next power up.
0	<b>RSTOP.</b> Read-write. This bit is used to control or indicate whether the watchdog timer is in the Running or Stopped state.  1 = Watchdog timer is in the Running state. 0 = Watchdog timer is in the Stopped state.  If the watchdog timer is in the Stopped state and a 1 is written to bit 0, the watchdog timer moves to the Running state but a count interval is not started until a 1 is written to bit 7.  If the watchdog timer is in the Running state, writing a 1 to bit 0 has no effect.

**Watchdog Timer Count Register****WDT08**

Default: 0000 0000h.

Attribute: see below.

Bits	Description
31:16	Read-only.
15:0	<b>WCD. Watchdog Count Data.</b> Read-write. This defines the countdown time for the counter in seconds. Reading this register results in the current counter value. Writing to the register has no effect until a one is written to the watchdog trigger bit of the Watchdog Control/Status Register. All bits have to be written by one access.

**4.4.7 High Precision Event Timer Registers**

The HPET Registers are non-enumerable memory mapped I/O registers. The base address is controlled by DevB:0xA0. All registers can be accessed by 32 bit memory accesses.



**Table 56. HPET Registers**

Address	Mnemonic	Name	Default
00h	HPET00	HPET General Capabilities and ID Register	0429_B17F_1022_82XXh
10h	HPET10	HPET General Config Register	0000_0000_0000_0000h
20h	HPET20	HPET General Interrupt Status Register	0000_0000_0000_0000h
F0h	HPETF0	HPET Main Counter Register	0000_0000_0000_0000h
100h	HPET100	HPE Timer 0 Configuration and Capabilities Register	000F_DEFA_0000_0010h
108h	HPET108	HPE Timer 0 Compare Register	0000_0000_FFFF_FFFFh
120h	HPET120	HPE Timer 1 Configuration and Capabilities Register	000F_DEFA_0000_0000h
128h	HPET128	HPE Timer 1 Compare Register	0000_0000_FFFF_FFFFh
140h	HPET140	HPE Timer 2 Configuration and Capabilities Register	000F_DEFA_0000_0000h
148h	HPET148	HPE Timer 2 Compare Register	0000_0000_FFFF_FFFFh

**HPET General Capabilities and ID Register****HPET00**

Default: 0429 B17F 1022\_82XXh

Attribute: Read-only.

Bits	Description
63:32	<b>PERIOD. Main Counter Clock Period in femtoseconds.</b>
31:16	<b>VENDID. Vendor ID.</b> Hardwired to 1022h (AMD).
15	<b>LEGSUP.</b> Hardwired to 1b to flag that legacy interrupt routing is supported.
14	Reserved. Hardwired to 0b.
13	<b>SIZE.</b> Hardwired to 0b to flag 32 bit main counter.
12:8	<b>NUMB.</b> Hardwired to 010b means 3 implemented Comparators.
7:0	<b>REVID. Hardware Revision ID.</b> The value of this register is revision-dependent.

**HPET General Config Register****HPET10**

Default: 0000\_0000\_0000\_0000h

Attribute: See below.

Bits	Description
63:2	Reserved. Read-only. Hardwired to 0h.
1	<b>LIEN. Legacy IRQ Routing Enable.</b> Read-write. 0: Use programmed value for IRQ routing 1: C0 delivers to IRQ0/INTIN2 instead of PIT, C1 delivers to IRQ8/INTIN8 instead of RTC periodic timer interrupt. RTC Alarm still delivers to SCI.
0	<b>GIEN. Global Interrupt Enable.</b> Read-write. This bus must be set to enable any of the timers to generate interrupts. 1: Allow Counter to Run, Allow Compare Interrupts if enabled 0: Halt Counter, Disable all Compare Interrupts

**HPET General Interrupt Status Register****HPET20**

Default: 0000\_0000\_0000\_0000h

Attribute: See below.

Bits	Description
63:3	Reserved. Read-only. These bits are hardwired to 0.
2	<b>C2_STS. Comparator 2 Interrupt Active.</b> Read-write. If set to level-triggered mode, this bit defaults to 0. It is set by hardware if the corresponding interrupt is active. Once the bit is set, it can be cleared by software writing a 1 to this bit. Writes to 0 have no effect. If 1, a write to 0 doesn't clear this bit. In edge sensitive mode this bit has to be ignored by software, software should always write a 0 to this bit.
1	<b>C1_STS. Comparator 1 Interrupt Active.</b> Read-write. If set to level-triggered mode, this bit defaults to 0. It is set by hardware if the corresponding interrupt is active. Once the bit is set, it can be cleared by software writing a 1 to this bit. Writes to 0 have no effect. If 1, a write to 0 doesn't clear this bit. In edge sensitive mode this bit has to be ignored by software, software should always write a 0 to this bit.
0	<b>C0_STS. Comparator 0 Interrupt Active.</b> Read-write. If set to level-triggered mode, this bit defaults to 0. It is set by hardware if the corresponding interrupt is active. Once the bit is set, it can be cleared by software writing a 1 to this bit. Writes to 0 have no effect. If 1, a write to 0 doesn't clear this bit. In edge sensitive mode this bit has to be ignored by software, software should always write a 0 to this bit.

**HPET Main Counter Register****HPETF0**

Default: 0000\_0000\_0000\_0000h

Attribute: See below.

Bits	Description
63:32	Reserved. Read-only. These bits are hardwired to 0000_0000h.
31:0	<b>MAINC.</b> Read-write. These bits represent the content of the main counter. Writes to this register should only be done while the counter is halted.

**HPE Timer 0 Configuration and Capabilities Register****HPET100**

Default: 000F\_DEFA\_0000\_0010h

Attribute: See below.

Bits	Description
63:32	<b>T0_ROUTE_CAP.</b> These bits indicate to which interrupts in the IOAPIC this timer's interrupt can be routed. Each bit represents one interrupt. Bit63 represents interrupt 31, bit 32 interrupt0. 63:52: Read-only. Hardwired to 000000000000b IOAPIC does not support Interrupts 31 down to 20. 51:46: Read-writeOnce. These bits indicate if routing to INTIN19 down to INTIN14 of IOAPIC is allowed. 45: Read-only. This bit is hardwired to 0b. Routing to INTIN13 of IOAPIC is not allowed. 44:41: Read-writeOnce. These bits indicate if routing to INTIN12 down to INTIN9 of IOAPIC is allowed. 40: Read-only. This bit is hardwired to 0b. Routing to INTIN8 of IOAPIC is not allowed. 39:35: Read-writeOnce. These bits indicate if routing to INTIN7 down to INTIN3 of IOAPIC is allowed. 34: Read-only. This bit is hardwired to 0b. Routing to INTIN2 of IOAPIC is not allowed. 33: Read-writeOnce. This bit indicates if routing to INTIN1 of IOAPIC is allowed. 32: Read-only. This bit is hardwired to 0b. Routing to INTIN0 of IOAPIC is not allowed.
31:16	Reserved. Read-only. Hardwired to 0000h. Software should only write a 0000h to these bits.
15	<b>T0_FSB_CAP.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
14	<b>T0_FSB_EN.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
13:9	<b>T0_INT_ROUTE.</b> Read-write. This 5-bit field defines the routing to the IOAPIC. If the written value to this register doesn't match bits[63:32] of this register, then the most significant enabled interrupt is written instead. If HPET10[LIEN] or bit14 of this register is set then these bits have no effect.
8	<b>M32.</b> Read-only. This bit is hardwired to 0b, because the timer doesn't support 64 bit mode and this bit is not needed.
7	Reserved. Read-only. Hardwired to 0b. Software should only write a 0b to this bit.
6	<b>SETVAL.</b> Read-write. Timer 0 Value Set. Software can set this bit to write the accumulator register of periodic timer. Software should not set this bit, if this timer is not configured as periodic timer. Software has not clear this bit. It clears itself after writing HPET108.
5	<b>SIZE.</b> Read-only. Hardwired to 0b to indicate that the timer is 32 bit only.
4	<b>TOPCAP.</b> Read-only. Hardwired to 1b to flag, that this timer supports the periodic interrupt.

Bits	Description (Continued)
3	<b>T0PEN. Timer 0 Periodic Mode Enable.</b> Read-write. If T0PCAP is 0, this bit is always 0. Writes have no effect. If T0PCAP is 1, this bits enables the period mode of timer 0. 0 = disables timer 0 periodic mode 1 = enables timer 0 periodic mode
2	<b>T0IEN. Timer 0 Interrupt Enable.</b> Read-write. This bit must be set to enable timer 0 to cause an interrupt when it times out. If this bit is 0, the timer can still count and generate the appropriate status bits, but does not cause an interrupt.
1	<b>T0ITYPE. Timer 0 Interrupt Type.</b> Read-write. 0 = The timer interrupt is edge triggered. Each new interrupt generates a new edge. 1 = The timer interrupt is level triggered. The interrupt is held active until the corresponding bit in the HPET20[C0_STS] register is cleared. If a new interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	Reserved. Read-only. Hardwired to 0. Software should only write a 0 to this bit.

**HPE Timer 0 Compare Register****HPET108**

Default: 0000\_0000\_FFFF\_FFFFh

Attribute: See below.

Bits	Description
63:32	Reserved. Read-only. These bits are hardwired to 0000_0000h.
31:0	<b>T0COMP. Timer 0 Comparator Value.</b> Read-write. In non-periodic mode (HPET100[T0PEN]=0) a write to this register sets the comparator value. Reads to this register return the value of the comparator. When the main counter equals T0COMP the corresponding interrupt is generated, if enabled. The value in this register does not change based on the generated interrupt. In periodic mode (HPET100[T0PEN]=1) this register consists of two registers, one for saving the last software-written comparator value and one for the accumulator value. If HPET100[SETVAL] is High a write to this address causes a write of this accumulator register, if HPET100[SETVAL] is Low a write to this address causes a change of the last software-written comparator value. The read value is the value of the accumulator register. When the main counter equals the accumulator register value, the last software-written comparator value is added to the accumulator, and the corresponding interrupt is generated, if enabled.

## HPE Timer 1 Configuration and Capabilities Register

HPET120

Default: 000F\_DEFA\_0000\_0000h

Attribute: See below.

Bits	Description
63:32	<b>T1_ROUTE_CAP.</b> These bits indicate to which interrupts in the IOAPIC this timer's interrupt can be routed. Each bit represents one interrupt. Bit63 represents interrupt 31, bit 32 interrupt0. 63:52: Read-only. Hardwired to 000000000000b IOAPIC doesn't support Interrupts 31 down to 20. 51:46: Read-writeOnce. These bits indicate if routing to INTIN19 down to INTIN14 of IOAPIC is allowed. 45: Read-only. This bit is hardwired to 0b. Routing to INTIN13 of IOAPIC is not allowed. 44:41: Read-writeOnce. These bits indicate if routing to INTIN12 down to INTIN9 of IOAPIC is allowed. 40: Read-only. This bit is hardwired to 0b. Routing to INTIN8 of IOAPIC is not allowed. 39:35: Read-writeOnce. These bits indicate if routing to INTIN7 down to INTIN3 of IOAPIC is allowed. 34: Read-only. This bit is hardwired to 0b. Routing to INTIN2 of IOAPIC is not allowed 33: Read-writeOnce. This bit indicates if routing to INTIN1 of IOAPIC is allowed. 32: Read-only. This bit is hardwired to 0b. Routing to INTIN0 of IOAPIC is not allowed.
31:16	Reserved. Read-only. Hardwired to 0000h. Software should only write a 0000h to these bits.
15	<b>T1_FSB_CAP.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
14	<b>T1_FSB_EN.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
13:9	<b>T1_INT_ROUTE.</b> Read-write. This 5-bit field defines the routing to the IOAPIC. If the written value to this register doesn't match bits[63:32] of this register, then the most significant enabled interrupt is written instead. If HPET10[LIEN] or bit14 of this register is set then these bits have no effect.
8	<b>M32.</b> Read-only. This bit is hardwired to 0, because the timer doesn't support 64 bit mode and this bit is not needed.
7	Reserved. Read-only. Hardwired to 0.
6	<b>SETVAL.</b> Read-only. Hardwired to 0.
5	<b>SIZE.</b> Read-only. Hardwired to 0 to indicate that the timer is 32 bit only.
4	<b>T1PCAP.</b> Read-only. Hardwired to 0 to flag, that this timer doesn't support the periodic interrupt.
3	<b>T1PEN.</b> Read-only. This bit is always 0.
2	<b>T1IEN. Timer 1Interrupt Enable.</b> Read-write. This bit must be set to enable timer 0 to cause an interrupt when it times out. If this bit is 0, the timer can still count and generate the appropriate status bits, but does not cause an interrupt.
1	<b>T1ITYPE. Timer 0 Interrupt Type.</b> Read-write. 0 = The timer interrupt is edge triggered. Each new interrupt generates a new edge. 1 = The timer interrupt is level triggered. The interrupt is held active until the corresponding bit in the HPET20[C1_STS] register is cleared. If a new interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	Reserved. Read-only. Hardwired to 0. Software should only write a 0 to this bit.

**HPE Timer 1 Compare Register****HPET128**

Default: 0000\_0000\_FFFF\_FFFFh

Attribute: See below.

Bits	Description
63:32	Reserved. Read-only. These bits are hardwired to 0.
31:0	<b>T1COMP. Timer 1 Comparator Value.</b> Read-write. Reads to this register return the value of the comparator.  A write to this register sets the comparator value. When the main counter equals the value last written, the corresponding interrupt is generated, if enabled. The value in this register does not change based on the generated interrupt.

**HPE Timer 2 Configuration and Capabilities Register****HPET140**

Default: 000F\_DEFA\_0000\_0000h

Attribute: See below.

Bits	Description
63:32	<b>T2_ROUTE_CAP.</b> These bits indicate to which interrupts in the IOAPIC this timer's interrupt can be routed. Each bit represents one interrupt. Bit63 represents interrupt 31, bit 32 interrupt0. 63:52: Read-only. Hardwired to 000000000000b IOAPIC doesn't support Interrupts 31 down to 20. 51:46: Read-writeOnce. These bits indicate if routing to INTIN19 down to INTIN14 of IOAPIC is allowed. 45: Read-only. This bit is hardwired to 0b. Routing to INTIN13 of IOAPIC is not allowed. 44:41: Read-writeOnce. These bits indicate if routing to INTIN12 down to INTIN9 of IOAPIC is allowed. 40: Read-only. This bit is hardwired to 0b. Routing to INTIN8 of IOAPIC is not allowed. 39:35: Read-writeOnce. These bits indicate if routing to INTIN7 down to INTIN3 of IOAPIC is allowed. 34: Read-only. This bit is hardwired to 0b. Routing to INTIN2 of IOAPIC is not allowed 33: Read-writeOnce. This bit indicates if routing to INTIN1 of IOAPIC is allowed. 32: Read-only. This bit is hardwired to 0b. Routing to INTIN0 of IOAPIC is not allowed.
31:16	Reserved. Read-only. Hardwired to 0000h. Software should only write a 0000h to these bits.
15	<b>T2_FSB_CAP.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
14	<b>T2_FSB_EN.</b> Read-only. This bit is hardwired to 0 to flag that no FSB routing is possible.
13:9	<b>T2_INT_ROUTE.</b> Read-write. This 5-bit field defines the routing to the IOAPIC. If the written value to this register doesn't match bits[63:32] of this register, then the most significant enabled interrupt is written instead. If HPET10[LIEN] or bit14 of this register is set then these bits have no effect.
8	<b>M32.</b> Read-only. This bit is hardwired to 0, because the timer doesn't support 64 bit mode and this bit is not needed.
7	Reserved. Read-only. Hardwired to 0. Software should only write a 0 to this bit.
6	<b>SETVAL.</b> Read-only. Hardwired to 0.
5	<b>SIZE.</b> Read-only. Hardwired to 0 to indicate that the timer is 32 bit only.
4	<b>T2PCAP.</b> Read-only. Hardwired to 0 to flag, that this timer doesn't support the periodic interrupt.

Bits	Description (Continued)
3	<b>T2PEN.</b> Read-only. This bit is always 0.
2	<b>T2IEN. Timer 1 Interrupt Enable.</b> Read-write. This bit must be set to enable timer 0 to cause an interrupt when it times out. If this bit is 0, the timer can still count and generate the appropriate status bits, but does not cause an interrupt.
1	<b>T2ITYPE. Timer 0 Interrupt Type.</b> Read-write. 0 = The timer interrupt is edge triggered. Each new interrupt generates a new edge. 1 = The timer interrupt is level triggered. The interrupt is held active until the corresponding bit in the HPET20[C2_STS] register is cleared. If a new interrupt occurs before the interrupt is cleared, the interrupt remains active.
0	Reserved. Read-only. Hardwired to 0. Software should only write a 0 to this bit.

**HPE Timer 2 Compare Register****HPET148**

Default: 0000\_0000\_FFFF\_FFFFh

Attribute: See below.

Bits	Description
63:32	Reserved. Read-only. These bits are hardwired to 0.
31:0	<b>T2COMP. Timer 2 Comparator Value.</b> Read-write. Reads to this register return the value of the comparator.  A write to this register sets the comparator value. When the main counter equals the value last written, the corresponding interrupt is generated, if enabled. The value in this register does not change based on the generated interrupt.

**4.4.8 Real-Time Clock Registers****Real-Time Clock Legacy Indexed Address****RTC70**

*Note:* RTC70[6:0] and RTC72 occupy the same physical register. After a write to RTC70, RTC72 reads back {0b, RTC70[6:0]}; after a write to RTC72, reads of RTC72 provide all 8 bits written. RTC70 and RTC72 are on the VDD\_CORE power plane; they are not preserved in the STR, STD, SOFF, or MOFF states.

I/O mapped (fixed); offset: 70h.

Default: 00.

Attribute: Write-only.

Bits	Description
7	<b>NMIDIS. NMI disable.</b> 1=Sources of NMIs (from serial IRQ logic and SERR_L pin) are disabled from being able to generate NMI interrupts. Note: the state of this register is read accessible through DevB:0x41[NMIDIS].
6:0	<b>RTCADDR. Real-time clock address.</b> Specifies the address of the real-time clock CMOS RAM. The data port associated with this index is RTC71. Only the lower 128 bytes of the CMOS RAM are accessible through RTC70 and RTC71.

**Real-Time Clock Legacy Data Port****RTC71**

I/O mapped (fixed); offset: 0071h.

Attribute: Read-write.

Bits	Description
7:0	<b>RTCDATA. Real-time clock data.</b> This is the data port for accesses to the real-time clock CMOS RAM that is indexed by RTC70.

**Real-Time Clock 256-Byte Address and Data Port****RTC72 and RTC73**

I/O mapped (fixed); offsets: 0072h and 0073h.

Attribute: Read-write.

These two 8-bit ports are similar to RTC70 and RTC71. However, RTC72, the address register, provides the full eight bits needed to access all 256 bytes of CMOS RAM. RTC73, the data port, provides access to the CMOS data indexed by RTC72. See RTC70 for details about reading RTC72.

**Real-Time Clock Alarm and Century Registers****RTC offsets 7F:7D**

RTC indexed address space (indexed by RTC70); offsets: 7Dh, 7Eh, and 7Fh.

Attribute: Read-write.

The day alarm, month alarm, and centenary value are stored in CMOS RAM indexed address space.

RTC70; CMOS RAM Offset	Function	Range for Binary Mode	Range for BCD Mode
7Dh	Date alarm	01h-1Fh	01h-31h
7Eh	Month alarm	01h-0Ch	01h-12h
7Fh	Century field	13h-63h	19h-99h

Bits[7:6] of the date alarm (offset 7Dh) are reserved; writes to them have no effect and they always read back as zero. To place the RTC into 24-hour alarm mode, an invalid code must be written to bits[5:0] of the date alarm byte (any value other than 01h to 31h for BCD mode or 01h to F1h for hex mode). Refer to the ACPI specification for details of the month alarm field (offset 7Eh) and century field (offset 7Fh).

## 4.5 Enhanced IDE Controller Registers

### 4.5.1 Enhanced IDE Configuration Registers (DevB:1xXX)

These registers are located in PCI configuration space on the primary PCI interface, in the second device (device B), function 1. See Section 4.1.2 on page 136 for a description of the register naming convention.



**EIDE Controller Vendor And Device ID Register****DevB:1x00**

Default: 7469 1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>EIDE controller device ID.</b>
15:0	<b>Vendor ID.</b>

**EIDE Controller Status And Command Register****DevB:1x04**

Default: 0200 0000h.

Attribute: See below.

Bits	Description
31:27	Read-only. These bits is fixed in the Low state.
26:25	<b>DEVSEL timing.</b> Read-only. These bits are fixed at STATUS[10:9] = 01b. This specifies "medium" timing as defined by the PCI specification.
24:3	Read-only. These bits are fixed at their default values.
2	<b>BMEN. Bus master enable.</b> Read-write. 1=Enables IDE bus master capability.
1	<b>Memory space enable.</b> Read-only. This bit is fixed in the Low state.
0	<b>IOEN. I/O space enable.</b> Read-write. 1=Enables access to the I/O space for the IDE controller.

**EIDE Revision ID, Programming Interface, Sub Class and Base Registers****DevB:1x08**

Default: 0101 8AXXh see below.

Attribute: See below.

Bits	Description
31:24	<b>BASECLASS.</b> Read-only. These bits are fixed at 01h indicating a mass storage device.
23:16	<b>SUBCLASS.</b> Read-only. These bits are fixed at 01h indicating an IDE controller.
15	<b>PROGIF[7]. Master IDE capability.</b> Read-only. This bit is fixed in the High state.
14:12	<b>PROGIF[6:4].</b> Read-only. These bits are fixed in the Low state.
11	<b>PROG IF[3]. Secondary native/compatibility mode selectable.</b> Read-only. This is High to indicate that PROGIF[2] is read-write.

Bits	Description (Continued)
10	<b>PROGIF[2]. Secondary native mode.</b> Read-write. 0=Compatibility mode for secondary port; DevB:1x18 and DevB:1x1C are ignored and not visible; address decode is based on legacy addresses 170h-177h, 376h; DevB:1x3C[7:0] Read-only zeros; DevB:1x3C[15:8] = 00h; IRQ15 may be used by the IDE controller. 1=Native mode; DevB:1x18 and DevB:1x1C are visible and used for address decode; DevB:1x3C[7:0] read-write; DevB:1x3C[15:8] = 01h; IRQ15 mapped to PIRQA per Section 3.4.2.1 on page 41 and used exclusively by the secondary IDE port.
9	<b>PROGIF[1]. Primary native/compatibility mode selectable.</b> Read-only. This is High to indicate that PROGIF[0] is read-write.
8	<b>PROGIF[0]. Primary native mode.</b> Read-write. 0=Compatibility mode for primary port; DevB:1x10 and DevB:1x14 are ignored and not visible; address decode is based on legacy addresses 1F0h-1F7h, 3F6h; DevB:1x3C[7:0] Read-only zeros; DevB:1x3C[15:8] = 00h; IRQ14 may be used by the IDE controller. 1=Native mode; DevB:1x10 and DevB:1x14 are visible and used for address decode; DevB:1x3C[7:0] read-write; DevB:1x3C[15:8] = 01h; IRQ14 mapped to PIRQA per Section 3.4.2.1 on page 41 and used exclusively by the primary IDE port.
7:0	<b>REVISIONID.</b> Read-only. EIDE Controller silicon revision. The value of this register is revision-dependent.

**EIDE Controller BIST, Header and Latency Register****DevB:1x0C**

Default: 0000 0000h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. These bits fixed at their default values.
23:16	<b>HEADER.</b> Read-only. These bits fixed at their default values.
15:8	<b>LATENCY.</b> Read-write. This field controls no hardware.
7:0	<b>CACHE.</b> Read-only. These bits fixed at their default values.

**EIDE Controller Primary Command Base Address****DevB:1x10**

Default: 0000 01F1h.

Attribute: See below.

Bits	Description
31:3	<b>BASE[31:3] Port Address.</b> Read-write. These bits specify an 8-byte I/O address space that maps to the ATA-compliant command register set for the primary port (legacy I/O space 1F0h-1F7h). Note: when DevB:1x08[8] is Low, the primary port is in compatibility mode and this register is ignored and not visible (reads as 0000_0000h).
2:0	Read-only. 001b.

**EIDE Controller Primary Control Base Address****DevB:1x14**

Default: 0000 03F5h.

Attribute: See below.

Bits	Description
31:2	<b>BASE[31:2] Port Address.</b> Read-write. These bits specify a 4-byte I/O address space that maps to the ATA-compliant control register set for the primary port (legacy I/O space 3F4h-3F7h). Note: when DevB:1x08[8] is Low, the primary port is in compatibility mode and this register is ignored and not visible (reads as 0000_0000h). Note: only byte 2 (legacy I/O address 3F6h) of this space is used.
1:0	Read-only. 01b.

**EIDE Controller Secondary Command Base Address****DevB:1x18**

Default: 0000 0171h.

Attribute: See below.

Bits	Description
31:3	<b>BASE[31:3] Port Address.</b> Read-write. These bits specify an 8-byte I/O address space that maps to the ATA-compliant command register set for the secondary port (legacy I/O space 170h-177h). Note: when DevB:1x08[10] is Low, the secondary port is in compatibility mode and this register is ignored and not visible (reads as 0000_0000h).
2:0	Read-only. 001b.

**EIDE Controller Secondary Control Base Address****DevB:1x1C**

Default: 0000 0375h.

Attribute: See below.

Bits	Description
31:2	<b>BASE[31:2] Port Address.</b> Read-write. These bits specify a 4-byte I/O address space that maps to the ATA-compliant control register set for the secondary port (legacy I/O space 374h-377h). Note: when DevB:1x08[10] is Low, the secondary port is in compatibility mode and this register is ignored and not visible (reads as 0000_0000h). Note: only byte 2 (legacy I/O address 376h) of this space is used.
1:0	Read-only. 01b.

**EIDE Controller Bus Master Control Registers Base Address****DevB:1x20**

Default: 0000 CC01h.

Attribute: See below.

Bits	Description
31:4	<b>BASE[31:4] Port Address.</b> Read-write. This field specifies the 16-byte I/O address space that maps to the EIDE controller register set that is compliant with the SFF-8038i specification (Bus Master Programming Interface for IDE ATA Controllers), IBMx.
3:0	Read-only. 0001b.

**EIDE Subsystem ID and Subsystem Vendor ID Register****DevB:1x2C**

Default: 0000 0000h.

Attribute: Read-only.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> This field is write accessible through DevB:1x70.
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> This field is write accessible through DevB:1x70.

**EIDE Controller Interrupt Line, Interrupt Pin, Min. Grant, Max Latency Register****DevB:1x3C**

Default: 0000 00FFh.

Attribute: See below.

Bits	Description
31:24	<b>MAX LATENCY.</b> Read-only. These bits are fixed at their default values.
23:16	<b>MIN GNT.</b> Read-only. These bits are fixed at their default values.
15:8	<b>INTERRUPT PIN.</b> Read-only. When either DevB:1x08[8] or DevB:1x08[10] is High, then field reads as 01h. When they are both Low, then it reads as 00h.
7:0	<b>INTERRUPT LINE.</b> This register is either read-write or read-only based on the state of DevB:1x08[10,8]. When either DevB:1x08[8] or DevB:1x08[10] is High, then this is a read-write register. When they are both Low, then it is a read-only register, reading FFh.

**EIDE Controller Configuration Register****DevB:1x40**

Default: 0000 04?0h.

Attribute: See below.

Bits	Description
31:24	Reserved.
23:20	<b>RW.</b> Read-write. These bits control no hardware.
19:16	<b>CABLE.</b> Read-write. These bits are intended to be programmed by BIOS to specify the cable type of each of the IDE drives to the driver software. 1=High speed 80-pin cable is present. The bits specify the following drive: Bit[16]: primary master. Bit[18]: secondary master. Bit[17]: primary slave. Bit[19]: secondary slave.
15	<b>RW.</b> Read-write. This bit controls no hardware.
14	<b>PRIPWB. Primary post write buffer.</b> Read-write. 1=The primary port posted-write buffer for PIO modes is enabled. Note: only 32-bit writes to the data register are allowed when this bit is set.
13	<b>RW.</b> Read-write. This bit is read-write accessible through software; it controls no hardware.
12	<b>SECPWB. Secondary post write buffer.</b> Read-write. 1=The secondary port posted-write buffer for PIO modes is enabled. Note: only 32-bit writes to the data register are allowed when this bit is set.
11	<b>PHYOR. ATA PHY override.</b> Read-write. 1=ATA PHY slew rate controlled by DevB:1x40[PHYORSEL]. 0=ATA PHY slew rate controlled by PHY select circuitry observed in DevB:1x40[PHYSEL].
10	<b>RW.</b> Read-write. This bit controls no hardware.
9:8	<b>PHYORSEL. ATA PHY override select.</b> Read-write. These bits specify the override value of the PHY speed select, as specified by DevB:1x40[PHYSEL]. This field is ignored when DevB:1x40[PHYOR]=0.
7	Reserved. This bit is fixed to 0.

Bits	Description (Continued)
6:5	<b>PHYSEL. PHY speed select.</b> Read-only. These specify the selected PHY speed selection as follows: 00b=bigger PHY resistor (fast corner); 10b=medium PHY resistor; x1b=smaller PHY resistor (slow corner). The power-up default for these bits is device specific.
4:2	Reserved. These bits are fixed to 0.
1	<b>PRIEN. Primary channel enable.</b> Read-write. 1=The primary port of the EIDE controller is enabled.
0	<b>SECEN. Secondary channel enable.</b> Read-write. 1=The secondary port of the EIDE controller is enabled.

**EIDE Controller Drive Timing Control Register****DevB:1x48**

This register specifies timing for PIO data transfers (1F0h and 170h or the native mode equivalents) and multi-word DMA transfers. The value in each 4-bit field, plus one, specifies a time in 30 nanosecond PCI clocks.

**Notes:**

1. For command and control transfers (1F1h-1F7h, 171h-177h, 3F6h and 376h) see DevB:1x4C.
2. The default state, A8h, results in a recovery time of 270ns and an active pulse width of 330ns for a 30ns PCI clock (total cycle time = 600ns) which corresponds to ATA PIO Mode 0.
3. PIO modes are controlled through DevB:1x48 and DevB:1x4C. To set the timing associated with the various modes, DevB:1x4C should be left at its default value and the appropriate byte of DevB:1x48 should be programmed as follows—mode 0=A8h; mode 1=65h; mode 2=42h; mode 3=22h; mode 4=20h.

Default: A8A8 A8A8h.

Attribute: Read-write.

Bits	Description
31:28	<b>PD0PW[3:0]. Primary drive 0 data DIOR_L/DIOW_L active pulse width.</b>
27:24	<b>PD0RT[3:0]. Primary drive 0 data DIOR_L/DIOW_L minimum recovery time.</b>
23:20	<b>PD1PW[3:0]. Primary drive 1 data DIOR_L/DIOW_L active pulse width.</b>
19:16	<b>PD1RT[3:0]. Primary drive 1 data DIOR_L/DIOW_L minimum recovery time.</b>
15:12	<b>SD0PW[3:0]. Secondary drive 0 data DIOR_L/DIOW_L active pulse width.</b>
11:8	<b>SD0RT[3:0]. Secondary drive 0 data DIOR_L/DIOW_L minimum recovery time.</b>
7:4	<b>SD1PW[3:0]. Secondary drive 1 data DIOR_L/DIOW_L active pulse width.</b>
3:0	<b>SD1RT[3:0]. Secondary drive 1 data DIOR_L/DIOW_L minimum recovery time.</b>

**EIDE Controller Cycle Time and Address Setup Time Register****DevB:1x4C**

For bits[7:0] the value in each 2-bit field, plus one, specifies the address setup time in 30 nanosecond PCI clocks; this applies to all PIO and multi-word DMA cycles. For bits[31:16] the value in each 4-bit field, plus one, specifies the time in 30 nanosecond PCI clocks; this applies to address ports 1F1h-1F7h, 171h-177h, 3F6h, and 376h; for 1F0h and 170h, see DevB:1x48.

Default: FFFF 00FFh.

Attribute: Read-write.

Bits	Description
31:28	<b>PXPW[3:0]. Primary command/control DIOR_L/DIOW_L active pulse width.</b>
27:24	<b>PXRT[3:0]. Primary command/control DIOR_L/DIOW_L recovery time.</b>
23:20	<b>SXPW[3:0]. Secondary command/control DIOR_L/DIOW_L active pulse width.</b>
19:16	<b>SXRT[3:0]. Secondary command/control DIOR_L/DIOW_L recovery time.</b>
15:8	Reserved.
7:6	<b>P0ADD[1:0]. Primary drive 0 address setup time.</b>
5:4	<b>P1ADD[1:0]. Primary drive 1 address setup time.</b>
3:2	<b>S0ADD[1:0]. Secondary drive 0 address setup time.</b>
1:0	<b>S1ADD[1:0]. Secondary drive 1 address setup time.</b>

**EIDE Controller UDMA Extended Timing Control Register****DevB:1x50**

The definition of each of the four 8-bit fields in this register are identical; they apply to different drives.

Default: 0303 0303h.

Attribute: Read-write.

Bits	Description
31:24	<b>P0UDMA. Primary drive 0 UDMA timing control.</b>
23:16	<b>P1UDMA. Primary drive 1 UDMA timing control.</b>
15:8	<b>S0UDMA. Secondary drive 0 UDMA timing control.</b>
7:0	<b>S1UDMA. Secondary drive 1 UDMA timing control.</b>

Here is the definition of each 8-bit field:

Bits	Description																											
31, 23, 15, 7	<b>[P0, P1, S0, S1]ENMODE. [Primary, secondary] drive [0, 1] ultra-DMA mode enable method.</b> 1=Enable ultra-DMA by setting bit 6 of this 8-bit register. 0=Enable ultra-DMA by detecting the "Set Feature" ATA command.																											
30, 22, 14, 6	<b>[P0, P1, S0, S1]UDMAEN. [Primary, secondary] drive [0, 1] ultra-DMA mode enable.</b> 1=Ultra-DMA mode is enabled.																											
29:27, 21:19, 13:11, 5:3	Reserved.																											
26:24, 18:16, 10:8, 2:0	<b>[P0, P1, S0, S1]CYCT[2:0]. [Primary, secondary] drive [0,1] Cycle Time.</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>CYCT[2:0]</th> <th>Ultra-DMA mode</th> <th>Cycle time</th> </tr> </thead> <tbody> <tr> <td>000b</td> <td>UDMA mode 2 (ATA-33)</td> <td>60 nanoseconds</td> </tr> <tr> <td>001b</td> <td>UDMA mode 1</td> <td>80 nanoseconds</td> </tr> <tr> <td>010b</td> <td>UDMA mode 0</td> <td>120 nanoseconds</td> </tr> <tr> <td>011b</td> <td>Slow UDMA mode 0</td> <td>150 nanoseconds</td> </tr> <tr> <td>100b</td> <td>UDMA mode 3 (ATA-44)</td> <td>45 nanoseconds</td> </tr> <tr> <td>101b</td> <td>UDMA mode 4 (ATA-66)</td> <td>30 nanoseconds</td> </tr> <tr> <td>110b</td> <td>UDMA mode 5 (ATA-100)</td> <td>20 nanoseconds</td> </tr> <tr> <td>111b</td> <td>UDMA mode 6 (ATA-133)</td> <td>15 nanoseconds</td> </tr> </tbody> </table>	CYCT[2:0]	Ultra-DMA mode	Cycle time	000b	UDMA mode 2 (ATA-33)	60 nanoseconds	001b	UDMA mode 1	80 nanoseconds	010b	UDMA mode 0	120 nanoseconds	011b	Slow UDMA mode 0	150 nanoseconds	100b	UDMA mode 3 (ATA-44)	45 nanoseconds	101b	UDMA mode 4 (ATA-66)	30 nanoseconds	110b	UDMA mode 5 (ATA-100)	20 nanoseconds	111b	UDMA mode 6 (ATA-133)	15 nanoseconds
CYCT[2:0]	Ultra-DMA mode	Cycle time																										
000b	UDMA mode 2 (ATA-33)	60 nanoseconds																										
001b	UDMA mode 1	80 nanoseconds																										
010b	UDMA mode 0	120 nanoseconds																										
011b	Slow UDMA mode 0	150 nanoseconds																										
100b	UDMA mode 3 (ATA-44)	45 nanoseconds																										
101b	UDMA mode 4 (ATA-66)	30 nanoseconds																										
110b	UDMA mode 5 (ATA-100)	20 nanoseconds																										
111b	UDMA mode 6 (ATA-133)	15 nanoseconds																										

**EIDE Power Management Register****DevB:1x54**

This register controls the power state of the two IDE ports. When an IDE port is powered up, it is designed to be fully operational. When it is powered down, outputs DADDR[S,P][2:0], DCS1[S,P]\_L, DCS3[S,P]\_L, DDACK[S,P]\_L, DIO[R,W][S,P]\_L and DRST[S,P]\_L are forced Low; DDATA[S,P][15:0], DDRQ[S,P], and DRDY[S,P] are ignored and in high-impedance mode. When transitioning into the power-down state, DRST[S,P]\_L assertion leads control over the rest of the signals by about 1 microsecond. When transitioning into the power-up state, DRST[S,P]\_L deassertion lags control over the rest of the signals by about 1 microsecond.

**Note:** Writes to [S,P]PWRDN cause [S,P]PWRX to be set High until the power state transition is complete; while [S,P]PWRX is High, writes to [S,P]PWRDN are ignored.

Default: 00h.

Attribute: See below.

Bits	Description
7:6	Reserved.
5	<b>SPWRX. Power state transition for secondary IDE port.</b> Read-only. 1=The secondary IDE port is transitioning from either the power-down state to the power up state (if SPWRDN = 0) or from the power-up state to the power-down state (if SPWRDN = 1).
4	<b>SPWRDN. Power down secondary IDE port.</b> Writing a 1 to this field causes the hardware to transition the secondary port from the powered-up state to the powered-down state. Writing a 0 to this field causes the hardware to transition the secondary port from the powered-down state to the powered-up state. When read, this bit reflects the last state written to it; however, it cannot be altered when SPWRX is High.
3:2	Reserved.
1	<b>PPWRX. Power state transition for primary IDE port.</b> Read-only. 1=The primary IDE port is transitioning from either the power-down state to the power up state (if PPWRDN = 0) or from the power-up state to the power-down state (if PPWRDN = 1).
0	<b>PPWRDN. Power down primary IDE port.</b> Writing a 1 to this field causes the hardware to transition the primary port from the powered-up state to the powered-down state. Writing a 0 to this field causes the hardware to transition the primary port from the powered-down state to the powered-up state. When read, this bit reflects the last state written to it; however, it cannot be altered when PPWRX is High.

**Read-Write Register****DevB:1x58**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	<b>RW.</b> Read-write. These bits are read-write accessible through software; they control no hardware.



**EIDE Device and Subsystem ID Read-Write Register****DevB:1x70**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> The value placed in this register is visible in DevB:1x2C[31:16].
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> The value placed in this register is visible in DevB:1x2C[15:0].

**4.5.2 EIDE Bus Master I/O Registers**

These registers are located in I/O space. The base address register is DevB:1x20. See Section 4.1.2 on page 136 for a description of the register naming convention.

These registers comply with SFF-8038i for control of DMA transfers between drives and system memory.

**Primary Bus Master IDE Command Register****IBM0**

Default: 00h.

Attribute: See below.

Bits	Description
7:4	Reserved. These bits are fixed to 0000b.
3	<b>RWCTL. Read or write control.</b> Read-write. 1=Read cycles specified (read from the drive; write to system memory). 0=Write cycles specified.
2:1	Reserved. These bits are fixed to 00b.
0	<b>STSP. Start/stop bus master.</b> Write-only; reads always return 0.

**Primary Bus Master IDE Status Register****IBM2**

Default: 00h.

Attribute: See below.

Bits	Description
7	<b>Simplex only.</b> Read-only. This bit is fixed to 0b. Both bus masters are able to operate independently.
6	<b>DMA1C. Drive 1 DMA capable.</b> Read-write
5	<b>DMA0C. Drive 0 DMA capable.</b> Read-write.
4:3	Reserved. These bits are fixed to 00b.
2	<b>IRQ. Interrupt.</b> Read; set by hardware; write 1 to clear. This bit is set by the rising edge of the IDE interrupt line.
1	<b>ERR. Error.</b> Read; write 1 to clear.
0	<b>ACTV. Bus master IDE active.</b> Read-only.

**Primary Bus Master IDE PRD Table Address Register****IBM4**

Default: 0000 0000h.

Attribute: See below.

Bits	Description
31:2	<b>PPRDADD[31:2]. Primary physical region descriptor table base address.</b> Read-write.
1:0	<b>PPRDADD[1:0].</b> Read-only. These bits are fixed in the Low state.

**Secondary Bus Master IDE Command Register****IBM8**

Default: 00h.

Attribute: See below.

Bits	Description
7:4	Reserved. These bits are fixed to 0000b.
3	<b>RWCTL. Read or write control.</b> Read-write. 1=Read cycles specified (read from the drive; write to system memory). 0=Write cycles specified.
2:1	Reserved. These bits are fixed to 00b.
0	<b>STSP. Start/stop bus master.</b> Write-only; reads always return 0.

**Secondary Bus Master IDE Status Register****IBMA**

Default: 00h.

Attribute: See below.

Bits	Description
7	<b>Simplex only.</b> Read-only. This bit is fixed to 0b. Both bus masters are able to operate independently.
6	<b>DMA1C. Drive 1 DMA capable.</b> Read-write
5	<b>DMA0C. Drive 0 DMA capable.</b> Read-write.
4:3	Reserved. These bits are fixed to 00b.
2	<b>IRQ. Interrupt.</b> Read; set by hardware; write 1 to clear. This bit is set by the rising edge of the IDE interrupt line.
1	<b>ERR. Error.</b> Read; write 1 to clear.
0	<b>ACTV. Bus master IDE active.</b> Read-only.

**Secondary Bus Master IDE PRD Table Address Register****IBMC**

Default: 0000 0000h.

Attribute: See below.

Bits	Description
31:2	<b>SPRDADD[31:2]. Secondary physical region descriptor table base address.</b> Read-write.
1:0	<b>SPRDADD[1:0].</b> Read-only. These bits are fixed in the Low state.

## 4.6 System Management Bus 2.0 Controller Registers

### 4.6.1 System Management Bus Configuration Registers (DevB:2xXX)

#### SMBus Controller Vendor and Device Id

DevB:2x00

Default: 746A\_1022h

Attribute: Read-only.

Bits	Description
31:16	<b>SMBID.</b> Provides the SMBus controller device identification.
15:0	<b>AMDID.</b> Provides AMD's PCI vendor identification.

#### SMBus Controller Command and Status

DevB:2x04

Default: 0200\_0000h

Attribute: See below.

Bits	Description
31:16	<b>STAT.</b> Read-only.
15:0	<b>CMD.</b> CMD[15:1]: Read-only. CMD[0]: I/O Space Enable, IOEN. Read-write. 1b = Access to I/O space for this device is enabled. 0b = Access to I/O space for this device is disabled.

#### SMBus Controller Revision Id and Class Code

DevB:2x08

This register is aliased to DevB:2x40 for write access.

Default: 0C05\_00XXh see below.

Attribute: Read-only.

Bits	Description
31:24	<b>BASECLASS.</b> These bits are fixed at 0Ch indicating a serial bus controller.
23:16	<b>SUBCLASS.</b> These bits are fixed at 05h indicating a SMBus controller.
15:8	<b>PROGIF.</b>
7:0	<b>REVID. SMBus Controller silicon revision.</b> The value of this register is revision-dependent.

**SMBus Controller BIST, Header and Latency****DevB:2x0C**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only.
23:16	<b>HEADER.</b> Read-only.
15:8	<b>LATENCY.</b> Read-write. The latency timer defines the minimum amount of time, in PCI clock cycles, that the bus master can retain ownership of the bus.
7:0	Read-only.

**SMBus Controller Command Interface Base Address****DevB:2x10**

Default: 0000\_0001h

Attribute: See below.

Bits	Description
31:5	<b>CIBA. Command Interface Base Address.</b> Read-write. These bits are used in the I/O space decode of the SC interface registers. There is a maximum block size of 32 bytes for this base address.
4:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is fixed to 1, indicating I/O address space.

**SMBus Controller Subsystem and Subsystem Vendor Id****DevB:2x2C**

This register is aliased to DevB:2x44 for write access.

Default: 0000\_0000h

Attribute: Read-only.

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b>
15:0	<b>SUBVENID. Subsystem Vendor ID.</b>

**SMBus Controller Interrupt Line and Interrupt Pin****DevB:2x3C**

Default: 0000\_0400h

Attribute: See below.

Bits	Description
31:11	Reserved.
10:8	<b>INTPIN.</b> Read-only. Indicates which PCI interrupt pin is used for the SMBus Controller interrupt. Hardwired to 100b to select PIRQD.
7:0	<b>INTLINE.</b> Read-write. This data is not used by hardware. It is used to communicate across software the interrupt line that the interrupt pin is connected to.

**SMBus Controller Revision Id and Class Code Alias****DevB:2x40**

Default: 0C05\_0001h

Attribute: See below.

Bits	Description
31:24	<b>BASECLASS.</b> Read-write. These bits default to 0Ch indicating a serial bus controller.
23:16	<b>SUBCLASS.</b> Read-write. These bits default to 05h indicating a SMBus controller.
15:8	<b>PROGIF.</b> Read-write.
7:0	<b>REVID. SMBus Controller silicon revision.</b> Read-only. 01h = revision A0.

**SMBus Controller Subsystem and Subsystem Vendor Id Alias****DevB:2x44**

Default: 0000\_0000h

Attribute: Read-write

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b>
15:0	<b>SUBVENID. Subsystem Vendor ID.</b>

**SMBus Controller Miscellaneous Control****DevB:2x48**

**Note:** This register resides on the VDD\_COREX power plane and is cleared to its default value only when coming out of mechanical off power state (MOFF).

Default: 0000\_0006h

Attribute: See below.

Bits	Description
31:3	Reserved.
2	<b>SET_SCISTS_EN.</b> When set to 1b setting of SC04[SCI_EVT] or SC04[OBF] or clearing of SC04[IBF] sets PM20[SMBC_STS].
1	<b>SET_INTSTS_EN.</b> When set to 1b setting of SC04[SCI_EVT] or SC04[OBF] or clearing of SC04[IBF] sets SC08[INT_STS].
0	<b>SU. SMBus clock speed-up.</b> Read-write. When set to 1b the nominal SMBus clock rate is increased by 16x.

**4.6.2 SMBus Controller Interface Registers**

The command interface registers are located in I/O space and provide an ACPI 2.0 chapter 13 compliant interface for communication with the SMBus controller. Refer to the ACPI 2.0 chapter 13 for a detailed description of the interface operation. The base address register for these registers is DevB:2x10. See Section 4.1.2 on page 136 for a description of the register naming convention.

**Data Register****SC00**

Default: 0000h

Attribute: Read-write

Bits	Description
7:0	<b>DATA. Data port.</b> This register acts as a port for transferring data between the host and the SMBus controller. Writes to this port by the host are stored in the internal input data register. Reads from this register return data from the internal output data register.

**Status Register****SC04**

This register is aliased to the same address as Command (SC04) but provides only read access.

Default: 00h

Attribute: Read-only.

Bits	Description
7:6	Reserved.
5	<b>SCI_EVT. SCI event pending.</b> 1: SCI event is pending (requesting SCI query). 0: No SCI events are pending. This bit is set by hardware when one of the events enabled by SC08[SMBALERT_EN] or indicated by SMB08[DONE, ALRM, STATUS] occurs. When enabled by DevB:2x48[SET_SCISTS_EN] setting this bit also sets PM20[SMBC_STS]. When enabled by DevB:2x48[SET_INTSTS_EN] setting this bit also sets SC08[INT_STS]. This bit is cleared by hardware when all notification headers were cleared by query commands from the host.
4	<b>BURST. Burst mode.</b> 1: Controller is in burst mode for polled command processing. 0: Controller is in normal mode for interrupt-driven command processing. This bit indicates that the controller has received the burst enable command from the host, has halted normal processing, and is waiting for a series of commands to be sent from the host. This bit is set by the controller upon receiving the Burst Acknowledge command (0x90) and cleared upon receiving the Burst Disable command (0x83).

Bits	Description
3	<p><b>CMD. Command byte.</b></p> <p>1: Byte in the internal input data register is a command byte.</p> <p>0: Byte in the internal input data register is a data byte.</p> <p>This bit is set by hardware upon write access to SC04 by the host. This bit is cleared by hardware upon write access to SC00 by the host.</p>
2	Reserved.
1	<p><b>IBF. Input buffer full.</b></p> <p>1: Input buffer is full.</p> <p>0: Input buffer is empty.</p> <p>This bit is set by hardware upon write access to either SC00 or SC04 by the host. This bit is cleared by hardware when the controller reads its internal input data register.</p>
0	<p><b>OBF. Output buffer full.</b></p> <p>1: Output buffer is full.</p> <p>0: Output buffer is empty.</p> <p>This bit is set by hardware when the controller places data into its internal output data register. This bit is cleared by hardware upon read access to SC00 by the host.</p>

**Command Register****SC04**

This register is aliased to the same address as Status (SC04) but provides only write access.

Default: 00h

Attribute: Write-only.

Bits	Description
7:0	<p><b>CMD. Command port. Data port.</b> This register acts as a port for transferring commands from the host to the SMBus controller. Writes to this port by the host are stored in the internal input data register.</p>

**Interrupt Control Register****SC08**

Default: 00h

Attribute: Read-write.

Bits	Description
7:4	Reserved.

3	<b>SMBALERT_EN. SMBALERT enable.</b> 1b enables an internal notification header of 80h to be set and SC04[SCI_EVT] to be set when an SMBALERT event occurs.  <i>Note:</i> This bit resides on the VDD_COREX power plane and is cleared to its default value only when coming out of the mechanical off power state (MOFF).
2	Reserved.
1	<b>INT_EN. Interrupt enable.</b> 1: PCI interrupt is driven when INT_STS is set. 0: PCI interrupt disabled.
0	<b>INT_STS. Interrupt status.</b> This bit is set by hardware upon setting of SC04[SCI_EVT] or SC04[OBF] or clearing of SC04[IBF] when enabled by DevB:2x48[SET_INTSTS_EN]. This bit is cleared by writing 1b to this position.

### 4.6.3 Host Controller Interface Registers

The host controller interface registers are located in the internal address space of the SMBus controller and provide an ACPI 2.0 chapter 13.9 compliant interface for communication with the host controller. Refer to the ACPI 2.0 chapter 13.9 for a detailed description of the interface operation. Access for host operations is provided through the SMBus controller interface register SC00 and SC04.

These registers reside on the VDD\_COREX power plane and are only cleared to their default values when coming out of the mechanical off power state (MOFF).

#### Protocol Register

**SMB00**

This register determines the type of SMBus transaction generated on the SMBus. Writing to this register initiates the transaction on the SMBus.

Default: 00h

Attribute: Read-write.

Bits	Description
7	<b>PEC. Packet error checking enable.</b> 1: PEC format used for the specified protocol. 0: standard (non-PEC) format used for the specified protocol.
6:0	<b>PRTCL. SMBus protocol.</b> This register determines the type of SMBus transaction generated on the SMBus. In addition to the commands described in the ACPI 2.0 specification the host controller also supports the following commands: 0x4A Block Write I2C 0x4B Block Read I2C



**Status Register****SMB01**

This register is cleared by hardware (except for the ALRM bit) whenever a new command is issued using a write access to SMB00. Any bit of this register being set causes an internal notification header of 20h to be set and SC04[SCI\_EVT] to be set.

Default: 00h

Attribute: See below.

Bits	Description
7	<b>DONE. Command completion.</b> Read-only. 1b indicates the last command has completed without error.
6	<b>ALRM.</b> Read/clear by write. 1b indicates an SMBus alarm message has been received. This bit is cleared by writing 00h to SMB01.
5	Reserved.
4:0	<b>STATUS.</b> Read-only. SMBus status. Indicates SMBus communication status as listed in the table below.

**Table 57. SMBus Status**

Status Code	Name	Description
00h	SMBus OK	Indicates the transaction has been successfully completed
10h	SMBus Device Address Not Acknowledged	Indicates the transaction failed because the slave device address was not acknowledged.
11h	SMBus Device Error Detected	Indicates the transaction failed because the slave device not acknowledged another byte
18h	SMBus Timeout	Indicates the transaction failed because the SMBus host detected a timeout on the bus
19h	SMBus Host Unsupported Protocol	Indicates the transaction failed because the SMBus host does not support the requested protocol
1Ah	SMBus Busy or Arbitration lost	Indicates the transaction failed because the SMBus host reports that: - Arbitration lost or - SMBus was busy for a time longer than 40,96 ms
1Fh	SMBus PEC (CRC-8) Error	Indicates that a Packet Error Checking (PEC) error occurred during the last transaction.

**Address Register****SMB02**

Default: 00h

Attribute: Read-write.

Bits	Description
7:1	<b>ADDR. SMBus address.</b> This register contains the 7 bit address for SMBus transactions.
0	Reserved.

**Command Register****SMB03**

Default: 00h

Attribute: Read-write.

Bits	Description
7:0	<b>CMD. SMBus command.</b> This register contains the command byte that is sent to the SMBus device and is used for the following protocols—send byte, write byte, write word, read byte, read word, process call, block read, block write, block read I2C and block write I2C. It is not used for the quick commands or the receive byte protocol, and as such, its value is a don't care for those commands

**Data Registers****SMB04-SMB23**

This bank of registers contains the bytes to be sent or received in any SMBus transaction. The registers are defined on a per-protocol basis and, as such, provide efficient use of register space.

Default: 00h

Attribute: Read-write.

Bits	Description
7:0	<b>DATA. SMBus data.</b>

**Block Count Register****SMB24**

Default: 00h

Attribute: Read-write.

Bits	Description
7:5	Reserved.
4:0	<b>BCNT. Block count.</b> This register contains the number of bytes of data present in the SMB[23:04] preceding any write block and following any read block transaction. For the read block I2C transaction the BCNT register contains the number of bytes to be received. The data size is defined on a per protocol basis. The following encoding is used: 00001:        1 byte 00010:        2 bytes ... 00000:        32 bytes

**Alarm Address Register****SMB25**

Default: 00h

Attribute: Read-write.

Bits	Description
7:1	<b>ADDR. Alarm address.</b> This register contains the address of an alarm message indicating the slave address of the device on the SMBus that initiated the alarm message.
0	Reserved.

**Alarm Data Register****SMB26-SMB27**

Default: 00h

Attribute: Read-write.

Bits	Description
7:0	<b>DATA. Alarm data.</b> These registers contain the two data bytes of an alarm message indicating the specific reason for the alarm message. Once an alarm message has been received, the host controller will not receive additional alarm messages until the SMB01[ALRM] is cleared.

## 4.7 System Management Registers

### 4.7.1 System Management Configuration Registers (DevB:3xXX)

These registers are located in PCI configuration space on the primary PCI interface, in the second device (device B), function 3. See Section 4.1.2 on page 136 for a description of the register naming convention.

**System Management Vendor And Device ID Register****DevB:3x00**

Default: 746B 1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>System management device ID.</b>
15:0	<b>Vendor ID.</b>

**System Management Status And Command Register****DevB:3x04**

Default: 0280 0000h.

Attribute: Read-only.

Bits	Description
31:0	These bits are fixed at their default values.

**System Management Revision And Class Code Register****DevB:3x08**

Default: 0000 00XXh See below.

Attribute: Read-only.

Bits	Description
31:8	<b>CLASSCODE.</b> This field is write accessible through DevB:3x60.
7:0	<b>REVISION. System management silicon revision.</b> The value of this register is revision-dependent.

**System Management BIST-Header-Latency-Cache Register****DevB:3x0C**

Default: 0000 1600h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. These bits fixed at their default values.
23:16	<b>HEADER.</b> Read-only. These bits fixed at their default values.
15:8	<b>LATENCY.</b> Read-write. This field controls no hardware.
7:0	<b>CACHE.</b> Read-only. These bits fixed at their default values.

**System Management Subsystem ID and Subsystem Vendor ID Register****DevB:3x2C**

Default: 0000 0000h.

Attribute: Read-only.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> This field is write accessible through DevB:3x7C.
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> This field is write accessible through DevB:3x7C.

**General Configuration 1 Register****DevB:3x40**

Default: 00h.

Attribute: Read-write.

Bits	Description										
7	<b>RNGEN. Random number generated enable.</b> Read; write to 1 only. 1=The RNG (accessible through PMF0 and PMF4) is enabled to generate random numbers. 0=The RNG is disabled. This bit can only be written to a 1; writing a 0 has no effect.										
6	<b>STOPTMR. Stop ACPI timer from counting.</b> 1=ACPI timer, PM08, stops counting (frozen in its current state). 0=ACPI timer enabled to count.										
5:4	<b>THMINEN. Throttling minimum enable time.</b> These bits specify the minimum time in which STPCLK_L is held in the deasserted state during throttling (normal or thermal), regardless of the state of the throttling duty cycle registers, DevB:3x4D and PM10. These bits do not affect the throttling cycle period; if the throttling duty cycle specifies a STPCLK_L deassertion time that is less than specified by this register, then the STPCLK_L assertion time is reduced. This has no effect on throttling in which the period is 244 microseconds.  <table border="0"> <tr> <td>THMINEN</td> <td>Time</td> </tr> <tr> <td>00b</td> <td>None.</td> </tr> <tr> <td>01b</td> <td>10 microseconds.</td> </tr> <tr> <td>10b</td> <td>20 microseconds.</td> </tr> <tr> <td>11b</td> <td>Reserved.</td> </tr> </table>	THMINEN	Time	00b	None.	01b	10 microseconds.	10b	20 microseconds.	11b	Reserved.
THMINEN	Time										
00b	None.										
01b	10 microseconds.										
10b	20 microseconds.										
11b	Reserved.										

Bits	Description
3	<b>NTPER. Normal throttling period.</b> 1=Normal throttling cycle period specified to be 244 microseconds (from an asserting edge of STPCLK_L to the next asserting edge). Minimum LDTSTOP_L assertion during throttling if DevB:3x70[NTLS]=1 is 16us. 0=Normal throttling period specified to be 30 microseconds. Minimum LDTSTOP_L assertion during throttling if DevB:3x70[NTLS]=1 is 2us.
2	<b>TTPER. Thermal throttling period.</b> 1=Thermal throttling cycle period specified to be 244 microseconds (from an asserting edge of STPCLK_L to the next asserting edge). Minimum LDTSTOP_L assertion during throttling if DevB:3x70[TTLS]=1 is 16us. If 0=Thermal throttling period specified to be 30 microseconds. Minimum LDTSTOP_L assertion during throttling if DevB:3x70[TTLS]=1 is 2us.
1	Reserved.
0	<b>TH2SD. Throttling 2 second delay.</b> 1=There is a 2.0 to 2.5 second delay after THERM_L is asserted before thermal throttling is initiated, as specified by DevB:3x4D. 0=Initiate throttling immediately after THERM_L is asserted.

**General Configuration 2 Register****DevB:3x41**

Default: 41h.

Attribute: See below.

Bits	Description
7	<b>PMIOEN. System management I/O space enable.</b> Read-write. 1=PMxx, the I/O space specified by DevB:3x58, is enabled.
6	<b>TMRRST. ACPI timer reset.</b> Read-write. 1=The ACPI timer, PM08, is asynchronously cleared at all times. 0=The timer is allowed to count.
5	<b>PCF9EN. Port CF9 enable.</b> Read-write. 1=Access to PORTCF9 is enabled.
4	<b>PBIN. Power button in.</b> Read-only. This bit reflect the current state of the PWRBTN_L pin (before the debounce circuit). 0=PWRBTN_L is currently asserted.
3	<b>TMR32. Timer size selection.</b> Read-write. 0=The ACPI timer, PM08, is 24 bits. 1=The ACPI timer is 32 bits.
2:1	Reserved.
0	<b>Must be High.</b> Read-write. This bit is required to be High at all times; setting it Low results in undefined behavior.

**SCI Interrupt Configuration Register****DevB:3x42**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:4	Reserved.
3:0	<b>SCISEL. SCI interrupt selection.</b> This field specifies the IRQ number routed to the interrupt controllers used for ACPI-defined SCI interrupts. A value of 0h disables SCI interrupts. Values of 2h, 8h, and Dh are reserved. All other values are valid. See Section 3.4.2.1 on page 41 for details about SCI interrupt routing.

**Previous Power State Register****DevB:3x43**

Default: see field specifications below.

Attribute: See below.

Bits	Description																				
7	<b>VDDA_STS. AL reset status.</b> Read; set by hardware; write 1 to clear. 1=AL became invalid. This bit resides on the VDD_COREAL plane.																				
6	<b>G3TOS5. Mechanical off (G3) to soft off (S5).</b> Read; write. 0=When power is applied to the AUX planes, the system automatically transitions into the FON state. 1=When power is applied to the AUX planes, the system enters the SOFF state. This bit resides on the VDD_COREAL plane. When VDD_COREAL powers up, this bit defaults to 0; when there is a RESET_L generated by a write to PORTCF9[RSTCMD, SYSRST] == 11b, then this bit is cleared.																				
5:3	<p><b>PWRFL_STS. Power failure status.</b> Read; updated by hardware; write 1 to LSB to reset to 4h. If power fails (system enters MOFF or PWROK goes from 1 to 0 while PWRON_L is asserted), then this register captures and retains the state the system was in when the failure occurred. This field resides on the VDD_COREAL plane. This field defaults to 4h when VDD_COREAL powers up. Writing a one to DevB:3x43[3] sets this field to 4h; writing a zero to Dev0:3x43[3] or writing any value to any other bits in this field has no effect. The possible states are:</p> <table border="0"> <tr> <td>PWRFL_STS</td> <td>Power state</td> <td>PWRFL_STS</td> <td>Power state</td> </tr> <tr> <td>0h</td> <td>FON full on</td> <td>4h</td> <td>No AC power failure</td> </tr> <tr> <td>1h</td> <td>POS power on suspend</td> <td>5h</td> <td>STR suspend to RAM</td> </tr> <tr> <td>2h</td> <td>C2</td> <td>6h</td> <td>STD suspend to disk</td> </tr> <tr> <td>3h</td> <td>C3</td> <td>7h</td> <td>SOFF soft off</td> </tr> </table> <p>This bit needs to be reset in order to allow the next power failure status to be captured.</p>	PWRFL_STS	Power state	PWRFL_STS	Power state	0h	FON full on	4h	No AC power failure	1h	POS power on suspend	5h	STR suspend to RAM	2h	C2	6h	STD suspend to disk	3h	C3	7h	SOFF soft off
PWRFL_STS	Power state	PWRFL_STS	Power state																		
0h	FON full on	4h	No AC power failure																		
1h	POS power on suspend	5h	STR suspend to RAM																		
2h	C2	6h	STD suspend to disk																		
3h	C3	7h	SOFF soft off																		
2:0	<p><b>PPSTATE. Previous power state.</b> Read-only. This field holds the most previous power state from which the system came into the FON state. This field resides on the VDD_COREX plane. Here are the possible states:</p> <table border="0"> <tr> <td>PPSTATE</td> <td>Power state</td> <td>PPSTATE</td> <td>Power state</td> </tr> <tr> <td>0h</td> <td>Reserved</td> <td>4h</td> <td>MOFF mechanical off</td> </tr> <tr> <td>1h</td> <td>POS power on suspend</td> <td>5h</td> <td>STR suspend to RAM</td> </tr> <tr> <td>2h</td> <td>C2</td> <td>6h</td> <td>STD suspend to disk</td> </tr> <tr> <td>3h</td> <td>C3</td> <td>7h</td> <td>SOFF soft off</td> </tr> </table>	PPSTATE	Power state	PPSTATE	Power state	0h	Reserved	4h	MOFF mechanical off	1h	POS power on suspend	5h	STR suspend to RAM	2h	C2	6h	STD suspend to disk	3h	C3	7h	SOFF soft off
PPSTATE	Power state	PPSTATE	Power state																		
0h	Reserved	4h	MOFF mechanical off																		
1h	POS power on suspend	5h	STR suspend to RAM																		
2h	C2	6h	STD suspend to disk																		
3h	C3	7h	SOFF soft off																		

**PNP IRQ Select Register****DevB:3x44**

Bits[11:0] assign PNPIRQ[2:0] pins to IRQs that are routed to interrupt controllers; see Section 3.4.2.1 on page 41 for more details.

Default: 0000h.

Attribute: Read-write.

Bits	Description										
15	<b>TCO_INT_EN. TCO interrupt enable.</b> 1=Enable TCO IRQ selected by DevB:3x44[TCO_INT_SEL] (if PM22[TCOSCI_EN] = 0).										
14:12	<b>TCO_INT_SEL. TCO interrupt select.</b> Specifies the IRQ line asserted by either PM46[INTRDR_STS] (if PM44[INTRDR_SEL] selects IRQ) or PM44[TCO_INT_STS]. Note: if PM22[TCOSCI_EN] is set, then this field is ignored. Note: if one of IRQ[11:9] is selected, then the interrupt controller must be programmed as level sensitive for this IRQ.  <table border="0"> <tr> <td>TCO_INT_SEL Interrupt</td> <td>TCO_INT_SEL Interrupt</td> </tr> <tr> <td>0h                   IRQ9</td> <td>4h                   APIC IRQ20</td> </tr> <tr> <td>1h                   IRQ10</td> <td>5h                   APIC IRQ21</td> </tr> <tr> <td>2h                   IRQ11</td> <td>6h                   APIC IRQ22</td> </tr> <tr> <td>3h                   Reserved</td> <td>7h                   APIC IRQ23</td> </tr> </table>	TCO_INT_SEL Interrupt	TCO_INT_SEL Interrupt	0h                   IRQ9	4h                   APIC IRQ20	1h                   IRQ10	5h                   APIC IRQ21	2h                   IRQ11	6h                   APIC IRQ22	3h                   Reserved	7h                   APIC IRQ23
TCO_INT_SEL Interrupt	TCO_INT_SEL Interrupt										
0h                   IRQ9	4h                   APIC IRQ20										
1h                   IRQ10	5h                   APIC IRQ21										
2h                   IRQ11	6h                   APIC IRQ22										
3h                   Reserved	7h                   APIC IRQ23										
11:8	<b>IRQ2SEL. PNPIRQ2 interrupt select.</b> This selects the IRQ number for PNPIRQ2. IRQ0, IRQ2, IRQ8, and IRQ13 are reserved. If PMD5 does not select the PNPIRQ2 function then this field has no effect. See Section 3.4.2.1 on page 41 for more details.										
7:4	<b>IRQ1SEL. PNPIRQ1 interrupt select.</b> This selects the IRQ number for PNPIRQ1. IRQ0, IRQ2, IRQ8, and IRQ13 are reserved. If PMD4 does not select the PNPIRQ1 function then this field has no effect. See Section 3.4.2.1 on page 41 for more details.										
3:0	<b>IRQ0SEL. PNPIRQ0 interrupt select.</b> This selects the IRQ number for PNPIRQ0. IRQ0, IRQ2, IRQ8, and IRQ13 are reserved. If PMD3 does not select the PNPIRQ0 function then this field has no effect. See Section 3.4.2.1 on page 41 for more details.										

**Pins Latched On The Trailing Edge Of Reset Register****DevB:3x48**

The default for all of these bits is specified by pull up or pull down resistors on pins during the trailing edge of the specified reset (PWROK). To latch a Low on these pins, a 10K to 100K ohm resistor to ground is placed on the signal. To latch a High on these pins, a 10K to 100K ohm resistor to the pin's power plane is placed on the signal.

Default: Each of these bits is latched on the trailing edge of reset.

Attribute: See below.

Bits	Description
15	<b>TBD15.</b> Read-only. The state of this bit is latched off of AD[15] at the trailing edge of PWROK reset. This bit controls no internal hardware.
14	<b>TBD14.</b> Read-only. The state of this bit is latched off of AD[14] at the trailing edge of PWROK reset. This bit controls no internal hardware.
13	<b>TBD13.</b> Read-only. The state of this bit is latched off of AD[13] at the trailing edge of PWROK reset. This bit controls no internal hardware.
12	<b>SPKRL. SPKR latch.</b> Read-write. The state of this bit is latched off of SPKR at the trailing edge of PWROK reset. This controls no internal logic.

Bits	Description (Continued)
11	<b>CMPOVR. HyperTransport™ technology automatic compensation override.</b> Read-write. 1=The compensation values specified by DevA:0xE0[RX_ORC, TXN_ORC, and TXP_ORC] are used by the HyperTransport PHY. 0=The automatic PHY compensation circuit values are used by the HyperTransport PHY. The state of this bit is latched off of AD[11] at the trailing edge of PWROK reset.
10	<b>NO_REBOOT. Do not reboot the system when a double TCO timer time out occurs.</b> Read-write. 0=Reboot system as specified by PORTCF9[FULLRST] when PM46[2NDTO_STS] is set. 1=Do not reboot the system. The state of this bit is latched off of AD[10] at the trailing edge of PWROK reset.
9	<b>Low strap required.</b> Read-write. The default state of this bit is latched off of AD[9] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
8	<b>Low strap required.</b> Read-only. The default state of this bit is latched off of AD[8] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
7	<b>Low strap required.</b> Read-write. The default state of this bit is latched off of AD[7] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
6	<b>TBD6.</b> Read-only. The state of this bit is latched off of AD[6] at the trailing edge of PWROK reset. This bit controls no internal hardware.
5	<b>High strap required.</b> Read-write. The default state of this bit is latched off of AD[5] at the trailing edge of PWROK reset. This bit is required to be High at all times; if it is Low then undefined behavior results.
4	<b>Low strap required.</b> Read-write. The default state of this bit is latched off of AD[4] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
3	<b>TBD3.</b> Read-only. The state of this bit is latched off of AD[3] at the trailing edge of PWROK reset. This bit controls no internal hardware.
2	<b>Low strap required.</b> Read-only. The default state of this bit is latched off of AD[2] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
1	<b>Low strap required.</b> Read-only. The default state of this bit is latched off of AD[1] at the trailing edge of PWROK reset. This bit is required to be Low at all times; if it is High then undefined behavior results.
0	<b>PCIBIOS.</b> Read-write. This specifies routing of accesses to BIOS address space (specified by DevB:0x43). 1b = BIOS address space is located on the PCI bus. 0b = BIOS address space is located on the LPC bus. The state of this bit is latched off of AD[0] at the trailing edge of reset.



**Serial IRQ Control Register****DevB:3x4A**

Default: 10h.

Attribute: Read-write.

Bits	Description
7	Reserved.
6	<b>CONTMD. Continuous mode selected versus quiet mode.</b> 1=The serial IRQ logic is in continuous mode. 0=The serial IRQ logic is in quiet mode. In continuous mode, the start frame is initiated by the IC immediately following each stop frame. In quiet mode, start frames are initiated by external slave devices.
5:2	<b>FRAMES. Number of IRQ frames for a serial IRQ cycle.</b> This specifies the number of 3-clock IRQ frames that the IC generates, during a serial IRQ cycle before, issuing the stop frame. Per the serial IRQ specification, the number of frames is 17 plus the value of this field.
1:0	<b>STARTCLKS. Number of clocks in start pulse.</b> This specifies the size of the start pulse over SERIRQ during the start frame of a serial IRQ cycle (including the slave cycle if in quiet mode) in PCLK cycles. 00b = 4 cycles. 01b = 6 cycles; 10b = 8 cycles and 11b = reserved.

**PRDY Timer Control Register****DevB:3x4C**

Each of these bits controls the ability of the PRDY input signal to disable internal counters. When PRDY becomes active then the counters that correspond to the active bits in this register stop counting until PRDY becomes inactive.

Default: 00h.

Attribute: Read-write.

Bits	Description
7:3	Reserved.
2	<b>SMT_DIS. System management timer disable.</b> 1=The all the system management timers specified by the PMxx space are disabled from counting while PRDY is active. These include the ACPI timer at PM08, re-trigger timers at PM[8C:50], the system inactivity timer at PM98, and the general purpose timer at PM94.
1	<b>RTC_DIS. Real-time clock disable.</b> 1=The real-time clock's counters that are clocked off of the 32 kHz clock are disabled from counting while PRDY is active.
0	<b>PIT_DIS. Programmable interval timer disable.</b> 1=The clock to the three timers of the internal PIT are disabled when PRDY is active.

**Thermal Throttling Register****DevB:3x4D**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:6	Reserved.

5	<b>TTHLOCK. Thermal throttling lock.</b> Write 1 only. 1=Writes to TTH_EN and TTH_RATIO are disabled. Once set, this bit cannot be cleared by software. It is only cleared by a RESET_L. TTH_EN and TTH_RATIO may change during the write command that sets TTHLOCK High.			
4	<b>TTH_EN. Thermal throttling enable.</b> 1=Thermal throttling is enabled when THERM_L is asserted. 0=Thermal throttling cannot be enabled.			
3:1	<b>TTH_RATIO. Thermal throttling duty cycle.</b> These specify the duty cycle of STPCLK_L when the system is in thermal throttling mode (initiated by the THERM_L pin when enabled by TTH_EN). The field is decoded as follows:			
	TTH_RATIO	Description	TTH_RATIO	Description
	0h	Reserved	4h	50.0% in Stop Grant state
	1h	87.5% in Stop Grant state	5h	37.5% in Stop Grant state
	2h	75.0% in Stop Grant state	6h	25.0% in Stop Grant state
	3h	62.5% in Stop Grant state	7h	12.5% in Stop Grant state
0	Reserved.			

**Read-Write Register****DevB:3x4E**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:4	Reserved.
3:0	<b>RW.</b> These bits control no hardware.

**C2/C3 State Control Register****DevB:3x4F**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:6	Reserved.
5	<b>ASTP_C3EN. Enable AGPSTOP_L assertion during C3.</b> 1=Enables the assertion of AGPSTOP_L during C3.
4	Reserved.
3	<b>CSLP_C3EN. Enable CPUSLEEP_L assertion during C3.</b> 1=Enables the control of the processor SLEEP_L pin during the C3 state transition. 0=CPUSLEEP_L is always High. This bit has no effect if the PMC6 does not select the CPUSLEEP_L function.
2	Reserved.
1	<b>C3EN. Enable C3 command.</b> 1=Enables the IC to place the processor into the Stop Grant state by asserting STPCLK_L when PM15 is read. Note, this bit must be High for CSLP_C3EN bits to function during the transition to C3.
0	<b>C2EN. Enable C2 command.</b> 1=Enables the IC to place the processor into the Stop Grant state by asserting STPCLK_L when PM14 is read. 0=Disable.

**Power On Suspend Control Register****DevB:3x50**

This register specifies the action taken by the IC when the sleep command is sent to PM04 with SLP\_TYP indicating power on suspend. Note: if POSEN is Low, then the rest of the bits in this register are ignored.

Default: 8000h.

Attribute: Read-write.

Bits	Description
15	<b>PITRSM_L. Enable the PIT to generate unmasked interrupts during POS.</b> 1=PIT does not generate an interrupt to the PIC or IOAPIC while in POS (starting from the time that the command to enter POS is sent to PM04); this may be used to prevent timer-tick interrupts from resuming the system while in POS. 0=PIT generates an interrupt to the PIC or IOAPIC while in POS.
14	<b>MSRSM_L. Enable the mouse interrupt to generate unmasked interrupts during POS.</b> 1=Disable interrupt from both the IRQ12 pin and IRQ 12 of the serial IRQ logic to the PIC or IOAPIC while in POS (starting from the time that the command to enter POS is sent to PM04); this may be used to prevent mouse interrupts from resuming the system while in POS. 0=Enable interrupt to the PIC or IOAPIC while in POS.
13:9	Reserved.
8	<b>SUSP. Enable SUSPEND_L assertion during POS.</b> 1=Enables the control of the SUSPEND_L pin during POS.
7	Reserved.
6	<b>CSLP. Enable CPUSLEEP_L assertion during POS.</b> 1=Enables assertion of the CPUSLEEP_L pin during POS. 0=Disable. This bit has no effect if the PMC6 does not select the CPUSLEEP_L function.
5	<b>DCSTP. Enable DCSTOP_L assertion during POS.</b> 1=Enables assertion of the DCSTOP_L pin during POS. This bit has no effect if the PMC8 does not select the DCSTOP_L function.
4	<b>ASTP. Enable AGPSTOP_L assertion during POS.</b> 1=Enables assertion of the AGPSTOP_L pin during POS. This bit has no effect if the PMC1 does not select the AGPSTOP_L function.
3	<b>PSTP. Enable PCISTOP_L assertion during POS.</b> 1=Enables the control of the PCISTOP_L pin during POS.
2	<b>CSTP. Enable CPUSTOP_L assertion during POS.</b> 1=Enables control of the CPUSTOP_L pin during POS. This bit has no effect if the PMC7 does not select the CPUSTOP_L function.
1	<b>RW.</b> Read-write. This bit is read-write accessible through software; it controls no hardware.
0	<b>POSEN. Enable Stop Grant during POS.</b> 1=Enables the IC to place the processor into the Stop Grant state by asserting STPCLK_L when a POS commands is sent to PM04. This bit required to be set High for any other bits in this register to be enabled.

**Misc ACPI Timing Register****DevB:3x52**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:2	Reserved.
1:0	<b>C3_ASTP_DT. C3 AGPSTOP_L Deassertion delay Time.</b> Specifies the delay between LDTSTOP_L deassertion and AGPSTOP_L deassertion when exiting C3 as follows: 00 = 1us (minimum, allowable positive tolerance 10%) 01 = 32us (+/- 10%) 10 = 64us (+/- 10%) 11 = 128us (+/- 10%)

**PLL Lock Timer Register****DevB:3x54**

Default: 0Fh.

Attribute: Read-write.

Bits	Description
7:4	<b>PLLCNT1. PLL lock timer count 1.</b> Specifies the PLL recovery time as follows: PLL synchronization time = (PLLCNT + 1) * 61.035 microseconds.
3:0	<b>PLLCNT. PLL lock timer count.</b> Specifies the PLL recovery time as follows: PLL synchronization time = (PLLCNT + 1) * 61.035 microseconds. See Section 3.7.1.6.5 on page 63 for details about this timer.

**PCI IRQ Routing Register****DevB:3x56**

Default: 0000h.

Attribute: Read-write.

Bits	Description																																				
15:12	<b>PIRQD_L select.</b> See bits[3:0].																																				
11:8	<b>PIRQC_L select.</b> See bits[3:0].																																				
7:4	<b>PIRQB_L select.</b> See bits[3:0].																																				
3:0	<b>PIRQA_L select.</b> These map the PCI IRQ pins to the internal interrupt controller (PIC and APIC). These are decoded as follows: <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">PIRQ[A,B,C,D]_L Selects</th> <th style="text-align: left;">Selected IRQ</th> <th style="text-align: left;">PIRQ[A,B,C,D]_L Selects</th> <th style="text-align: left;">Selected IRQ</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>None</td> <td>8h</td> <td>Reserved</td> </tr> <tr> <td>1h</td> <td>IRQ1</td> <td>9h</td> <td>IRQ9</td> </tr> <tr> <td>2h</td> <td>Reserved</td> <td>Ah</td> <td>IRQ10</td> </tr> <tr> <td>3h</td> <td>IRQ3</td> <td>Bh</td> <td>IRQ11</td> </tr> <tr> <td>4h</td> <td>IRQ4</td> <td>Ch</td> <td>IRQ12</td> </tr> <tr> <td>5h</td> <td>IRQ5</td> <td>Dh</td> <td>Reserved</td> </tr> <tr> <td>6h</td> <td>IRQ6</td> <td>Eh</td> <td>IRQ14</td> </tr> <tr> <td>7h</td> <td>IRQ7</td> <td>Fh</td> <td>IRQ15</td> </tr> </tbody> </table>	PIRQ[A,B,C,D]_L Selects	Selected IRQ	PIRQ[A,B,C,D]_L Selects	Selected IRQ	0h	None	8h	Reserved	1h	IRQ1	9h	IRQ9	2h	Reserved	Ah	IRQ10	3h	IRQ3	Bh	IRQ11	4h	IRQ4	Ch	IRQ12	5h	IRQ5	Dh	Reserved	6h	IRQ6	Eh	IRQ14	7h	IRQ7	Fh	IRQ15
PIRQ[A,B,C,D]_L Selects	Selected IRQ	PIRQ[A,B,C,D]_L Selects	Selected IRQ																																		
0h	None	8h	Reserved																																		
1h	IRQ1	9h	IRQ9																																		
2h	Reserved	Ah	IRQ10																																		
3h	IRQ3	Bh	IRQ11																																		
4h	IRQ4	Ch	IRQ12																																		
5h	IRQ5	Dh	Reserved																																		
6h	IRQ6	Eh	IRQ14																																		
7h	IRQ7	Fh	IRQ15																																		

**PMxx System Management I/O Space Pointer****DevB:3x58**

Default: 0000 DD01h.

Attribute: See below.

Bits	Description
31:16	Reserved.
15:8	<b>PMBASE.</b> Read-write. Specifies PCI address bits[15:8] of the 256-byte block of I/O-mapped registers used for system management (address space PMxx). Access to this address space is enabled by DevB:3x41[PMIOEN].
7:0	<b>PMBLSB.</b> Read-only. These fixed bits read 01h.

**System Management Class Code Write Register****DevB:3x60**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:8	<b>CCWRITE.</b> The value placed in this register is visible in DevB:3x08.
7:0	Reserved.

**Clock Control Register****DevB:3x64**

This register resides on the VDD\_COREX power plane.

Default: 0000 0013h.

Attribute: Read-write.

Bits	Description
7:6	Reserved.
5	<b>SMB1_DB. SMBUSC0 debounce. (Revision C0 and later)</b> .1=Enable the SMBus 1.0 controller clock input debounce logic. When the SMBUSC0 signal is asserted High or Low for less than 60ns, the debounce logic does not propagate a change of the signal value to the internal logic. The signal must be asserted for at least 80ns to be safely detected by the internal logic. SMBC_EN must be set for the debounce logic to function. 0=Disable SMBUSC0 debounce logic.  Note: The functionality of the SMBus 1.0 controller apart from the debounce logic is not affected by the value of SMBC_EN.

Bits	Description (Continued)																									
4	<b>SMBC_EN. SMBus controller clock enable.</b> 1=Enable the clock to the SMBus controller. 0=disable the clock to the SMBus controller. Whenever the SMBus controller clock is disabled, then reset is asserted to that device.																									
3:2	<b>Reserved.</b>																									
1	<b>L7_S3EN. LAN Ethernet controller clock enable.</b> See bit [0].																									
0	<b>L7_EN. LAN Ethernet controller clock enable.</b> Bits [1] and [0] control the enable to the clock that is used by the LAN Ethernet controller as follows:  <table border="1"> <thead> <tr> <th>L7_EN</th> <th>L7_S3EN</th> <th>FON, POS</th> <th>STR, STD, SOFF</th> <th>Comment</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>No clock</td> <td>No clock</td> <td>Power savings mode.</td> </tr> <tr> <td>0</td> <td>1</td> <td>No clock</td> <td>No clock</td> <td>Power savings mode.</td> </tr> <tr> <td>1</td> <td>0</td> <td>Clock enabled</td> <td>No clock</td> <td>LAN Ethernet controller not enabled for wake on LAN. The corresponding MII interface is not driven during STR, STD, and SOFF.</td> </tr> <tr> <td>1</td> <td>1</td> <td>Clock enabled</td> <td>Clock enabled</td> <td>LAN Ethernet controller enabled for wake on LAN.</td> </tr> </tbody> </table> Whenever the LAN Ethernet controller is in a no-clock state, then reset is asserted to that device.	L7_EN	L7_S3EN	FON, POS	STR, STD, SOFF	Comment	0	0	No clock	No clock	Power savings mode.	0	1	No clock	No clock	Power savings mode.	1	0	Clock enabled	No clock	LAN Ethernet controller not enabled for wake on LAN. The corresponding MII interface is not driven during STR, STD, and SOFF.	1	1	Clock enabled	Clock enabled	LAN Ethernet controller enabled for wake on LAN.
L7_EN	L7_S3EN	FON, POS	STR, STD, SOFF	Comment																						
0	0	No clock	No clock	Power savings mode.																						
0	1	No clock	No clock	Power savings mode.																						
1	0	Clock enabled	No clock	LAN Ethernet controller not enabled for wake on LAN. The corresponding MII interface is not driven during STR, STD, and SOFF.																						
1	1	Clock enabled	Clock enabled	LAN Ethernet controller enabled for wake on LAN.																						

## System Management Action Field Register

DevB:3x70

This register includes eight 4-bit groups, one for each of several system management events. The registers specify whether LDTSTOP\_L is asserted for the system management event and the system management action field (SMAF) associated with the event (LDTSTOP\_L cannot be asserted for C2, so this bit is reserved; LDTSTOP\_L is always asserted for VID/FID, so this bit is reserved).

For each LDTSTOP\_L assertion bit: 1=LDTSTOP\_L is asserted after the Stop Grant cycle associated with the STPCLK\_L assertion; LDTSTOP\_L is deasserted prior to each STPCLK deassertion message. 0=LDTSTOP\_L is not asserted for the system management event.

For each SMAF field: the bits are applied to bits[3:1] of the HyperTransport technology system management STPCLK assertion and deassertion messages sent to the host for the system management event.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31	<b>TTLS. Thermal throttling LDTSTOP_L assertion.</b>
30:28	<b>TTSMAF. Thermal throttling system management action field.</b>
27	<b>NTLS. Normal throttling LDTSTOP_L assertion.</b>
26:24	<b>NTSMAF. Normal throttling system management action field.</b>
23	Reserved.
22:20	<b>STRSMAF. Suspend to RAM system management action field.</b> This applies to the STPCLK cycle associated with STR after PM04[SLP_TYP] is written with the STR value.

19	<b>POSLS. Power on suspend LDTSTOP_L assertion.</b> This applies to the Stop Grant cycle associated with POS after PM04[SLP_TYP] is written with the POS value.
18:16	<b>POSSMAF. Power on suspend system management action field.</b> This applies to the STPCLK cycle associated with POS after PM04[SLP_TYP] is written with the POS value.
15	Reserved.
14:12	<b>VFSMAF. VID/FID system management action field.</b> This applies to the STPCLK cycle associated with the HyperTransport™ technology system management VID/FID change request message.
11	<b>C3LS. C3 LDTSTOP_L assertion.</b> This applies to the Stop Grant cycle associated with C3 after PM15 is accessed.
10:8	<b>C3SMAF. C3 system management action field.</b> This applies to the STPCLK cycle associated with C3 after PM15 is accessed.
7	Reserved.
6:4	<b>C2SMAF. C2 system management action field.</b> This applies to the STPCLK cycle associated with C2 after PM14 is accessed.
3	Reserved.
2:0	<b>S45SMAF. S4/S5 system management action field.</b> This applies to the STPCLK cycle associated with S4 and S5 after PM04[SLP_TYP] is written with the either of these value.

**VID/FID LDTSTOP\_L Time Register**

**DevB:3x74**

Default: 00h.

Attribute: Read-write.

Bits	Description																				
7:5	Reserved.																				
4:3	<p><b>C3S1LST. C3, S1 LDTSTOP_L assertion delay timer.</b> See Section 3.7.1.6 on page 57 for functional description of FON to C3 and FON to POS. (The times listed below are minimums, values up to 10% longer are within spec)</p> <table border="1"> <thead> <tr> <th>C3S1LST</th> <th>Time</th> <th>C3S1LST</th> <th>Time</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>1 microsecond</td> <td>10b</td> <td>64 microseconds</td> </tr> <tr> <td>01b</td> <td>32 microseconds</td> <td>11b</td> <td>128 microseconds</td> </tr> </tbody> </table>	C3S1LST	Time	C3S1LST	Time	00b	1 microsecond	10b	64 microseconds	01b	32 microseconds	11b	128 microseconds								
C3S1LST	Time	C3S1LST	Time																		
00b	1 microsecond	10b	64 microseconds																		
01b	32 microseconds	11b	128 microseconds																		
2:0	<p><b>FVLST. Frequency ID/Voltage ID LDTSTOP_L timer.</b> This field specifies the minimum duration of LDTSTOP_L assertion when any of the following occur:</p> <ul style="list-style-type: none"> <li>a) ACPI State transition for which LDTSTOP_L assertion is enabled,</li> <li>b) HyperTransport™ technology system management VID/FID change request,</li> <li>c) LDTSTOP command through DevB:3x78.</li> </ul> <p>The times shown below except FVLST=000b have a tolerance of +/- 10%.</p> <table border="1"> <thead> <tr> <th>FVLST</th> <th>Time</th> <th>FVLST</th> <th>Time</th> </tr> </thead> <tbody> <tr> <td>000b</td> <td>1 microseconds (min, +10%)</td> <td>100b</td> <td>16 microseconds</td> </tr> <tr> <td>001b</td> <td>2 microseconds</td> <td>101b</td> <td>32 microseconds</td> </tr> <tr> <td>010b</td> <td>4 microseconds</td> <td>110b</td> <td>64 microseconds</td> </tr> <tr> <td>011b</td> <td>8 microseconds</td> <td>111b</td> <td>128 microseconds</td> </tr> </tbody> </table>	FVLST	Time	FVLST	Time	000b	1 microseconds (min, +10%)	100b	16 microseconds	001b	2 microseconds	101b	32 microseconds	010b	4 microseconds	110b	64 microseconds	011b	8 microseconds	111b	128 microseconds
FVLST	Time	FVLST	Time																		
000b	1 microseconds (min, +10%)	100b	16 microseconds																		
001b	2 microseconds	101b	32 microseconds																		
010b	4 microseconds	110b	64 microseconds																		
011b	8 microseconds	111b	128 microseconds																		

**Miscellaneous Register****DevB:3x75**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>Must be Low.</b> Read-write. These bits are required to be Low at all times; if one bit is High then undefined behavior results.

**Link Frequency Change and Resize LDTSTOP\_L Command Register****DevB:3x78**

Default: (not applicable)

Attribute: Write-only.

Bits	Description
31:1	Reserved.
0	<b>LSCMD. LDTSTOP command.</b> Writing a 1 to this bit results in: (1) the IC generates a STPCLK assertion with SMAF bits as specified by DevB:3x70[C3SMAF], (2) upon receipt of the Stop Grant message, LDTSTOP_L is asserted for a period as specified by DevB:3x74[FVLST], and (3) after the HyperTransport™ link reconnects, the IC generates a STPCLK deassertion message.

**System Management Device and Subsystem ID Read-Write Register****DevB:3x7C**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> The value placed in this register is visible in DevB:3x2C[31:16].
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> The value placed in this register is visible in DevB:3x2C[15:0].

**Read-Write Register****DevB:3x80**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	<b>RW.</b> Read-write. These bits are read-write accessible through software; they control no hardware.



**CardBus Trap 1 and 2 I/O Address Register****DevB:3xB4**

DevB:3xB4, DevB:3xB8, DevB:3xBC, and DevB:3xC0 combine to specify the address space for the CARDBUS0 and CARDBUS1 trap events. These events can be used to generate SMIs or SCIs, load the associated re-trigger timers (PM6C and PM70), or load the system inactivity timer. The CARDBUS0 and CARDBUS1 trap events occur when the following is true:

CARDBUS0:

```
( (AD[15:0] | MASKPIO1 == ADDRPIO1 | MASKPIO1) & (PCI IO access) )
| ( (AD[31:10] | MASKPME1 == ADDRPME1 | MASKPME1) & (PCI MEM access) );
```

CARDBUS1:

```
( (AD[15:0] | MASKPIO2 == ADDRPIO2 | MASKPIO2) & (PCI IO access) )
| ( (AD[31:10] | MASKPME2 == ADDRPME2 | MASKPME2) & (PCI MEM access) );
```

Where AD is the address field of the host transaction. The mask bits for the I/O addresses cover bits[7:0]. The mask bits for the memory addresses cover bits[17:10].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>ADDRPIO2. I/O address for the CARDBUS1 trap event.</b>
15:0	<b>ADDRPIO1. I/O address for the CARDBUS0 trap event.</b>

**PCMCIA Trap 1 Memory Address Registers****DevB:3xB8**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:10	<b>ADDRPME1. Memory address for the CARDBUS0 trap event. See DevB:3xB4 for details.</b>
9:0	Reserved.

**PCMCIA Trap 2 Memory Address Registers****DevB:3xBC**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:10	<b>ADDRPME2. Memory address for the CARDBUS1 trap event. See DevB:3xB4 for details.</b>
9:0	Reserved.

**PCMCIA Trap Mask Registers****DevB:3xC0**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:24	<b>MASKPME2. Address mask for the PCMCIA trap events.</b> See DevB:3xB4 for details.
23:16	<b>MASKPME1. Address mask for the PCMCIA trap events.</b> See DevB:3xB4 for details.
15:8	<b>MASKPIO2. Address mask for the PCMCIA trap events.</b> See DevB:3xB4 for details.
7:0	<b>MASKPIO1. Address mask for the PCMCIA trap events.</b> See DevB:3xB4 for details.

**Programmable I/O Range Monitor 1 and 2 Trap Address Register****DevB:3xC4**

DevB:3xC4, DevB:3xC8, and DevB:3xCC combine to specify the address space for programmable I/O range monitor trap events (PIORM[4:1]). These events can be used to generate SMIs or SCIs, load the associated re-trigger timer (PM78, PM7C, PM80, and PM84), or load the system inactivity timer. The trap events occurs following equations are true:

```
PIORM1 : (AD[15:0] | MASKIO1 == ADDRIO1 | MASKIO1) ;
PIORM2 : (AD[15:0] | MASKIO2 == ADDRIO2 | MASKIO2) ;
PIORM3 : (AD[15:0] | MASKIO3 == ADDRIO3 | MASKIO3) ;
PIORM4 : (AD[15:0] | MASKIO4 == ADDRIO4 | MASKIO4) ;
```

Where AD is the address field a host I/O cycle. The mask bits cover any combination of bits[7:0].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>ADDRIO2. Address for the PIORM2 trap event.</b>
15:0	<b>ADDRIO1. Address for the PIORM1 trap event.</b>

**Programmable I/O Range Monitor 3 and 4 Trap Address Register****DevB:3xC8**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>ADDRIO4. Address for the PIORM4 trap event.</b> See DevB:3xC4 for details.
15:0	<b>ADDRIO3. Address for the PIORM3 trap event.</b> See DevB:3xC4 for details.

**Programmable I/O Range Monitor Trap Mask Register****DevB:3xCC**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:24	<b>MASKIO4. Address masks for the PIORM4 trap events.</b> See DevB:3xC4 for details.
23:16	<b>MASKIO3. Address masks for the PIORM3 trap events.</b> See DevB:3xC4 for details.
15:8	<b>MASKIO2. Address masks for the PIORM2 trap events.</b> See DevB:3xC4 for details.
7:0	<b>MASKIO1. Address masks for the PIORM1 trap events.</b> See DevB:3xC4 for details.

**Programmable Memory/Configuration Range Monitor 1 Trap Address Register****DevB:3xD0**

DevB:3xD0, DevB:3xD4, and DevB:3xD8 combine to specify the address space for the programmable memory or configuration space range monitor 1 and 2 trap events (PMEMRM[1,2]). These events can be used to generate SMIs or SCIs, load the associated re-trigger timers (PM88 and PM8C), or load the system inactivity timer. These trap events occur when the following equations are true:

```

PMEMRM1: (AD[31:2] | MASKMEM1 == ADDRMEM1[31:2] | MASKMEM1) & MEMSP & ~CFGSPEN1
          | (AD[15:2] | MASKMEM1 == ADDRMEM1[15:2] | MASKMEM1) & CFGSP & CFGSPEN1 &
            ~AD[24] & ~ADDRMEM1[24]
          | (AD[23:2] | MASKMEM1 == ADDRMEM1[23:2] | MASKMEM1) & CFGSP & CFGSPEN1 &
            AD[24] & ADDRMEM1[24];
PMEMRM2: (AD[31:2] | MASKMEM2 == ADDRMEM2[31:2] | MASKMEM2) & MEMSP & ~CFGSPEN2
          | (AD[15:2] | MASKMEM2 == ADDRMEM2[15:2] | MASKMEM2) & CFGSP & CFGSPEN2 &
            ~AD[24] & ~ADDRMEM2[24]
          | (AD[23:2] | MASKMEM2 == ADDRMEM2[23:2] | MASKMEM2) & CFGSP & CFGSPEN2 &
            AD[24] & ADDRMEM1[24];

```

Where AD is the address field of a host transaction, MEMSP indicates a memory space transaction and CFGSP indicates a configuration space transaction. The mask bits cover address bits[17:2]. Note that for configuration traps, AD[24] and ADDRMEM[2,1][24] carry the determination of whether the trap is a type 0 or type 1 configuration cycle; if it is type 0, then the bus number bits are ignored.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:2	<b>ADDRMEM1. Memory address for the PMEMRM1 trap event.</b>
1	Reserved.
0	<b>CFGSPEN1. Configuration space enable 1.</b> 1=PMEMRM1 is a configuration space trap. 0=PMEMRM1 is a memory space trap.

**Programmable Memory/Configuration Range Monitor 2 Trap Address Register**      **DevB:3xD4**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:2	<b>ADDRMEM2. Memory address for the PMEMRM2 trap event.</b> See DevB:3xD0 for details.
1	Reserved.
0	<b>CFGSPEN2. Configuration space enable 2.</b> 1=PMEMRM2 is a configuration space trap. 0=PMEMRM2 is a memory space trap. See DevB:3xD0 for details.

**Programmable Memory/Configuration Range Monitor Trap Mask Registers**      **DevB:3xD8**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>MASKMEM2. Address mask for the PMEMRM2 trap event.</b> See DevB:3xD0 for details.
15:0	<b>MASKMEM1. Address mask for the PMEMRM1 trap event.</b> See DevB:3xD0 for details.

**4.7.2 System Management I/O Mapped Registers (PMxx)**

These registers are located in I/O space. The base address registers for these registers is DevB:3x58. See Section 4.1.2 on page 136 for a description of the register naming convention.

**Power Management 1 Status Register (ACPI PM1a\_STS)****PM00**

Each of these bits are status bits set by hardware events. Most have the ability to generate an SCI/SMI interrupt, if they are enabled to do so in PM02.

Default: 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
15	<b>WAK_STS. Wake status.</b> This bit is set by hardware when the system is in any sleep state (POS, STR, STD, or SOFF) and an enabled resume event occurs. Upon setting this bit, the system resumes.
14:12	Reserved.
11	<b>PBOR_STS. Power button override status.</b> This bit is set by hardware when a power button override event occurs. A power button override event occurs if PM26[PBOR_DIS] is Low and PWRBTN_L is held in the active state for more than four seconds or if PM26[SBOR_DIS] is Low, the SLPBTN_L function is enabled by PMD7, and SLPBTN_L is held in the active state for more than four seconds. This bit resides on the VDD_COREX power plane.
10	<b>RTC_STS. Real-time clock status.</b> This bit is set by hardware when the real-time clock generates an interrupt. This bit resides on the VDD_COREX power plane.

Bits	Description
9	<b>SLPBTN_STS. Sleep button status.</b> 1=Indicates that the sleep button (SLPBTN_L) has been asserted. The debounce circuitry causes a 12-to-16 millisecond delay from the time the input signal stabilizes until this bit changes. If the GPIO debounce circuitry specified by PMD7 is enabled, then the debounce period is twice as long before setting the status bit. If the SLPBTN_L function is not selected by PMD7, then this bit is not set. If PM26[SBOR_DIS] is Low and SLPBTN_L is held Low for more than four seconds, then this bit is cleared and PBOR_STS is set. This bit resides on the VDD_COREX power plane. Note: the debounce circuit functions in the High-to-Low and Low-to-High directions.
8	<b>PWRBTN_STS. Power button status.</b> 1=Indicates that the power button (PWRBTN_L) has been asserted. The debounce circuitry causes a 12-to-16 millisecond delay from the time the input signal stabilizes until this bit changes. If PM26[PBOR_DIS] is Low and PWRBTN_L is held Low for more than four seconds, then this bit is cleared and PBOR_STS is set. This bit resides on the VDD_COREX power plane. Note: the debounce circuit functions in the High-to-Low and Low-to-High directions.
7:6	Reserved.
5	<b>GBL_STS. Global status.</b> This bit is set by hardware when a 1 is written to PM2C[BIOS_RLS].
4	<b>BM_STS. Bus master status.</b> This bit is set by hardware when a secondary PCI bus request signal becomes active, LDTREQ_L is asserted, or any internal source requests access to the host. Based on the state of PM04[BM_RLD], this may result in a power state transition. Note: this bit is not set by HyperTransport™ technology system management messages, HyperTransport interrupt requests, and HyperTransport fence or flush commands.
3:1	Reserved.
0	<b>TMR_STS. ACPI timer status.</b> This bit is set by hardware when the MSB (either bit 23 or 31 based on DevB:3x41[3]) of the ACPI timer (PM08) toggles (from 0 to 1 or 1 to 0).

**Power Management 1 Enable Register (ACPI PM1a\_EN)****PM02**

Most of these bits work in conjunction with the corresponding STS bits in PM00 to generate SCI or SMI interrupts (based on the state of PM04[SCI\_EN]).

Default: 0100h.

Attribute: Read-write.

Bits	Description
15:11	Reserved.
10	<b>RTC_EN. Real-time clock alarm SCI/SMI enable.</b> 1=Enable an SCI or SMI interrupt when PM00[RTC_STS] is set High. This bit resides on the VDD_COREX power plane.
9	<b>SLPBTN_EN. Sleep button SCI/SMI enable.</b> 1=Enable an SCI or SMI interrupt when PM00[SLPBTN_STS] is set High. This bit resides on the VDD_COREX power plane.
8	<b>PWRBTN_EN. Power button SCI/SMI enable.</b> 1=Enable an SCI or SMI interrupt when PM00[PWRBTN_STS] is set High. This bit resides on the VDD_COREX power plane.
7:6	Reserved.
5	<b>GBL_EN. Global SCI enable.</b> 1=Enable an SCI interrupt when PM00[GBL_STS] is set High. Note: this results in an SCI interrupt, regardless as to the state of PM04[SCI_EN].
4:1	Reserved.
0	<b>TMR_EN. ACPI timer SCI enable.</b> 1=Enable an SCI interrupt when PM00[TMR_STS] is set High. Note: this results in an SCI interrupt, regardless as to the state of PM04[SCI_EN].

**Power Management 1 Control Register (ACPI PM1\_CNTa)****PM04**

Default: 0000h.

Attribute: See below.

Bits	Description
15:14	Reserved.
13	<b>SLP_EN. Sleep enable.</b> Write-only; reads back as 0. Writing a 1 to this bit causes the system to sequence into the sleep state specified by SLP_TYP.
12:10	<b>SLP_TYP. Sleep type.</b> Read-write. Specifies the type of sleep state the system enters when SLP_EN is set High.
0	FON, S0. Full on.
1	POS, S1. Power on suspend. Power state transition specified by DevB:3x50.
2-4	Reserved.
5	STR, S3. Suspend to RAM.
6	STD, S4. Suspend to disk.
7	SOFF, S5. Soft off.
9:3	Reserved.
2	<b>GBL_RLS. Global release.</b> Read; write 1 only; cleared by hardware. When this bit is set High, the hardware sets PM28[BIOS_STS] High. GBL_RLS is cleared by the hardware when PM28[BIOS_STS] is cleared by software.
1	<b>BM_RLD. Bus master reload.</b> Read-write. 1=Enables the transition of the processor power state from C3 to C0 to be triggered by any bus master requests or HyperTransport™ source link requests (when PM00[BM_STS] is set).
0	<b>SCI_EN. SCI-SMI select.</b> Read-write. Selects the type of interrupt generated by power management events. 0=SMI interrupt. 1=SCI interrupt. Note that certain power management events can be programmed individually to generate SMI interrupts independent of the state of this bit. See Section 3.7.1.1 on page 54 for details. Also, TMR_STS and GBL_STS always generate SCI interrupts regardless as to the state of this bit.

**ACPI Power Management Timer (ACPI PM\_TMR)****PM08**

This is either a 24- or a 32-bit counter, based on the state of DevB:3x41[3]. It is a free-running, incrementing counter clocked off of a 3.579545 MHz. clock. It does not count when in the system is in MOFF, SOFF, STD, or STR state. When the MSB toggles (either bit[23] or bit[31]) then PM00[TMR\_STS] is set. This timer is asynchronously cleared when DevB:3x41[TMRRST] is

High.Default:0000 0000h.

Attribute: Read-only.

Bits	Description
31:24	<b>ETM_VAL. Extended timer value.</b> If DevB:3x41[3] is High, then these are the 8 MSBs of the ACPI power management timer. If DevB:3x41[3] is Low, then this field always reads back as all zeros.
23:0	<b>TMR_VAL. Timer value.</b> Read-only. This field returns the running count of the ACPI power management timer.

**Processor Clock Control Register (ACPI P\_CNT)****PM10**

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description																				
31:5	Reserved.																				
4	<b>NTH_EN. Normal throttling enable.</b> 1=Normal throttling (duty cycle specified by NTH_RATIO) is enabled. Normal throttling is disabled when thermal throttling. 0=Normal throttling is not enabled.																				
3:1	<b>NTH_RATIO. Normal throttling duty cycle.</b> These bits specify the duty cycle of the STPCLK_L pin when the system is in normal throttling mode, enabled by NTH_EN. The field is decoded as follows:																				
	<table border="1"> <thead> <tr> <th>NTH_RATIO</th> <th>Description</th> <th>NTH_RATIO</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0h</td> <td>Reserved</td> <td>4h</td> <td>50.0% in Stop Grant state</td> </tr> <tr> <td>1h</td> <td>87.5% in Stop Grant state</td> <td>5h</td> <td>37.5% in Stop Grant state</td> </tr> <tr> <td>2h</td> <td>75.0% in Stop Grant state</td> <td>6h</td> <td>25.0% in Stop Grant state</td> </tr> <tr> <td>3h</td> <td>62.5% in Stop Grant state</td> <td>7h</td> <td>12.5% in Stop Grant state</td> </tr> </tbody> </table>	NTH_RATIO	Description	NTH_RATIO	Description	0h	Reserved	4h	50.0% in Stop Grant state	1h	87.5% in Stop Grant state	5h	37.5% in Stop Grant state	2h	75.0% in Stop Grant state	6h	25.0% in Stop Grant state	3h	62.5% in Stop Grant state	7h	12.5% in Stop Grant state
NTH_RATIO	Description	NTH_RATIO	Description																		
0h	Reserved	4h	50.0% in Stop Grant state																		
1h	87.5% in Stop Grant state	5h	37.5% in Stop Grant state																		
2h	75.0% in Stop Grant state	6h	25.0% in Stop Grant state																		
3h	62.5% in Stop Grant state	7h	12.5% in Stop Grant state																		
0	Reserved.																				

**Processor Level 2 Register (ACPI P\_LVL2)****PM14**

Default: 00h.

Attribute: Read-only.

Bits	Description
7:0	<b>P_LVL2.</b> Reads from this register place the processor into the C2 power state, as specified by DevB:3x4F. Reads from this register always return 00h. This register is byte readable only.

**Processor Level 3 Register (ACPI P\_LVL3)****PM15**

Default: 00h.

Attribute: Read-only.

Bits	Description
7:0	<b>P_LVL3.</b> Reads from this register place the processor into the C3 power state, as specified by DevB:3x4F. Reads from this register always return 00h. This register is byte readable only.

**Power Management Control Register****PM1C**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:1	Reserved.
0	<b>ARB_DIS. Arbiter disable.</b> 1=Disable all secondary PCI masters from being granted the bus and all internal HyperTransport™ sources from initiating memory space transactions. This is in support of the ACPI C3 state in which the processor is not capable of supporting snoops. 0=Enable.

**Software SMI Trigger Register****PM1E (PM2F)**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	This address accesses the same physical register located at PM2F. I.e., both accesses to PM1E and PM2F identically access the same register and both can be used to set PM28[SWI_STS].

**Fan RPM Count Register****PM1F**

Default: 00h.

Attribute: Read-only.

Bits	Description
7:0	<b>RPMCNT. FANRPM Count Register.</b> This provides the state of a counter that increments on every rising edge of the signal FANRPM. If the FANRPM function is not selected by PMCA, then this register does not change.

**General Purpose 0 Status Register (ACPI GPE0\_STS)****PM20**

Most of the bits in this register may be enabled to generate SCI or SMI interrupts (based on the state of PM04[SCI\_EN]) through PM22 or may be enabled to generate SMI interrupts through PM2A and PM2C.

Default: 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
15	<b>USBRSM_STS. USB-defined resume event status.</b> This bit is set High by the hardware when a USB-defined resume event has occurred from either USB controller. This bit resides on the VDD_COREX power plane.
14	<b>RI_STS. RI_L pin status.</b> This bit is set High by the hardware when the RI_L pin is asserted. This bit resides on the VDD_COREX power plane.
13	<b>LID_STS. LID Status.</b> This bit is set High by the hardware when the LID signal has changed state indicating that the LID has either opened or closed. This bit resides on the VDD_COREX power plane.
12	<b>ACAV_STS. AC change status.</b> This bit is set High by the hardware when the ACAV signal has changed state indicating that AC power has either been added or removed from the system. This bit resides on the VDD_COREX power plane. Note: the default value of this bit (from MOFF) is indeterminate.
11	<b>SMBUS_STS. SMBus status.</b> This bit is set High if an SMBus status bit in PME0[SNP_STS, HSLV_STS, and SMBA_STS] is High while enabled by PME2[SNP_EN, HSLV_EN, and SMBA_EN], respectively, or if any of PME0[ABRT_STS, COL_STS, PRERR_STS, HCYC_STS, TO_STS] are set High while enabled by PME2[HCYC_EN]. Note: only PME0[SMBA_EN, HSLV_STS, SNP_STS] can be enabled to wake the system out of sleep states. This bit resides on the VDD_COREX power plane.
10	<b>THERM_STS. THERM_L pin status.</b> This bit is set High by the hardware when the THERM_L pin is asserted.
9	<b>EXTSMI_STS. External SMI pin status.</b> This bit is set High by the hardware when the EXTSMI_L pin is asserted. This bit resides on the VDD_COREX power plane.



Bits	Description (Continued)
8	<b>PME_STS. PME_L pin status.</b> This bit is set High by the hardware when the PME_L pin is asserted Low. This bit may be set High by the PME function from the Ethernet controller. This bit resides on the VDD_COREX power plane.
7	<b>TCOSCI_STS. TCO SCI interrupt status.</b> This bit is set High by the hardware when there is a 0 to 1 transition on PM46[INTRDR_STS] or PM44[TCO_INT_STS].
6	Reserved.
5	<b>SIT_STS. System inactivity timer time out status.</b> This bit is set High by the hardware when the system inactivity timer times out.
4	Reserved.
3	<b>SMBC_STS. SMBus controller system management event.</b> This bit is set High by the SMBus controller. This bit resides on the VDD_COREX power plane.
2	Reserved. Software should ignore this bit.
1	<b>AC97_STS. AC '97 wake event status.</b> This bit is set High by the hardware when the AC '97 link generates a wake event. This bit resides on the VDD_COREX power plane. This bit is also set if AC30/MC40[GPIINT] is '1'.
0	<b>DM_STS. Hardware device monitor (parent) event status.</b> This bit is set High by the hardware when any of the device monitor event status bits in PMA0 become active when enable by the corresponding bits in PMA4.

**General Purpose 0 ACPI Interrupt Enable Register (ACPI GPE0\_EN)****PM22**

For each of the bits in this register: 1=Enable a corresponding status bit in PM20 to generate an SMI or SCI interrupt (based on the state of PM04[SCI\_EN]). 0=Do not enable the SMI or SCI interrupt.

Default: 0000h.

Attribute: Read-write.

Bits	Description
15	<b>USBRSM_EN. USB resume event ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
14	<b>RI_EN. RI_L pin ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
13	<b>LID_EN. LID pin ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
12	<b>ACAV_EN. ACAV pin ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
11	<b>SMBUS_EN. SMBus (parent) ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
10	<b>THERM_EN. THERM_L pin ACPI interrupt enable.</b>
9	<b>EXTSMI_EN. External SMI pin ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
8	<b>PME_EN. PME_L pin ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
7	<b>TCOSCI_EN. TCO SCI enable.</b> Note: when this is High, DevB:3x44[TCO_INT_SEL] is ignored.
6	Reserved.
5	<b>SIT_EN. System inactivity timer time out ACPI interrupt enable.</b>
4	Reserved.

Bits	Description
3	<b>SMBC_EN. SMBus controller ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
2	Reserved. Software should only write '0' to this bit.
1	<b>AC97_EN. AC '97 wake event ACPI interrupt enable.</b> This bit resides on the VDD_COREX power plane.
0	<b>DM_EN. Device monitor (parent) ACPI interrupt enable.</b>

### Sleep State Resume Enable Register

PM26

All the bits in this register reside in the VDD\_COREX power plane.

Default: 2000h.

Attribute: Read-write.

Bits	Description
15	<b>BATLOW_CTL. Battery low enable control.</b> 1=Enable BATLOW_L assertion to prevent a system resume from any sleep state (POS, STD, STR, or SOFF) if PMC0 enables ACAV functionality and the ACAV input is Low or if PMC0 disables ACAV functionality. All resume events except that associated with ACAV_STS are affected by this bit.
14	<b>MINSLEEP. Minimum sleep time.</b> 1=The minimum time in which PWRON_L is deasserted in support of STR, STD, or SOFF is 500 to 800 milliseconds. I.e., resume events that occur shortly after entry into these sleep states are not acted upon until this minimum time has transpired. 0=There is no minimum time in which PWRON_L is deasserted.
13	<b>SBOR_DIS. SLPBTN_L override disable.</b> 1=The power button override event from the SLPBTN_L pin (holding SLPBTN_L active for four seconds) does not set PM00[PBOR_STS] High and does not automatically transition the system into SOFF. 0=The power button override event from the SLPBTN_L pin is enabled to set PM00[PBOR_STS] High and place the system into the SOFF mode.
12	Reserved.
11	<b>PBOR_DIS. Power button override disable.</b> 1=The power button override event (holding PWRBTN_L active for four seconds) does not set PM00[PBOR_STS] High and does not automatically transition the system into SOFF. 0=The power button override event is enabled to set PM00[PBOR_STS] High and place the system into the SOFF mode.
10:0	Reserved.

**Global Status Register****PM28**

Each of the EVT bits specify SMI-interrupt-enabled status bits in other registers. These are not sticky bits; they reflect the combinatorial equation of:  $\_EVT = (status1 \& SMI\ enable1) \mid (status2 \& SMI\ enable2)...$

Default: 0000h.

Attribute: See below.

Bits	Description
15	<b>MISC_EVT. Miscellaneous SMI event.</b> Read-only. This bit is read as 1 when there are set status bits in PM30 that are enabled in PM32.
14	Reserved.
13	<b>GPE0_EVT. General purpose event 0 event status.</b> Read-only. This bit goes High when any of the PM20 status bits that are SMI enabled by PM2A or PM2C become active (this does not include PM20[DM_STS]). Note: PM20[TCOSCI_STS] is not included.
12	<b>USB_EVT. USB SMI event.</b> Read-only. This bit is read as 1 when one of the USB-defined SMI events occurs in either USB controller. This occurs when HcControl[8] is High and an enabled interrupt occurs (HcInterruptStatus and HcInterruptControl). This bit is not affected by PM20[USBRSM_STS].
11	<b>SMBUS_EVT. SMBus event status.</b> Read-only. This bit is read as 1 when an SMBus status bit in PME0[SNP_STS, HSLV_STS, and SMBA_STS] is High while enabled by PME2[SNP_EN, HSLV_EN, and SMBA_EN], respectively, or when any of PME0[ABRT_STS, COL_STS, PRERR_STS, HCYC_STS, TO_STS] are set High while enabled by PME2[HCYC_EN].
10:8	Reserved.
7	<b>SWI_STS. Software SMI status.</b> Read; set by hardware; write 1 to clear. This bit is set High by the hardware when a write of any value is sent to PM1E or PM2F. This bit can trigger SMI interrupts if enabled by PM2A[SWISMI_EN].
6	<b>BIOS_STS. BIOS status.</b> Read; set by hardware; write 1 to clear. This bit is set High by the hardware when PM04[GBL_RLS] is set High. BIOS_STS is cleared when a 1 is written to it; writing a 1 to BIOS_STS also causes the hardware to clear PM04[GBL_RLS]. This bit can trigger SMI interrupts if enabled by PM2A[BIOSSMI_EN].
5	Reserved.
4	<b>IRQRSM_STS. IRQ Resume Status.</b> Read; set by hardware; write 1 to clear. 1=System was resumed from POS due to an unmasked interrupt assertion and PM2A[IRQ_RSM] is High. 0=System was not resumed from POS due to an interrupt. Note: this bit is set only by resumes from POS. Note: DevB:3x50[PITRSM_L, MSRSM_L] can be set to inhibit timer tick and mouse interrupts during POS.
3	<b>GPIO_EVT. GPIO interrupt status.</b> Read-only. This bit is read as 1 when any of the general purpose I/O status bits in PMB0 that are SMI enabled by PMB8 become active.
2	<b>PM1_EVT. Power Management 1 status.</b> Read-only. This bit is read as 1 when any of the status bits in PM00 are active (they do not need to be enabled by PM02).
1	<b>TCO_EVT. TCO SMI interrupt event.</b> Read-only. This bit is read as 1 when any of PM44[NMI2SMI_STS, SW_TCO_SMI, TOUT_STS, IBIOS_STS] are set.
0	<b>DM_EVT. Hardware device monitor event status.</b> Read-only. This bit is read as 1 when any of the device monitor status bits in PMA0 that are SMI enabled by PMA8 become active.

**Global SMI Enable Register****PM2A**

Most of these bits enable the status bits to generate SMI interrupts. For each of these bits: 1=enable the corresponding STS bit in the specified register to generate an SMI interrupt, regardless as to the state of PM04[SCI\_EN].

Default: 0000h.

Attribute: Read-write.

Bits	Description
15	<b>USBSMI_EN. USB resume SMI enable.</b> Generate SMI when PM20[USBRSM_STS]=1.
14	<b>RISMI_EN. RI_L pin SMI enable.</b> Generate SMI when PM20[RI_STS]=1.
13	<b>SBSMI_EN. SLPBTN_L pin SMI enable.</b> Generate SMI when PM00[SLPBTN_STS]=1.
12	<b>PBSMI_EN. PWRBTN_L pin SMI enable.</b> Generate SMI when PM00[PWRBTN_STS]=1.
11	<b>SMBSMI_EN. SMBus event SMI enable.</b> Generate SMI when PM28[SMBUS_EVT]=1.
10	<b>THMSMI_EN. THERM_L pin SMI enable.</b> Generate SMI when PM20[THERM_STS]=1.
9	<b>EXTSMI_EN. External SMI pin SMI enable.</b> Generate SMI when PM20[EXTSMI_STS]=1.
8	<b>PMESMI_EN. PME_L pin SMI enable.</b> Generate SMI when PM20[PME_STS]=1.
7	<b>SWISMI_EN. Software SMI enable.</b> Generate SMI when PM28[SWI_STS]=1.
6	<b>BIOSMI_EN. BIOS SMI enable.</b> Generate SMI when PM28[BIOS_STS]=1.
5	<b>SITSMI_EN. System inactivity timer time out SMI enable.</b> Generate SMI when PM20[SIT_STS]=1.
4	<b>IRQ_RSM. Resume from POS on unmasked interrupt.</b> 1=Enable resume from POS when PM28[IRQRSM_STS] is set High.
3	<b>BMSMI_EN. Bus master SMI enable.</b> Generate SMI when PM00[BM_STS]=1.
2	<b>LIDSMI_EN. LID pin SMI enable.</b> Generate SMI when PM20[LID_STS]=1.
1	<b>TCO_EN. TCO SMI interrupt enable.</b> Generate SMI when PM28[TCO_EVT]=1. Note: if PM48[NMI2SMI_EN]=1, PM44[NMI2SMI_STS] generates SMI interrupts regardless of the state of this bit. Even if the TCO_EN bit is 0, NMIs are routed to generate SMIs.
0	<b>ACAVSMI_EN. ACAV pin SMI enable.</b> Generate SMI when PM20[ACAV_STS]=1.

**Global SMI Control Register****PM2C**

Default: 0000h.

Attribute: See below.

Bits	Description
15:8	Reserved.
7	<b>SMBCSMI_EN. SMBus controller system management event SMI enable.</b> Read-write. 1=An SMI is generated if PM20[SMBC_STS] goes High.
6	<b>Reserved.</b> Software should only write a '0' to this bit.
5	<b>SMIACT. SMI active.</b> Read; set by hardware; write 1 to clear. This bit is set High by the hardware on the leading edge of the SMI output. If SMILK is High, then SMIACT holds the SMI_L pin in the active state. If SMILK is Low, then SMIACT has no effect on the SMI signal.
4	<b>SMILK. SMI lock control.</b> Read-write. 1=The SMI_L pin is locked into the active state by a latch after it is asserted. The latch is controlled by SMIACT. 0=The state of SMIACT does not affect SMI_L.

3	<b>EOS. End of SMI.</b> Write-only. Writing a 1 to this bit forces the SMI_L pin to be deasserted for 4 PCI clocks. This bit always reads as a 0.
2	<b>AC97SMI_EN. AC '97 wake event SMI enable.</b> Read-write. 1=Generate SMI when PM20[AC97_STS] goes High.
1	<b>BIOS_RLS. BIOS SCI/SMI lock release.</b> Read; write 1 only; cleared by hardware. This bit is set High by software to indicate the release of the SCI/SMI lock. When this bit is set High, PM00[GBL_STS] is set High by the hardware. BIOS_RLS is cleared by the hardware when PM00[GBL_STS] is cleared by software. Note that if PM02[GBL_EN] is set, then setting this bit generates an SCI interrupt.
0	<b>SMI_EN. SMI enable control.</b> Read-write. 1=Enable generation of SMIs from the system management logic (including all SMI sources in the system management logic and the SMI from the serial IRQ logic, but not SMIs from the IOAPIC). 0=Disable SMIs from the system management logic (however, if SMIACT is set and SMI_EN is cleared, SMI_L remains asserted until SMIACT is cleared).

**Advanced Power Management Status Register****PM2E**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>APM_STATUS.</b> This has no affect on hardware. This register may be used to pass status information between the OS and the SMI handler.

**Software SMI Trigger Register****PM2F**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>SMI_CMD. SMI command.</b> Writes to this register set PM28[SWI_STS]. If PM2A[SWISMI_EN] is set, then writes to this port generate SMI interrupts. Reads of this register provide the data last written to it. This register is accessible PM1E as well.

**Miscellaneous SMI Status Register****PM30**

These bits may be enabled to generate SMI interrupts through PM32.

Default: 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
15:5	Reserved.
4	<b>64MS_STS. 64 millisecond timer status.</b> After PM32[64MS_EN] is set High, the 64 millisecond timer counts out 64 +/- 4 milliseconds and then sets this bit. The timer continues counting the next 64 millisecond period after this bit is set.

3	<b>1MIN_STS. One minute status bit.</b> After entering the FON state, this bit is set every 60 +/- 4 seconds.
2	<b>SIRQSMI_STS. Serial IRQ SMI status.</b> This bit is set an SMI is transferred into the IC through the serial IRQ.
1	<b>RWR_STS. BIOS ROM write enable status.</b> This bit is set when DevB:0x40[RWR] is written from a 0 to a 1.
0	<b>SLPCMD_STS.</b> This bit is set if there is a write of PM04[SLP_EN] to a High.

### Miscellaneous SMI Enable Register

PM32

For each of the bits in this register: 1=enable a corresponding status bit in PM30 to generate an SMI interrupt. 0=Do not enable the SMI interrupt.

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:5	Reserved.
4	<b>64MS_EN. 64 millisecond SMI enable.</b> 1=Enable the 64 millisecond counter, as well as the corresponding SMI interrupt. 0=The 64 millisecond timer is cleared.
3	<b>1MIN_EN. One minute SMI enable.</b>
2	<b>SIRQSMI_EN. Serial IRQ SMI enable.</b>
1	<b>RWR_EN. BIOS ROM write enable SMI enable.</b> Read, write to 1 only. Once this bit is set, it can only be cleared by RESET_L.
0	<b>SLPCMD_EN. Enable SMI on sleep command.</b> Note: when this bit is set High and the sleep command is sent to PM04, the system power state is disabled from changing. It is expected that the SMI interrupt service routine will clear PM30[SLPCMD_STS], clear this bit, and then re-issue the command in order to change the system power state.

### TCO Timer Reload and Current Value Register

PM40

The TCO timer is a decrementing counter that is clocked approximately every 0.6 seconds providing times of up to 38 seconds. If it decrements past zero, it sets PM44[TOUT\_STS], rolls over to the value in PM41, and keeps counting.

Default: 04h.

Attribute: Read; write command.

Bits	Description
7:6	Reserved.
5:0	<b>TCORLD. TCO timer.</b> Reads of this register return the current count of the TCO timer. Writes of any value to this register cause the TCO timer to be reloaded with the value in PM41.

**TCO Timer Initial Value Register****PM41**

Default: 04h.

Attribute: Read-write.

Bits	Description
7:6	Reserved.
5:0	<b>TCOTIME. TCO timer reload value.</b> Specifies the value to be loaded into the TCO timer, PM40.

**TCO SMI Data In Register****PM42**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>TCOSMI. TCO SMI data.</b> Writes to this register set PM44[SW_TCO_SMI] and generate an SMI.

**TCO SMI Data Out Register****PM43**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>TCOOUT. TCO output data to OS.</b> Writes to this register set PM44[TCO_INT_STS] and generate an IRQ as specified by DevB:3x44[TCO_INT_SEL] and PM22[TCOSCI_EN].

**TCO Status 1 Register****PM44**

Default: 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
15:9	Reserved.
8	<b>IBIOS_STS. BIOS illegal access status.</b> 1=Hardware sets this bit and an SMI is generated (if PM2A[TCO_EN]=1) as a result of an illegal access to BIOS address space; this occurs when: (1) there is a read to a read-locked address or a write to a write-locked address as specified by DevB:0x40[RWR], DevB:0x80, DevB:0x84, DevB:0x88, and DevB:0x8C[3:0] or (2) DevB:0x40[BLE]=1 and DevB:0x40[RWR] is written from a 0 to a 1.
7:4	Reserved.
3	<b>TOUT_STS. TCO timer timeout status.</b> 1=Hardware sets this bit and an SMI is generated (if PM2A[TCO_EN]=1) as a result of the TCO timer (PM40) counting past zero.
2	<b>TCO_INT_STS. TCO interrupt status.</b> 1=Hardware sets this bit and an IRQ is generated (as specified by DevB:3x44[TCO_INT_SEL] and PM22[TCOSCI_EN]) by a write to PM43.
1	<b>SW_TCO_SMI. Software-generated SMI status.</b> 1=Hardware sets this bit and an SMI is generated (if PM2A[TCO_EN]=1) by a write to PM42.
0	<b>NMI2SMI_STS. NMI to SMI status.</b> 1=Hardware sets this bit and an SMI is generated as a result of an NMI while PM48[NMI2SMI_EN] is High. Note: PM48[NMI_NOW] does not set this status bit.

**TCO Status 2 Register****PM46**

Default: 00h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
7:4	Reserved.
3	<b>TT_STS. THERMTRIP status.</b> 1= THERMTRIP_L was asserted while PWRON_L = 0, PWROK = 1, and RESET_L = 1. This bit resides in the VDD_COREX power plane.
2	<b>BOOT_STS. Boot status.</b> 1= Hardware sets this bit when PM46[2NDTO_STS] changes from 0 to 1 after any RESET_L before any ROM accesses have occurred. This bit resides on the VDD_COREX power plane.
1	<b>2NDTO_STS. Second TCO time out status.</b> 1= Hardware sets this bit when the TCO timer, PM40, times out a second time before PM44[TOUT_STS] is cleared. If enabled by DevB:3x48[NO_REBOOT], this may trigger a reboot of the system. This bit resides on the VDD_COREX power plane.
0	<b>INTRDR_STS. Intruder detect status.</b> 1= Hardware sets this bit when the INTRUDER_L pin is asserted for more than 60 microseconds (debounce time); this bit functions in all power states (unless AL is not powered). This bit resides in the VDD_COREAL power plane; when VDD_COREAL is powered, this bit defaults to 0. This may trigger an interrupt as specified by PM4A[INTRDR_SEL].

**TCO Control 1 Register****PM48**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:12	Reserved.
11	<b>TCOHALT. TCO timer halt.</b> 1= Freeze the TCO timer (PM40) in its current state; PM44[TOUT_STS] and PM46[2NDTO_STS] cannot be set.
10	Reserved.
9	<b>NMI2SMI_EN. NMI interrupts generate SMI interrupts.</b> 1= Whenever an NMI is detected, PM44[NMI2SMI_STS] is set and no NMI is generated; this bit does not affect NMI_NOW (setting NMI_NOW generates an NMI regardless of the state of NMI2SMI_EN). Note: if this bit is set and RTC70[NMIDIS] is set, then no NMIs or associated SMIs are generated.
8	<b>NMI_NOW. Generate NMI.</b> 1= Generate NMI. It is expected that the NMI handler clears this bit.
7:0	Reserved.

**TCO Control 2 Register****PM4A**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:3	Reserved.
2:1	<b>INTRDR_SEL.</b> Select the action to take if PM46[INTRDR_STS] is set. 00b=Reserved; 01b=IRQ (as specified by DevB:3x44[TCO_INT_SEL]); 10b=SMI; 11b=Reserved.
0	Reserved.



**Read write Register****PM4C**

All these bits reside on the VDD\_COREX power plane.

Default: 0000h.

Attribute: Read-write.

Bits	Description
31:24	Reserved.
23:0	<b>RW.</b> These bits control no hardware.

**Timer and Device Monitor Registers****PM[8C:50] Bits[19:0]**

Each of these registers provide access to the re-trigger timers. Each timer decrements at a rate specified by CLKSRC when enabled. Each timer is associated with device monitor events including address traps, DMA acknowledge cycles and interrupt requests. Each time a device monitor event associated with a re-trigger timer occurs, the timer is reloaded. So, if the hardware is regularly accessed, then the timer never reaches zero. If the timer decrements past zero, then it is disabled from counting further (staying at zero) and the appropriate device monitor status bit is set. Here is a summary of the device monitor events associated with these registers:

**Table 58. Device Monitor Events**

Register	Function	Address specification; DMA channels; IRQs
PM50	Access to the primary or secondary floppy disk controllers.	I/O space 3F0h-3F5h, 3F7h, or 370h-375h, 377h fixed; DMA channel 2 in PM50
PM54 <sup>1</sup>	Access to the parallel ports.	See PM54; one of DMA[3:0]; see note.
PM58	Access to serial port COMA. (modem)	See PM58
PM5C	Access to serial port COMB. (IR)	See PM5C
PM60	Access to the audio hardware.	See PM60; any or all of DMA[7:5, 3:0]
PM64	User Interface: access to the video adapter; access to the legacy keyboard and PS/2 mouse ports; PCI bus utilization	Memory space 0A0000h-0BFFFFh fixed; I/O space 3B0h-3DFh, 60h, 64h; IRQ1, IRQ12
PM68	PIO access to any IDE drives.	I/O address space specified by DevB:1xXX
PM6C	Access to CARDBUS 0.	Address in space DevB:3xB4, DevB:3xB8,
PM70	Access to CARDBUS 1.	DevB:3xBC, DevB:3xC0
PM74	Timer.	None.
PM78	Access to programmable I/O range monitor 1.	Address in space DevB:3xC4, DevB:3xC8,
PM7C	Access to programmable I/O range monitor 2.	DevB:3xCC
PM80	Access to programmable I/O range monitor 3.	
PM84	Access to programmable I/O range monitor 4.	
PM88	Access to programmable memory/config range monitor 1.	Address in space DevB:3xD0, DevB:3xD4,
PM8C	Access to programmable memory/config range monitor 2.	DevB:3xD8
<b>Notes:</b>		
1. PM54 can alternately be used as an inactivity timer for PCI bus master activity based on REQ_L[6:0], PREQ_L, and internal master requests.		

The following provides the field definitions for bits[19:0] common to each of the sixteen 32-bit registers from PM50 to PM8C. See PM50-PM8C bits[31:20], below, for the unique bit definitions.

Default: 0000h (for each register).

Attribute: See below.

Offset: 8Fh-40h (four bytes for each register).

Bits	Description															
31:20	See PM50-PM8C below															
19	Reserved.															
18	<b>TMR_EN. Re-trigger timer enable.</b> Read-write. 1=Enable the re-trigger timer to decrement and to set the corresponding bit in the device monitor status register (PMA0) to generate an SMI or SCI interrupt. 0=Disable. If the timer is enabled and decrements past zero, then the PMA0 status bit is set and the timer stops (at zero). Also, whenever a High is written to TMR_EN, the corresponding re-trigger timer is loaded with its reload value.															
17	<b>SIT_RLD. System inactivity timer reload on device monitor event.</b> Read-write. 1=Enable system inactivity timer (PM98) to be reloaded by associated device monitor events; the event (not the associated STS bit) causes the SIT to be reloaded. 0=SIT not reloaded by associated device monitor event.															
16	Reserved.															
15:14	<b>CLKSRC. Clock source.</b> Read-write. Specifies the clock to the re-trigger timer per the following table. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>CLKSRC</th> <th>Clock period</th> <th>Maximum time (clock times 128)</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>1 millisecond</td> <td>128 milliseconds</td> </tr> <tr> <td>01b</td> <td>32 milliseconds</td> <td>4.1 seconds</td> </tr> <tr> <td>10b</td> <td>1 second</td> <td>128 seconds = 2.13 minutes</td> </tr> <tr> <td>11b</td> <td>64 seconds</td> <td>136.5 minutes = 2.28 hours</td> </tr> </tbody> </table>	CLKSRC	Clock period	Maximum time (clock times 128)	00b	1 millisecond	128 milliseconds	01b	32 milliseconds	4.1 seconds	10b	1 second	128 seconds = 2.13 minutes	11b	64 seconds	136.5 minutes = 2.28 hours
CLKSRC	Clock period	Maximum time (clock times 128)														
00b	1 millisecond	128 milliseconds														
01b	32 milliseconds	4.1 seconds														
10b	1 second	128 seconds = 2.13 minutes														
11b	64 seconds	136.5 minutes = 2.28 hours														
13:7	<b>CURCOUNT. Re-trigger timer current count value.</b> Read-only.															
6:0	<b>RELOAD. Re-trigger timer reload value.</b> Read-write. Device monitor events cause the re-trigger timer to be loaded with the state of this register. Also, writes to this field cause the re-trigger counter CURCOUNT to be updated.															

### Floppy Disk Controller Device Monitor Unique Controls

PM50 Bits[31:20]

See PM[8C:50], above, for bits[19:0].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:23	Reserved.
22	<b>FDCDEC_EN. Floppy disk controller decode enable.</b> 1=Enable accesses to floppy disk controller address range specified by PM50[FDCDEC_SEL] to generate a device monitor event. 0=Disable.
21	<b>FDCDEC_SEL. Floppy disk controller decode select.</b> This selects the floppy disk controller I/O address range for the FDC device monitor events. 1=Secondary FDC Address (370h-375h, 377h). 0=Primary FDC Address (3F0h-3F5h, 3F7h).
20	<b>FDCDMA_EN. Floppy disk controller DMA enable.</b> 1=Enable DMA channel 2 to generate the FDC device monitor event. 0=Disable.

**Parallel Port Device Monitor Unique Controls****PM54 Bits[31:20]**

See PM[8C:50], above, for bits[19:0].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:27	Reserved.
26	<b>BMIT_EN. Bus master inactivity timer enable.</b> 1=Re-trigger timer associated with PM54 is reloaded whenever a master access is indicated by REQ_L[6:0], PREQ_L, or internal masters. 0=Re-trigger timer associated with PM54 is reloaded by parallel port device monitor events specified by LPTDEC_EN and LPTDMA_EN.
25	<b>LPTDEC_EN. LPT Port Monitoring enable.</b> 1=Enable accesses to the I/O address range selected by PM54[LPTDEC_SEL] to generate a parallel port device monitor event.
24:23	<b>LPTDEC_SEL. LPT controller decode select.</b> Selects the I/O address range used for parallel port device monitor events. Bits[24:23]      LPT Decode 00                    3BCh–3BFh, 7BCh–7BEh 01                    378h–37Fh, 778h–77Ah 10                    278h–27Fh, 678h–67Ah 11                    Reserved
22	<b>LPTDMA_EN. LPT DMA enable.</b> 1=Enable the DMA channel selected by PM54[LPTDMA_SEL] to generate a parallel port device monitor event.
21:20	<b>LPTDMA_SEL. LPT DMA select.</b> Selects the DMA channel used for the parallel port device monitor event. Bits[21:20]      DMA channel 00                    DMA channel 0 01                    DMA channel 1 10                    DMA channel 3 11                    Reserved

**Serial Port A Device Monitor Unique Controls****PM58 Bits[31:20]**

See PM[8C:50], above, for bits[19:0].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:24	Reserved.
23	<b>CMADEC_EN. Serial port A monitor enable.</b> 1=Enable accesses to the I/O address range selected by PM58[CMADEC_SEL] to generate a serial port A device monitor event.
22:20	<b>CMADEC_SEL. Serial port A decode select.</b> Selects the I/O address range used for serial port A device monitor events.
Bits[22:20]	Serial A Decode
000	3F8h–3FFh (COM1)
001	2F8h–2FFh (COM2)
010	220h–227h
011	228h–22Fh
Bits[22:20]	Serial A Decode
100	238h–23Fh
101	2E8h–2EFh (COM4)
110	338h–33Fh
111	3E8h–3EFh (COM3)

**Serial Port B and Audio Device Monitor Unique Controls****PM5C Bits[31:20]**

See PM[8C:50], above, for bits[19:0]. Note: bits[31:24] of this register apply to the audio device monitor controls at PM60.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:29	Reserved.
28	<b>MSS_EN. Microsoft sound system decode enable.</b> 1=Enable accesses to the I/O address range selected by PM5C[MSS_SEL] to generate an audio device monitor event. Note: this bit applies to the audio device monitor, PM60.
27:26	<b>MSS_SEL. Microsoft sound system decode select.</b> Selects the MSS I/O address range used for audio device monitor events. Note: this bit applies to the audio device monitor, PM60.
Bits[4:3]	MSS Decode
00	530h–537h
01	604h–60Bh
10	E80h–E87h
11	F40h–F47h

Bits	Description (Continued)			
25	<b>GAME_EN. Game port enable.</b> 1=Enable accesses to the I/O address range 200h-207h to generate audio device monitor events. Note: this bit applies to the audio device monitor, PM60.			
24	<b>MIDI_EN. MIDI enable.</b> 1=Enable accesses to the I/O address range selected by PM60[MIDI_SEL] to generate an audio device monitor event. Note: this bit applies to the audio device monitor, PM60.			
23	<b>CMBDEC_EN. Serial port B monitor enable.</b> 1=Enable accesses to the I/O address range selected by PM5C[CMBDEC_SEL] to generate a serial port B device monitor event.			
22:20	<b>CMBDEC_SEL. Serial port B decode select.</b> Selects the I/O address range used for serial port B device monitor events.			
	Bits[22:20]	Serial B Decode	Bits[22:20]	Serial B Decode
	000	3F8h–3FFh (COM1)	100	238h–23Fh
	001	2F8h–2FFh (COM2)	101	2E8h–2EFh (COM4)
	010	220h–227h	110	338h–33Fh
	011	228h–22Fh	111	3E8h–3EFh (COM3)

**Audio Device Monitor Unique Controls****PM60 Bits[31:20]**

See PM[8C:50], above, for bits[19:0]. Note: PM5C[31:24] contains some control bits for the audio device monitors as well as PM60.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description	
31:30	<b>MIDI_SEL. MIDI decode select.</b> Selects the MIDI I/O range used for audio device monitor events (the enable for this range is PM5C[MIDI_EN]).	
	Bits [31:30]	MIDI Decode
	00	300h–303h
	01	310h–313h
	10	320h–323h
	11	330h–333h

Bits	Description (Continued)
29	Reserved.
28	<b>SB_EN. Sound Blaster 8/16 decode enable.</b> 1=Enable accesses to the I/O address range selected by the SB_SEL field and to 388h-38Bh to generate audio device monitor events.
27:26	<b>SB_SEL. Sound Blaster decode select.</b> Selects the Sound Blaster I/O address range used for audio device monitor events. Bits [27:26]      SB_SEL Decode 00                    220h–233h 01                    240h–253h 10                    260h–273h 11                    280h–293h
25:20	<b>ADMA_EN. Audio DMA enable.</b> Each of these bits specifies whether a DMA channel is enabled for audio device monitor events. Bit[20] specifies DMA channel 0; bit[21] specifies DMA channel 1; bit[22] specifies DMA channel 3; bit[23] specifies DMA channel 5; bit[24] specifies DMA channel 6; bit[25] specifies DMA channel 7. 1=Enable the corresponding DMA request signal to generate an audio device monitor event. 0=Disable.

**User Interface Device Monitor Unique Controls**

**PM64 Bits[31:20]**

See PM[8C:50], above, for bits[19:0].

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:25	Reserved.
24	<b>GRAB_EN. Graphics A/B segment memory enable.</b> 1=Enable accesses to VGA frame buffer memory address ranges A_0000h through B_FFFFh to generate user interface device monitor events.
23	<b>GRIO_EN. Graphics I/O enable.</b> 1=Enable accesses to VGA I/O addresses 3B0h-3DFh to generate user interface device monitor events.
22	<b>KBC_EN. Keyboard enable.</b> 1=Enable accesses to the keyboard controller I/O address range (ports 60h and 64h) to generate user interface device monitor events.
21	<b>IRQ1_EN. IRQ1 enable.</b> 1=Enable IRQ1 (keyboard activity) to generate a user interface device monitor event.
20	<b>IRQ12_EN. IRQ12 enable.</b> 1=Enable IRQ12 (mouse activity) to generate user interface device monitor events.

**General Purpose Timer****PM94**

The purpose of this timer is to generate an interrupt at a specified time in the future.

Default: 0000h.

Attribute: Read-write.

Bits	Description									
15:13	Reserved.									
12	<b>GPT_CLK. General purpose timer clock source.</b> Specifies the clock to the general purpose timer as follows:									
	<table border="0"> <tr> <td>GPT_CLK</td> <td>Clock period</td> <td>Maximum time (clock times 4096)</td> </tr> <tr> <td>0</td> <td>30.52 microseconds</td> <td>125 milliseconds</td> </tr> <tr> <td>1</td> <td>32 milliseconds</td> <td>21.8 minutes</td> </tr> </table>	GPT_CLK	Clock period	Maximum time (clock times 4096)	0	30.52 microseconds	125 milliseconds	1	32 milliseconds	21.8 minutes
GPT_CLK	Clock period	Maximum time (clock times 4096)								
0	30.52 microseconds	125 milliseconds								
1	32 milliseconds	21.8 minutes								
11:0	<b>GPT_CNT. General purpose timer count value.</b> Reads provides the current state of the general purpose timer. Writes to this field load a value in the timer. Once loaded—if either PMA4[GPT_EN] or PMA8[GPTSMI_EN] is set High—the timer counts down to zero and stops. On the clock after GPT_CNT reaches zero, PMA0[GPT_STS] is set High. If PMA4[GPT_EN] and PMA8[GPTSMI_EN] are Low, then the general purpose timer does not count and PMA0[GPT_STS] cannot be set High.									

**System Inactivity Timer Register****PM98**

The system inactivity timer is a counter that may be reloaded by device monitor events as specified by PM[8C:50][SIT\_RLD] or by events specified by PMAC. When it reaches zero, it sets PM20[SIT\_STS] and may be enabled to generate interrupts.

Default: 0000 0000h.

Attribute: See below.

Bits	Description															
31:18	Reserved.															
17:16	<b>CLKSRC. Clock source.</b> Read-write. Specifies the clock to the system inactivity timer per the following table.															
	<table border="0"> <tr> <td>CLKSRC</td> <td>Clock period</td> <td>Maximum time (clock times 128)</td> </tr> <tr> <td>00b</td> <td>1 millisecond</td> <td>128 milliseconds</td> </tr> <tr> <td>01b</td> <td>32 milliseconds</td> <td>4.1 seconds</td> </tr> <tr> <td>10b</td> <td>1 second</td> <td>128 seconds = 2.13 minutes</td> </tr> <tr> <td>11b</td> <td>64 seconds</td> <td>136.5 minutes = 2.28 hours</td> </tr> </table>	CLKSRC	Clock period	Maximum time (clock times 128)	00b	1 millisecond	128 milliseconds	01b	32 milliseconds	4.1 seconds	10b	1 second	128 seconds = 2.13 minutes	11b	64 seconds	136.5 minutes = 2.28 hours
CLKSRC	Clock period	Maximum time (clock times 128)														
00b	1 millisecond	128 milliseconds														
01b	32 milliseconds	4.1 seconds														
10b	1 second	128 seconds = 2.13 minutes														
11b	64 seconds	136.5 minutes = 2.28 hours														
15:8	<b>CURCOUNT.</b> Read-only. System inactivity timer current count value.															
7:0	<b>RELOAD. System inactivity timer reload value.</b> Read-write. Writes to this field cause the system inactivity counter to be reloaded with the value written.															

**Device Monitor Status Register****PMA0**

Each of status bits[19:0] is set by a device monitor event (an access to an I/O or memory address range, an IRQ or a DMA request as specified by PM[8C:50] or a re-trigger timer time out) and bit[20] is set by the general purpose timer (specified by PM94). If any of these events occur, then the status bit is set. If any of these status bits are High and the corresponding enable bit is High (PMA4 and PMA8), then an interrupt is generated.

Default: 0000 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
31:21	Reserved.
20	<b>GPT_STS. General purpose timer status.</b> This bit is set High by the hardware on the clock after PM94[GPT_CNT] reaches zero. However, if PMA4[GPT_EN] and PMA8[GPTSMI_EN] are both Low, then the PMA0[GPT_STS] is not set High.
19	<b>PMM2_DM_STS. Programmable memory range monitor 2 device monitor status.</b>
18	<b>PMM1_DM_STS. Programmable memory range monitor 1 device monitor status.</b>
17	<b>PRM4_DM_STS. Programmable range monitor 4 device monitor status.</b>
16	<b>PRM3_DM_STS. Programmable range monitor 3 device monitor status.</b>
15	<b>PRM2_DM_STS. Programmable range monitor 2 device monitor status.</b>
14	<b>PRM1_DM_STS. Programmable range monitor 1 device monitor status.</b>
13	<b>TIM_DM_STS. Timer timeout device monitor status.</b>
12	<b>CARDBUS1_DM_STS. CARDBUS1 access device monitor status.</b>
11	<b>CARDBUS0_DM_STS. CARDBUS0 access device monitor status.</b>
10	Reserved.
9	<b>USRINT_DM_STS. User interface device monitor status.</b>
8	<b>AUD_DM_STS. Audio functions device monitor status.</b>
7	<b>CMB_DM_STS. Serial port B (COM B) device monitor status.</b>
6	<b>CMA_DM_STS. Serial port A (COM A) device monitor status.</b>
5	<b>LPT_DM_STS. Parallel port (LPT) device monitor status.</b>
4	<b>FDD_DM_STS. Floppy disk drive device monitor status.</b>
3	<b>DSS_DM_STS. IDE secondary slave port device monitor status.</b>
2	<b>DSM_DM_STS. IDE secondary master port device monitor status.</b>
1	<b>DPS_DM_STS. IDE primary slave port device monitor status.</b>
0	<b>DPM_DM_STS. IDE primary master port device monitor status.</b>



**Device Monitor ACPI Interrupt Enable Register****PMA4**

For each of these bits: 1=Enable either an SCI or an SMI interrupt based on the state of PM04[SCI\_EN] if the corresponding status bit in PMA0 is set.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:21	Reserved.
20	<b>GPT_EN.</b> General purpose timer ACPI interrupt enable.
19	<b>PMM2_DM_EN.</b> Programmable memory range monitor 2 device monitor ACPI interrupt enable.
18	<b>PMM1_DM_EN.</b> Programmable memory range monitor 1 device monitor ACPI interrupt enable.
17	<b>PRM4_DM_EN.</b> Programmable range monitor 4 device monitor ACPI interrupt enable.
16	<b>PRM3_DM_EN.</b> Programmable range monitor 3 device monitor ACPI interrupt enable.
15	<b>PRM2_DM_EN.</b> Programmable range monitor 2 device monitor ACPI interrupt enable.
14	<b>PRM1_DM_EN.</b> Programmable range monitor 1 device monitor ACPI interrupt enable.
13	<b>TIM_DM_EN.</b> Timer timeout device monitor ACPI interrupt enable.
12	<b>CARDBUS1_DM_EN.</b> CARDBUS1 access device monitor ACPI interrupt enable.
11	<b>CARDBUS0_DM_EN.</b> CARDBUS0 access device monitor ACPI interrupt enable.
10	Reserved.
9	<b>USRINT_DM_EN.</b> User interface device monitor ACPI interrupt enable.
8	<b>AUD_DM_EN.</b> Audio functions device monitor ACPI interrupt enable.
7	<b>CMB_DM_EN.</b> Serial port B (COM B) device monitor ACPI interrupt enable.
6	<b>CMA_DM_EN.</b> Serial port A (COM A) device monitor ACPI interrupt enable.
5	<b>LPT_DM_EN.</b> Parallel port (LPT) device monitor ACPI interrupt enable.
4	<b>FDD_DM_EN.</b> Floppy disk drive device monitor ACPI interrupt enable.
3	<b>DSS_DM_EN.</b> IDE secondary slave port device monitor ACPI interrupt enable.
2	<b>DSM_DM_EN.</b> IDE secondary master port device monitor ACPI interrupt enable.
1	<b>DPS_DM_EN.</b> IDE primary slave port device monitor ACPI interrupt enable.
0	<b>DPM_DM_EN.</b> IDE primary master port device monitor ACPI interrupt enable.

**Device Monitor SMI Interrupt Enable Register****PMA8**

For each of these bits: 1=Enable SMI interrupt if the corresponding status bit in PMA0 is set.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:21	Reserved.
20	<b>GPTSMI_EN.</b> General purpose timer SMI enable.
19	<b>PMM2_DMSMI_EN.</b> Programmable memory range monitor 2 device monitor SMI enable.
18	<b>PMM1_DMSMI_EN.</b> Programmable memory range monitor 1 device monitor SMI enable.
17	<b>PRM4_DMSMI_EN.</b> Programmable range monitor 4 device monitor SMI enable.
16	<b>PRM3_DMSMI_EN.</b> Programmable range monitor 3 device monitor SMI enable.
15	<b>PRM2_DMSMI_EN.</b> Programmable range monitor 2 device monitor SMI enable.
14	<b>PRM1_DMSMI_EN.</b> Programmable range monitor 1 device monitor SMI enable.
13	<b>TIM_DMSMI_EN.</b> Timer timeout device monitor SMI enable.

Bits	Description (Continued)
12	<b>CARDBUS1_DMSMI_EN.</b> CARDBUS1 access device monitor SMI enable.
11	<b>CARDBUS0_DMSMI_EN.</b> CARDBUS0 access device monitor SMI enable.
10	Reserved.
9	<b>USRINT_DMSMI_EN.</b> User interface device monitor SMI enable.
8	<b>AUD_DMSMI_EN.</b> Audio functions device monitor SMI enable.
7	<b>CMB_DMSMI_EN.</b> Serial port B (COM B) device monitor SMI enable.
6	<b>CMA_DMSMI_EN.</b> Serial port A (COM A) device monitor SMI enable.
5	<b>LPT_DMSMI_EN.</b> Parallel port (LPT) device monitor SMI enable.
4	<b>FDD_DMSMI_EN.</b> Floppy disk drive device monitor SMI enable.
3	<b>DSS_DMSMI_EN.</b> IDE secondary slave port device monitor SMI enable.
2	<b>DSM_DMSMI_EN.</b> IDE secondary master port device monitor SMI enable.
1	<b>DPS_DMSMI_EN.</b> IDE primary slave port device monitor SMI enable.
0	<b>DPM_DMSMI_EN.</b> IDE primary master port device monitor SMI enable.

### IRQ Reload Enable For System Inactivity Timer Register

PMAC

Each of these bits enable signals to trigger a reload of the system inactivity timer (SIT). In the case of the IRQRL signals, they only cause the SIT to be reloaded whenever they toggle.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:20	Reserved.
19	<b>BMREQRL.</b> 1=Any condition that sets PM00[BM_STS] causes a reload of the system inactivity timer.
18	<b>EXTSMIRL.</b> 1=The status bit associated with the assertion of the EXTSMI_L pin (PM20[EXSMI_STS]) causes a reload of the system inactivity timer. Note: as long as the status bit is set, the system inactivity timer is held in its reload value and it does not decrement.
17	<b>INITRL.</b> <b>INIT reload for the system inactivity timer.</b> 1=Enables INIT interrupt to reload the system inactivity timer.
16	<b>NMIRL.</b> <b>NMI reload for the system inactivity timer.</b> 1=Enables NMI interrupt to reload the system inactivity timer. 0=NMI does affect the system inactivity timer.
15:0	<b>IRQRL.</b> <b>IRQs reload the system inactivity timer.</b> Each of these bits corresponds to an IRQ (e.g., bit[12] corresponds to IRQ12). The exception to this is bit[2], which corresponds to the interrupt output of the PIC or IOAPIC (see DevB:0x4B[APICEN]). For each bit: 1=Enable the corresponding IRQ signal to cause the system inactivity timer to reload whenever it changes state (High to Low or Low to High). 0=IRQ signal does affect the system inactivity timer.

**GPIO Pin Interrupt Status Register (ACPI GPE1\_STS)****PMB0**

Each of these status bits is driven by the output of the input circuit associated with the GPIO pins. Bit[0] corresponds to GPIO 0; bit[1] corresponds to GPIO1, and so forth. The latch associated with each GPIO input circuit is cleared when the corresponding bit in this register is written with a 1; writing a 0 has no effect.

GPIO\_STS[25, 24, 23, 22, 18, 14, 3, 2, 1, 0] reside in the VDD\_COREX power plane and can be used in conjunction with PMB4 as resume events from any sleep state.

Default: 0000 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
31:0	GPIO_STS. GPIO IRQ status bits.

**GPIO Pin ACPI Interrupt Enable Register (ACPI GPE1\_EN)****PMB4**

For each of these bits: 1=Enable either an SCI or SMI interrupt based on the state of PM04[SCI\_EN] if the corresponding status bit in PMB0 is set.

GPIO\_EN[25, 24, 23, 22, 18, 14, 3, 2, 1, 0] reside in the VDD\_COREX power plane and can be used in conjunction with PMB0 as resume events from any sleep state.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:0	GPIO_EN. GPIO SCI/SMI enable bits.

**GPIO Pin SMI Interrupt Enable Register****PMB8**

For each of these bits: 1=Enable an SMI interrupt if the corresponding status bit in PMB0 is set.

Default: 0000 0000h.

Attribute: Read-write.

Bits	Description
31:0	GPIO SMI enable bits.

**GPIO Output Clock 0 and 1 Register****PMBC**

This register specifies the High time and the Low time for the GPIO output clocks. These clocks can be selected as the output for any of the GPIO pins. These output clocks consist of a 7-bit down counter that is alternately loaded with the High time and the Low time. The clock for the counters is selected by CLK[1,0]BASE.

Default: FFFF FFFFh.

Attribute: Read-write.

Bits	Description
31:30	<b>CLK1BASE.</b> See bits[15:14].
29:23	<b>CLK1HI.</b> See bits[13:7].
22:16	<b>CLK1LO.</b> See bits[6:0].
15:14	<b>CLK0BASE. GPIO output clock timer base.</b> Specifies the clock for the counter that generates the GPIO output clock. 00b=250 microseconds; 01b=2 milliseconds; 10b=16 milliseconds; 11b=128 milliseconds. CLK0BASE specifies the clock for GPIO output clock 0 and CLK1BASE specifies the clock for GPIO output clock 1.
13:7	<b>CLK0HI. GPIO output clock High time.</b> Specifies the High time for the GPIO output clocks in increments of clock specified by CLK[1,0]BASE (if the base is 16 milliseconds, then 0 specifies 16 milliseconds, 1 specifies 32 milliseconds, etc.). CLK0HI specifies the High time for GPIO output clock 0 and CLK1HI specifies the High time for GPIO output clock 1.
6:0	<b>CLK0LO. GPIO output clock Low time.</b> Specifies the Low time for the GPIO output clocks in increments of the clock specified by CLK[1,0]BASE (if the base is 16 milliseconds, then 0 specifies 16 milliseconds, 1 specifies 32 milliseconds, etc.). CLK0LO specifies the Low time for GPIO output clock 0 and CLK1LO specifies the Low time for GPIO output clock 1.

**General Purpose I/O Pins GPIO[31:0] Select Registers****PM[DF:C0]**

See Section 3.7.5 on page 66 for details about the GPIO hardware.

Usage note: to set a GPIO pin as a software-controlled output, its corresponding GPIO register should be written with the value 04h for a Low and the value 05h for a High.

Default: See the MODE field definition.

Attribute: See below.

Offset: DFh-C0h (one single-byte register for each GPIO pin).

Bits	Description
7	Reserved.
6	<b>LTCH_STS. GPIO latch status.</b> Read; set by hardware; write 1 to clear. This provides the current state of the latch associated with the input path for the GPIO pin that corresponds to the register.

Bits	Description (Continued)
5	<b>RTIN. Real time in.</b> Read-only. This provides the current, not-inverted state of the pad for the GPIO pin that corresponds to the register.
4	<b>DEBOUNCE. Debounce the input signal.</b> Read-write. 1=The input signal is required to be held active without glitches for 12 to 16 milliseconds before being allowed to set the GPIO latch or being capable of being passed along to the circuitry being controlled by the output of the input path.
3:2	<b>MODE[1:0]. Pin mode select.</b> Read-write. These specify the GPIO pin mode as follows: MODE[1:0]      GPIO pin mode 00b              GPIO input 01b              GPIO output 1Xb              Pin specified to as perform alternate function (non-GPIO mode).
1:0	<b>X[1:0].</b> Read-write. If the GPIO function is not used by the pin (if the pin is programmed as an alternative to the GPIO function, e.g., CPUSLEEP_L), then this field does not matter. If the GPIO function is selected, then based on whether the GPIO pin is a input or an output (selected by MODE also), this register has the meanings shown in the table below.

**Table 59. GPIO X[1:0] Bit Decoding**

I/O Mode	Bits	Name	Function
Input	X0	ACTIVEHI	0=The pin is active Low and the signal is inverted as it is brought into the input path. 1=The pin is active High and therefore not inverted as it is brought through the input path. Note: the IRQ1, IRQ8, IRQ12 and PNPIRQ[2:0] signals that pass through the GPIO input path are inverted in the interrupt routing logic before being passed to the PIC; therefore, for an active High IRQx or PNPIRQ pin, ACTIVEHI should be set High for active-Low (level triggered) interrupts and Low for active-High (edge triggered) interrupts.
Input	X1	LATCH	0=The latched version of the signal is not selected. 1=The latched version is selected.
Output	X[1:0]=0h		Output is forced Low.
Output	X[1:0]=1h		Output is forced High.
Output	X[1:0]=2h		GPIO output clock 0 (specified by PMBC[15:0]).
Output	X[1:0]=3h		GPIO output clock 1 (specified by PMBC[31:16]).

The table below shows the default states for these registers and the pin definitions base on the state of MODE[1:0]. The “Default” field shows the defaults for all the bits in the register. The “Mode” and “X[1:0]” field show the value required in order to enable the function specified in the “Signal Name” column (“x” specifies that the bit does not matter). The “Input Path” field shows how the alternate function signal is mapped into internal logic; “GPIO” specifies that the signal passes through the GPIO input path (and can therefore use the polarity, latch and debounce controls from the GPIO circuit); “Direct” specifies that the signal comes directly from the pad; “N/A” specifies that it is an output signal.

**Table 60. GPIO Register Default States**

GPIO Name	Control Reg	Signal Name	Default	Mode	Input Path	Notes
GPIO0	PMC0	ACAV	08h (ACAV input)	1Xb	GPIO	2
GPIO1	PMC1	AGPSTOP_L	05h (GPIO1 output, High)	1Xb	N/A	2
GPIO2	PMC2	BATLOW_L	08h (BATLOW_L input)	1Xb	GPIO	2
GPIO3	PMC3	C32KHZ	08h (C32KHZ output)	1Xb	N/A	2
GPIO4	PMC4	GPIO4	04h (GPIO4 output, Low)	1Xb	N/A	4
GPIO5	PMC5	CLKRUN_L	04h (GPIO5 output, Low)	1Xb	Direct	3
GPIO6	PMC6	CPUSLEEP_L	05h (GPIO6 output, High)	1Xb	N/A	
GPIO7	PMC7	CPUSTOP_L	05h (GPIO7 output, High)	1Xb	N/A	
GPIO8	PMC8	GPIO8	05h (GPIO8 output, High)	1Xb	N/A	
GPIO9	PMC9	FANCON1	08h (FANCON1 output)	1Xb	N/A	
GPIO10	PMCA	FANRPM	08h (FANRPM input)	1Xb	direct	
GPIO11	PMCB	INTIRQ8_L	05h (GPIO11 output, High)	1Xb	N/A	
GPIO12	PMCC	IRQ1	09h (IRQ1 input)	1Xb	GPIO	
GPIO13	PMCD	IRQ6	09h (IRQ6 input)	1Xb	GPIO	
GPIO14	PMCE	GPIO14	00h (GPIO14 input)	0Xb	N/A	2
GPIO15	PMCF	IRQ12	09h (IRQ12 input)	1Xb	GPIO	
GPIO16	PMD0	GPIO16	00h (GPIO 16 input)	0Xb	N/A	
GPIO17	PMD1	GPIO17	00h (GPIO 17 input)	0Xb	N/A	
GPIO18	PMD2	LID	00h (GPIO 18 input)	1Xb	Direct	2
GPIO19	PMD3	PNPIRQ0	09h (PNPIRQ0 input)	1Xb	GPIO	
GPIO20	PMD4	PNPIRQ1	09h (PNPIRQ1 input)	1Xb	GPIO	
GPIO21	PMD5	PNPIRQ2	09h (PNPIRQ2 input)	1Xb	GPIO	
GPIO22	PMD6	SMBALERT0_L	08h (SMBALERT0_L input)	1Xb	Direct	2
GPIO23	PMD7	SLPBTN_L	08h (SLPBTN_L input)	1Xb	GPIO	2
GPIO24	PMD8	SMBALERT1_L	08h (SMBALERT1_L input)	1Xb	Direct	2
GPIO25	PMD9	SUSPEND_L	08h (SUSPEND_L output)	1Xb	N/A	2
GPIO26	PMDA	GPIO26	00h (GPIO 26 input)	0Xb	N/A	
GPIO27	PMDB	GPIO27	00h (GPIO 27 input)	0Xb	N/A	
GPIO28	PMDC	GPIO28	00h (GPIO28 input)	0Xb	N/A	1
GPIO29	PMDD	GPIO29	00h (GPIO29 input)	0Xb	N/A	1
GPIO30	PMDE	GPIO30	00h (GPIO30 input)	0Xb	N/A	1
GPIO31	PMDF	GPIO31	00h (GPIO31 input)	0Xb	N/A	1

**Notes:**

1. The signals from GPIO[31:28] go directly to the IOAPIC to drive the interrupt request inputs to some of the redirection-table entries. The polarity of these signals at the IOAPIC is as seen at the external pins, the polarity is not altered by programming of DevB:3x[DF:DC]. These signals, from the GPIO pins to the IOAPIC, are not ever disabled, even if the pin's function is other-than GPIO. Also, see DevB:0x4B[MPIRQ] for details on how GPIO[31:28] may be mapped to PIRQ[A,B,C,D]\_L.
2. These pins, corresponding registers, and GPIO logic reside on the VDD\_IOX and VDD\_COREX power planes, respectively, and are reset by RST\_SOFT. All the other GPIO pins reside on the main power supply.
3. When the CLKRUN\_L function is enabled, CLKRUN\_L functionality is automatically enabled. To disable CLKRUN\_L functionally, this pin should be programmed as a GPIO.

4. When programmed for alternate function, the status of WDT00[WFIR] can be observed externally.

**SMBus Global Status Register****PME0**

Default: 0000h.

Attribute: Read; set by hardware; write 1 to clear.

Bits	Description
15:12	Reserved.
11	<b>SMB_BSY. SMBus busy.</b> Read-only. 1=The SMBus is currently busy with a cycle generated by either the host or another SMBus master.
10	<b>SMBA_STS. SMBALERT_L interrupt status.</b> This bit is set High by the hardware when SMBALERT_L is asserted Low. This bit is not set unless the SMBALERT_L function is selected by PMD6. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[SMBA_EN].
9	<b>HSLV_STS. Host-as-slave address match status.</b> This bit is set High by the hardware when an SMBus master (including the host controller) generates an SMBus write cycle with a 7-bit address that matches the one specified by PМЕЕ. This bit is not set until the end of the acknowledge bit after the last byte is transferred; however, if a time out occurs after the address match occurs and before last acknowledge, then this bit is not set. This may trigger an SMI or SCI interrupt if enabled to do so by PME2[HSLV_EN]. This bit resides on the VDD_COREX power plane.
8	<b>SNP_STS. Snoop address match status.</b> This bit is set High by the hardware when an SMBus master (including the host controller) generates an SMBus cycle with a 7-bit address that matches the one specified by PМЕF. This bit is not set until the end of the acknowledge bit after the last byte is transferred; however, if a time out occurs after the address match occurs and before the last acknowledge, then this bit is not set. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[SNP_EN]. This bit resides on the VDD_COREX power plane.
7:6	Reserved.
5	<b>TO_STS. Time out error status.</b> This bit is set High by the hardware when a slave device forces a time out by holding the SMBUSC pin Low for more than 25 milliseconds. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[HCYC_EN].
4	<b>HCYC_STS. Host cycle complete status.</b> This bit is set High by the hardware when a host cycle completes successfully. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[HCYC_EN]. Note: it is illegal for SW to attempt to clear this bit when it is not yet set.
3	<b>HST_BSY. Host controller busy.</b> Read-only. 1=The SMBus host controller is busy with a cycle.
2	<b>PRERR_STS. Protocol error status.</b> This bit is set High by the hardware when a slave device does not generate an acknowledge at the appropriate time during a host SMBus cycle. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[HCYC_EN].
1	<b>COL_STS. Host collision status.</b> This bit is set High by the hardware when a host transfer is initiated while the SMBus is busy. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[HCYC_EN]. Note: If the SMBus is detected busy prior to the command to initiate a host transfer, then the IC waits for the initial transaction to complete before executing the host transfer and this bit is not set; this bit may only be set when host transactions start at approximately the same time as other transactions.
0	<b>ABRT_STS. Host transfer abort status.</b> This bit is set High by the hardware after a host transfer is aborted by PME2[ABORT] command. This bit may trigger an SMI or SCI interrupt if enabled to do so by PME2[HCYC_EN].

**SMBus Global Control Register****PME2**

Most of these bits enable either an SCI or an SMI interrupt based on the state of PM04[SCI\_EN] if the corresponding status bit is set.

Default: 0000h.

Attribute: Read-write.

Bits	Description																								
15:11	Reserved.																								
10	<b>SMBA_EN. SMBALERT_L interrupt enable.</b> 1=Enables an SMI or SCI interrupt when PME0[SMBA_STS] is set High. This bit has no effect unless the SMBALERT_L function is selected by PMD6.																								
9	<b>HSLV_EN. Host-as-slave address match interrupt enable.</b> 1=Enables an SMI or SCI interrupt when PME0[HSLV_STS] is set High. This bit resides on the VDD_COREX power plane.																								
8	<b>SNP_EN. Snoop address match interrupt enable.</b> 1=Enables an SMI or SCI interrupt when PME0[SNP_STS] is set High. This bit resides on the VDD_COREX power plane.																								
7:6	Reserved.																								
5	<b>ABORT. Abort current host transfer command.</b> Write-only. 1=The SMBus logic generates a stop event on the SMBus pins as soon as possible (there may be a delay if the SMBus slave is generating zeros during a read cycle). After the stop event is generated, PME0[ABRT_STS] is set High.																								
4	<b>HCYC_EN. Host SMBus controller interrupt enable.</b> 1=The SMBus host controller status bits, PME0[TO_STS, HCYC_STS, PRERR_STS, COL_STS, ABRT_STS], are enabled to generate SMI or SCI interrupts.																								
3	<b>HOSTST. Host start command.</b> Write-only. 1=The SMBus host logic initiates the SMBus cycle specified by CYCTYPE. Writes to this field are ignored while PME0[HST_BSY] is active.																								
2:0	<p><b>CYCTYPE. Host-generated SMBus cycle type.</b> Writes to this field are ignored while PME0[HST_BSY] is active. This field specifies the type of SMBus cycle that is generated when it is initiated by the HOSTST command. Here is how it is decoded (for each of the registers, the slave address is specified by PME4[7:1] and “receive” or “read” versus “send” or “write” is specified by PME4[0]):</p> <table border="1"> <thead> <tr> <th>CYCTYPE</th> <th>SMBus Cycle Type</th> <th>Registers</th> </tr> </thead> <tbody> <tr> <td>000b</td> <td>Quick command</td> <td>Data bit in PME4[0]</td> </tr> <tr> <td>001b</td> <td>Receive or send byte</td> <td>Data in PME6[7:0]. If the address in PME4 is 0001_1001b and data received is 111_0XXXb, then another byte is received in PME6[15:8]; see also the SMBALERT description in Section 3.7.3 on page 65.</td> </tr> <tr> <td>010b</td> <td>Read or write byte</td> <td>Command in PME8; data in PME6[7:0]</td> </tr> <tr> <td>011b</td> <td>Read or write word</td> <td>Command in PME8; data in PME6[15:0]</td> </tr> <tr> <td>100b</td> <td>Process call</td> <td>Command in PME8; write data is placed in PME6[15:0]; then this data is replaced with the read data in the second half of the command</td> </tr> <tr> <td>101b</td> <td>Read or write block</td> <td>Command in PME8; count data in PME6[5:0]; block data in the PME9 FIFO</td> </tr> <tr> <td>11Xb</td> <td>Reserved</td> <td></td> </tr> </tbody> </table>	CYCTYPE	SMBus Cycle Type	Registers	000b	Quick command	Data bit in PME4[0]	001b	Receive or send byte	Data in PME6[7:0]. If the address in PME4 is 0001_1001b and data received is 111_0XXXb, then another byte is received in PME6[15:8]; see also the SMBALERT description in Section 3.7.3 on page 65.	010b	Read or write byte	Command in PME8; data in PME6[7:0]	011b	Read or write word	Command in PME8; data in PME6[15:0]	100b	Process call	Command in PME8; write data is placed in PME6[15:0]; then this data is replaced with the read data in the second half of the command	101b	Read or write block	Command in PME8; count data in PME6[5:0]; block data in the PME9 FIFO	11Xb	Reserved	
CYCTYPE	SMBus Cycle Type	Registers																							
000b	Quick command	Data bit in PME4[0]																							
001b	Receive or send byte	Data in PME6[7:0]. If the address in PME4 is 0001_1001b and data received is 111_0XXXb, then another byte is received in PME6[15:8]; see also the SMBALERT description in Section 3.7.3 on page 65.																							
010b	Read or write byte	Command in PME8; data in PME6[7:0]																							
011b	Read or write word	Command in PME8; data in PME6[15:0]																							
100b	Process call	Command in PME8; write data is placed in PME6[15:0]; then this data is replaced with the read data in the second half of the command																							
101b	Read or write block	Command in PME8; count data in PME6[5:0]; block data in the PME9 FIFO																							
11Xb	Reserved																								



**SMBus Host Address Register****PME4**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:8	<b>HST10BA. Host 10-bit address LSBs.</b> This field stores the second byte of the address, used in 10-bit SMBus host-as-master transfers. If HSTADDR == 1111_0XXb, then the cycle is specified to use 10-bit addressing. If HSTADDR is any other value, then HST10BA is not utilized.
7:1	<b>HSTADDR. Host cycle address.</b> This specifies the 7-bit address to the SMBus generated by the host (as a master) during SMBus cycles that are initiated by PME2[HOSTST].
0	<b>READCYC. Host read (High) write (Low) cycle.</b> 1=Specifies that the cycle generated by a write to PME2[HOSTST] is a read or receive command. 0=Cycle is a write or send command.

**SMBus Host Data Register****PME6**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:0	<b>HSTDATA. Host cycle data.</b> This register is written to by software to specify the data to be passed to the SMBus during write and send cycles. It is read by software to specify the data passed to host controller by the SMBus during read and receive cycles. Bit[0] specifies the data written or read during the quick command cycle. Bits[7:0] specify the data for byte read and write cycles, send byte cycles, and receive byte cycles. Bits[15:0] are used for word read and write cycles and process calls. Bits[5:0] are used to specify the count for block read and write cycles.

**SMBus Host Command Field Register****PME8**

Default: 00h.

Attribute: Read-write.

Bits	Description
7:0	<b>HSTCMD. Host cycle command.</b> This specifies the command field passed to the SMBus by the host controller during read byte, write byte, read word, write word, process call, block read, and block write cycles. Host cycles are initiated by PME2[HOSTST].

**SMBus Host Block Data FIFO Access Port****PME9**

Default: 00h.

Attribute: See below.

Bits	Description
7:0	<b>HSTFIFO. Host block read-write FIFO.</b> For block write commands, software loads 1 to 32 bytes into this port before sending them to the SMBus through the PME2[HOSTST] command. For block read commands, software reads 1 to 32 bytes from this port after the block read cycle is complete. If, during a block read or write, an error occurs, then the FIFO is flushed by the hardware. Read and write accesses to this port while the host is busy (PME0[HST_BSY]) are ignored.

**SMBus Host-As-Slave Data Register****PMEA**

Default: 0000h.

Attribute: Read-only.

Bits	Description
15:0	<b>HSLVDATA. Host-as-slave data.</b> When the logic detects that the current SMBus cycle is directed to the host's slave logic (because the address matches PMEE), then the data transmitted to the IC during the cycle is latched in this register. Also, if the address matches the snoop address in PMEF, then the cycle type is assumed to be a write word and the data is stored in this register. This register resides on the VDD_COREX power plane.

**SMBus Host-As-Slave Device Address Register****PMEC**

This register resides on the VDD\_COREX power plane.

Default: 0000h.

Attribute: Read-only.

Bits	Description
15:8	<b>HSLV10DA. Host-as-slave 10-bit device address LSBs.</b> This field stores the second byte of the device address used in 10-bit SMBus transfers to the host as a slave. If HSLVDA == 1111_0XXb, then the cycle is specified by the SMBus specification to transmit a 10-bit device address to the host-as-slave logic and the second byte of that device address is stored in this field. If HSLVDA is any other value, then HSLV10BA is not utilized.
7:1	<b>HSLVDA. Host-as-slave device address.</b> When the logic detects that the current SMBus cycle is directed to the host's slave logic (because the address matches PMEE), then the device address transmitted to the IC during the "command" phase of the cycle is latched in this register. Also, if the SMBus address matches the snoop address in PMEF, then the cycle type is assumed to be a write word and bits[7:1] of the command field for the cycle are placed in this field.
0	<b>SNPLSB. Snoop command LSB.</b> If the SMBus cycle address matches PMEF, then the cycle type is assumed to be a write word. The LSB of the command field for the cycle is placed in this bit (and the other 7 bits are placed in HSLVDA).

**SMBus Host-As-Slave Host Address Register****PMEE**

This register resides on the VDD\_COREX power plane.

Default: 10h.

Attribute: Read-write.

Bits	Description
7:1	<b>HSLVADDR. Host-as-slave address.</b> The IC compares the address generated by masters over the SMBus to this field to determine if there is a match (also, for a match to occur, the read-write bit is required to specify a write command). If a match occurs, then the cycle is assumed to be a write word command to the host, with the slave's device address transmitted during the normal command phase. The device address is captured in PMEC and the data is capture in PMEA for the cycle. After the cycle is complete, PME0[HSLV_STS] is set.
0	Reserved.

**SMBus Snoop Address Register****PMEF**

This register resides on the VDD\_COREX power plane.

Default: 10h.

Attribute: Read-write.

Bits	Description
7:1	<b>SNPADDR. Snoop address.</b> The IC compares the address generated by masters over the SMBus to this field to determine if there is a match (regardless as to whether it is a read or a write). If there is a match, then PME0[SNP_STS] is set High after the cycle completes. If the address specified here matches PMEE, then PME0[SNP_STS] is not set High.
0	Reserved.

**Random Number Register****PMF0**

PMF0 and PMF4 together support of the random number generator (RNG) function. When PMF4[RNGDONE] is 1, then the value in PMF0 has been updated. Reading PMF0 clears RNGDONE until the next valid random number is available in PMF0. New random numbers are generated approximately 128 microseconds after PMF0 is read. If PMF0 is read while RNGDONE = 0, then the value returned is all zeros.

Default: Not deterministic.

Attribute: Read-only.

Bits	Description
31:0	<b>RANDOM_NUM.</b> Random number.

**Random Number Status Register****PMF4**

Default: 0000 0001h (see note in bit 0).

Attribute: Read-only.

Bits	Description
31:1	Reserved.
0	<b>RNGDONE. Random number generator number generation process complete.</b> See PMF0. Note: this bit is Low after the trailing edge of RESET_L and only goes High after the first random number is valid in PMF0, less than 500 microseconds later.

**Fan Control Register****PMF8**

This register specifies the frequency and duty cycle for the FANCON[1:0] signals. The FANCON signals are clocks with a period specified by FC[1:0]FQ and a duty cycle specified by FC[1:0]HI. Bits[7:0] specify FANCON0 and bits[15:8] specify FANCON1. The system sleep state and these configuration bits interact to specify the FANCON[1:0] signal behavior as follows:

```
if ((system state=POS) & (FCxPOS=1)) then FANCONx=0;
  else if((FCxTHERM=1) & (THERM_L=0)) then FANCONx=1;
  else FANCONx is controlled by the FCxFQ and FCxHI fields;
```

Default: 0F0Fh.

Attribute: Read-write.

Bits	Description
15	Reserved.
14	<b>FC1POS.</b> See bit 6.
13	<b>FC1THERM.</b> See bit 5.
12	<b>FC1FQ.</b> See bit 4.
11:8	<b>FC1HI.</b> See bits[3:0].
7	Reserved.
6	<b>FC0POS. FANCON0 power-on-suspend control.</b> 1=When the system is placed into the POS state, FANCON0 goes Low. 0=The system sleep state has no affect on FANCON0.
5	<b>FC0THERM. FANCON0 THERM_L control.</b> 1=If the THERM_L pin is asserted then FANCON0 goes High. 0=THERM_L has no affect on FANCON0. This bit is ignored if the system is in POS and FC0POS is High.
4	<b>FC0FQ. FANCON0 frequency.</b> The frequency of FANCON[1:0] is specified by this bit. 0=A FANCON[1:0] period of 15 milliseconds. 1=A FANCON[1:0] period of 240 milliseconds.
3:0	<b>FC0HI. FANCON0 High time.</b> Specifies the percentage of time that FANCON[1:0] is High as: $\text{PERCENT HIGH TIME} = \text{FC}[1:0]\text{HI}/15;$ (the rest of the time of the clock period, FANCON[1:0] is Low). E.g., if FC1HI == 0, then the FANCON1 is Low all the time; if FC1HI == 1, then FANCON1 is High 1/15 <sup>th</sup> of the time; if FC1HI == 15, then FANCON1 is High all the time.

## 4.8 AC '97 Controller Registers

### 4.8.1 AC '97 Audio Register

#### 4.8.1.1 AC '97 Audio Configuration Registers (DevB:5xXX)

#### AC '97 Audio Controller Vendor and Device Id DevB:5x00

Default: 746D\_1022h

Attribute: Read-write Once

Bits	Description
31:16	<b>AC97ID.</b> Provides the AC '97 device identification.
15:0	<b>AMDDID.</b> Provides AMD's PCI vendor identification.

#### AC '97 Audio Controller Command and Status DevB:5x04

Default: 0200\_0000h

Attribute: See below.

Bits	Description
31:16	<b>STAT.</b> Read-only.
15:0	<b>CMD.</b> CMD[15:3]: Read-only. CMD[2]: <b>BMEN. Bus Master Enable.</b> Read-Write. 1=Enables Bus Master capability (scatter/gather). 0=Scatter/gather operation is disabled. CMD[1]: Read-only. CMD[0]: <b>IOEN. I/O Space.</b> Read-Write. 1=Enables access to the I/O space for this device, the mixer registers and the bus master controller registers. 0=Accesses to these registers are disabled.

#### AC '97 Audio Controller Revision Id and Class Code DevB:5x08

Default: 0401\_00XXh see below.

Attribute: Read-only

Bits	Description
31:24	<b>BASECLASS.</b> These bits are fixed at 04h indicating a multimedia device.
23:16	<b>SUBCLASS.</b> These bits are fixed at 01h indicating an audio device.
15:8	<b>PROGIF.</b>
7:0	<b>REVID. AC '97 Audio Controller silicon revision.</b> The value of this register is revision-dependent.

**AC '97 Audio Controller BIST, Header and Latency****DevB:5x0C**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only.
23:16	<b>HEADER.</b> Read-only.
15:8	<b>LATENCY.</b> Read-write. The latency timer defines the minimum amount of time, in PCI clock cycles, that the bus master can retain ownership of the bus.
7:0	Read-only.

**AC '97 Audio Mixer Base Address****DevB:5x10**

Default: 0000\_0001h

Attribute: See below.

Bits	Description
31:8	<b>AMBA. Base Address.</b> Read-write. These bits are used in the I/O space decode of the Audio interface registers. There is a maximum I/O block size of 256 bytes for this base address
7:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is fixed to 1, indicating a request for I/O space.

**AC '97 Audio Controller Bus Master Base Address****DevB:5x14**

Default: 0000\_0001h

Attribute: See below.

Bits	Description
31:6	<b>ABMBA. Base Address.</b> Read-write. These bits are used in the I/O space decode of the Audio PCI bus master controller registers. There is a maximum I/O block size of 64 bytes for this base address.
5:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is fixed to 1, indicating a request for I/O space.

**AC '97 Audio Controller Subsystem and Subsystem Vendor Id****DevB:5x2C**

Default: 0000\_0000h

Attribute: Read-write Once

Bits	Description
31:16	<b>SUBSYSID.</b> This register should be implemented for any function that could be instantiated more than once in a given system. For example, a system with 2 audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot, should have this register implemented. This register, in combination with the Subsystem Vendor ID register, enables the operating environment to distinguish one audio subsystem from the other(s).
15:0	<b>SUBVENID.</b> This register should be implemented for any function that could be instantiated more than once in a given system. For example a system with 2 audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot, should have this register implemented. This register, in combination with the Subsystem ID register, enables the operating environment to distinguish one audio subsystem from the other(s).

**AC '97 Audio Controller Interrupt Line and Interrupt Pin****DevB:5x3C**

Default: 0200h

Attribute: See below.

Bits	Description
15:11	Read-only.
10:8	<b>INTPIN.</b> Read-only. Indicates which PCI interrupt pin is used for the AC '97 audio interrupt. Hardwired to 010b to select PIRQB_L.
7:0	<b>INTLINE.</b> Read-write. This data is not used by hardware. It is used to communicate across software the interrupt line that the interrupt pin is connected to.

**AC '97 Audio Controller PCM Out Descriptor Shadow****DevB:5x40**

Default: 0000\_0000h

Attribute: Read-only

Bits	Description
31:1	<b>DESCR. Descriptor current value.</b> This is a test register for reading the current descriptor value.
0	Read-only.

**AC '97 Audio Controller PCM In Descriptor Shadow****DevB:5x44**

Default: 0000\_0000h

Attribute: Read-only

Bits	Description
31:1	<b>DESCR. Descriptor current value.</b> This is a test register for reading the current descriptor value.
0	Read-only.

**AC '97 Audio Controller Microphone In Descriptor Shadow****DevB:5x48**

Default: 0000\_0000h

Attribute: Read-only

Bits	Description
31:1	<b>DESCR. Descriptor current value.</b> This is a test register for reading the current descriptor value.
0	Read-only.

**AC '97 Audio Controller General Control****DevB:5x4C**

Default: 00h

Attribute: See below

Bits	Description
7:3	Reserved
2	<b>SWLSB. Subwoofer is LSB.</b> Read-write. When cleared to 0b the Center Front channel sample is always placed in the least significant sample of the Subwoofer/Center Front information pair, i.e., the audio data stream as represented in the host memory buffer pages must start with a Center Front channel sample. When set to 1b the Subwoofer channel sample is always placed in the least significant sample of the Center Front/Subwoofer information pair, i.e., the audio data stream as represented in the host memory buffer pages must start with a Subwoofer channel sample.
1	<b>RRLSB. Right Rear is LSB.</b> Read-write. When cleared to 0b the Left Rear channel sample is always placed in the least significant sample of the Right Rear/Left Rear information pair, i.e., the audio data stream as represented in the host memory buffer pages must start with a left rear channel sample. When set to 1b the Right Rear channel sample is always placed in the least significant sample of the Right Rear/Left Rear information pair, i.e., the audio data stream as represented in the host memory buffer pages must start with a Right Rear channel sample.
0	<b>RFLSB. Right Front is LSB.</b> Read-write. When cleared to 0b the left channel sample is always placed in the least significant sample of the stereo information pair, i.e., the stereo audio data stream as represented in the host memory buffer pages must start with a left channel sample. When set to 1b the right channel sample is always placed in the least significant sample of the stereo information pair, i.e., the stereo audio data stream as represented in the host memory buffer pages must start with a right channel sample.

**4.8.1.2 AC '97 Audio Mixer Registers**

The audio mixer registers are physically located in the codec. Accesses to these registers are forwarded through the AC link to the codec. Writes to a codec are completed when the data are inserted in the output slot. Reads from a codec may cause a PCI retry cycle. For more information about these registers see the AC '97 spec.

Table 61 on page 256 shows the register address space for the audio mixer registers. These registers are exposed in I/O space. The base address register for these registers is DevB:5x10.

**Table 61. Audio Mixer Register**

Address		Audio Mixer Registers
Primary	Secondary <sup>1</sup>	



**Table 61. Audio Mixer Register**

00h	80h	Reset
02h	82h	Master Volume Mute
04h	84h	Headphone Volume Mute
06h	86h	Master Volume Mono Mute
08h	88h	Master Tone (R & L)
0Ah	8Ah	PC_BEEP Volume Mute
0Ch	8Ch	Phone Volume Mute
0Eh	8Eh	Mic Volume Mute
10h	90h	Line In Volume Mute
12h	92h	CD Volume Mute
14h	94h	Video Volume Mute
16h	96h	Aux Volume Mute
18h	98h	PCM Out Volume Mute
1Ah	9Ah	Record Select
1Ch	9Ch	Record Gain Mute
1Eh	9Eh	Record Gain Mic Mute
20h	A0h	General Purpose
22h	A2h	3D Control
24h	A4h	AC '97 RESERVED
26h	A6h	Powerdown Ctrl/Stat
28h	A8h	Extended Audio
2Ah	AAh	Extended Audio Ctrl/Stat
2Ch	ACh	PCM Front DAC Rate
2Eh	AEh	PCM Surround DAC Rate
30h	B0h	PCM LFE DAC Rate
32h	B2h	PCM LR ADC Rate
34h	B4h	MIC ADC Rate
36h	B6h	6Ch Vol: C, LFE Mute
38h	B8h	6Ch Vol: L, R Surround Mute
3Ah:56h	BAh:D6h	RESERVED
58h	D8h	Vendor Reserved <sup>2</sup>
7Ah	FAh	Vendor Reserved <sup>2</sup>
7Ch	FCh	Vendor ID1 <sup>2</sup>
7Eh	FEh	Vendor ID2 <sup>2</sup>

**Notes:**

1. The IC does not support audio as a secondary codec.
2. Registers 58h, 7Ah, 7Ch, 7Eh, D8h, FAh, FCh, and FEh are multiplexed between audio and modem functions.

**4.8.1.3 AC '97 Audio Controller Bus Master Registers**

These registers are located in I/O space and reside in the AC '97 controller. The three channels, PCM Out, PCM In, and Microphone In, each have their own set of bus mastering registers. The following register descriptions apply to all three channels.

The base address register for these registers is DevB:5x14. See Section 4.1.2 on page 136 for a description of the register naming convention.

**Table 62. Audio Controller Bus Master Registers**

Mnemonic	Name	Default
AC00	PCM In Buffer Descriptor List Base Address	0000_0000h
AC04	PCM In Current Index Value	00h
AC05	PCM In Last Valid Index	00h
AC06	PCM In Status	0001h
AC08	PCM In Position in Current Buffer	0000h
AC0A	PCM In Prefetched Index Value	00h
AC0B	PCM In Control	00h
AC10	PCM Out Buffer Descriptor List Base Address	0000_0000h
AC14	PCM Out Current Index Value	00h
AC15	PCM Out Last Valid	00h
AC16	PCM Out Status	0001h
AC18	PCM Out Position in Current Buffer	0000h
AC1A	PCM Out Prefetched Index	00h
AC1B	PCM Out Control	00h
AC20	Mic. In Buffer Descriptor List Base Address	0000_0000h
AC24	Mic. In Current Index Value	00h
AC25	Mic. In Last Valid	00h
AC26	Mic. In Status	0001h
AC28	Mic. In Position in Current Buffer	0000h
AC2A	Mic. In Prefetched Index	00h
AC2B	Mic. In Control	00h
AC2C	Global Control	0000_0000h
AC30	Global Status	0030_0000h
AC34	Codec Access Semaphore	00h

The Global Control (AC2C), Global Status (AC30), and Codec Access Semaphore (AC34) registers are aliased to the same global registers in the audio and modem I/O space. Therefore, a Read-write to these registers in either audio or modem I/O space affects the same physical register.

The following status and configuration bits are powered by the AUX plane:

- AC30[MD3]
- AC30[AD3]

**AC '97 Audio Controller Buffer Descriptor List Base Address****AC00, AC10, AC20**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:3	<b>BDLBA. Buffer Descriptor List Base address.</b> Read-write. The entries should be aligned on 8 byte boundaries.
2:0	0h

**AC '97 Audio Controller Current Index Value****AC04, AC14, AC24**

Default: 00h

Attribute: See below.

Bits	Description
7:5	0h
4:0	<b>CIV. Current Index Value.</b> Read-only. These bits represent which buffer descriptor within the list of 32 descriptors is being processed currently. As each descriptor is processed, this value is incremented.

**AC '97 Audio Controller Last Valid Index****AC05, AC15, AC25**

Default: 00h

Attribute: See below.

Bits	Description
7:5	0h
4:0	<b>LVI.</b> Read-write. These bits indicate the last valid descriptor in the list. This value is updated by the software as it prepares new buffers and adds to the list.

**AC '97 Audio Controller Status****AC06, AC16, AC26**

Default: 0001h

Attribute: See below.

Bits	Description
15:5	Reserved.
4	<b>FIFOERR. FIFO Error.</b> Read-write. The hardware sets this bit if an under-run or over-run occurs. This bit is cleared by writing a 1 to this bit position.
3	<b>BCIS. Buffer Completion Interrupt Status.</b> Read-write. This bit is set by the hardware after the last sample of a buffer has been processed, and if the Interrupt on Completion (IOC) bit is set in the command byte of the buffer descriptor. Remains active until software clears bit. This bit is cleared by writing a 1 to this bit position.
2	<b>LVBCI. Last Valid Buffer Completion Interrupt.</b> Read-write. This bit is set to 1 by hardware when last valid buffer has been processed. It remains active until cleared by software. This bit indicates the occurrence of the event signified by the last valid buffer being processed. Thus, this is an event status bit that can be cleared by software once this event has been recognized. This event causes an interrupt if the enable bit in the Control Register is set. This bit is cleared by writing a "1" to this bit position.
1	<b>CELV. Current Equals Last Valid.</b> Read-only. 1 = Current Index is equal to the value in the Last Valid Index Register, and the buffer pointed to by the CIV has been processed (i.e., after the last valid buffer has been processed). This bit is very similar to bit 2, except this bit reflects the state rather than the event. This bit reflects the state of the controller, and remains set until the controller exits this state. Hardware clears this bit when controller exits state (i.e., until a new value is written to the LVI register).
0	<b>BMCH. Bus master controller halted.</b> Read-only. This bit is set to 1b because of the Run/Pause bit being cleared in which case BMCH may or may not be asserted immediately, depending on whether a PCI transaction had previously been requested. If a PCI transaction had previously been requested, BMCH assertion occurs after completion of the single transaction. Bus master controller halted can also happen once the controller has processed the last valid buffer in which case it sets ACx6[CELV] and halts. This bit is cleared and the bus master controller resumes operation when both the Run/Pause bit is set and the controller has exited the ACx6[CELV] status.

**Notes:**

1. *Modem in: FIFO error indicates a FIFO overrun. The FIFO pointers don't increment, the incoming data is not written into the FIFO, thus is lost.*
2. *Modem out: FIFO error indicates a FIFO under-run. The sample transmitted in this case should be the last valid sample.*

**AC '97 Audio Controller Position in Current Buffer****AC08, AC18, AC28**

Default: 0000h

Attribute: Read-only.

Bits	Description
15:0	<b>PICB. Position in Current Buffer.</b> These bits represent the number of samples left to be processed in the current buffer.

**AC '97 Audio Controller Prefetched Index Value****AC0A, AC1A, AC2A**

Default: 00h

Attribute: Read-only.

Bits	Description
7:5	0h
4:0	<b>PIV.</b> These bits represent which buffer descriptor in the list has been prefetched.

**AC '97 Audio Controller Control Register****AC0B, AC1B, AC2BG**

Default: 00h

Attribute: See below.

Bits	Description
7:5	Reserved
4	<b>IOCEN. Interrupt On Completion Enable.</b> Read-write. I This bit controls whether or not an interrupt occurs when a buffer completes with the IOC bit set in its descriptor. 1 = Enable. 0 = Disable; bit 3 in the Status register is still set, but the interrupt does not occur.

3	<b>FEIEN. FIFO Error Interrupt Enable.</b> Read-write. This bit controls whether the occurrence of a FIFO error causes an interrupt or not. 1 = Enable. 0 = Disable; bit 4 in the Status Register is set, but the interrupt does not occur.
2	<b>LVBIEN. Last Valid Buffer Interrupt Enable.</b> Read-write. This bit controls whether the completion of the last valid buffer causes an interrupt or not. 1 = Enable. 0 = Disable; bit 2 in the Status register is still set, but the interrupt does not occur.
1	<b>REGRST. Register Reset.</b> Read-write. 1 = Contents of all registers of the bus master engine including the FIFOs are reset, except the interrupt enable bits (bits 4,3,2 of this register). Software needs to set this bit. It must be set only when the Run/Pause bit RUNBM is cleared and ACx6[BMCH] is asserted. Setting it when either RUNBM is asserted (set) or when ACx6[BMCH] is de-asserted (clear) causes undefined consequences. This bit is self-clearing (software need not clear it). 0 = Removes reset condition.
0	<b>RUNBM. Run/Pause Bus Master.</b> Read-write. 1 = Enable bus master operation. 0 = Disable bus master; this results in all state information being retained (i.e., master mode operation can be stopped and then resumed).

**AC '97 Audio Controller Global Control**

**AC2C**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:22	Reserved
21:20	<b>PCM46EN. PCM 4/6 Enable.</b> Read-write. Configures PCM Output for 2, 4 or 6 channel mode 00 = 2-channel mode (default) 01 = 4 channel mode 10 = 6 channel mode 11 = Reserved
19:6	Reserved
5	<b>SRIEN. Secondary Resume Interrupt Enable.</b> Read-write. 1 = Enable an interrupt to occur when the secondary codec causes a resume event on the AC-link. 0 = Disable interrupt.
4	<b>PRIEN. Primary Resume Interrupt Enable.</b> Read-write. 1 = Enable an interrupt to occur when the primary codec causes a resume event on the AC-link. 0 = Disable interrupt.

Bits	Description
3	<b>SHUTOFF. ACLINK Shut Off.</b> Read-write. 1 = Disable the AC-link signals (drive all AC '97 outputs Low and ignore all AC '97 input buffer enables). 0 = Enable the AC-link signals. Wake-up event detection is independent of this bit.
2	<b>WRST. AC '97 Warm Reset.</b> Read-write. Writing a 1 to this bit causes a warm reset to occur on the AC-link. The warm reset awakens a suspended codec without clearing its internal registers. If software attempts to perform a warm reset while ACCLK is running, the write is ignored and the bit is not changed. A warm reset can only occur in the absence of ACCLK. This bit is self-clearing (it clears itself after the reset has occurred and ACCLK has started).
1	<b>CRST_L. AC '97 Cold Reset, active Low.</b> Read-write. Writing a 0 to this bit causes a cold reset to occur on the AC-link. All data in the codec is lost. Software needs to set this bit no sooner than after 1 $\mu$ s has elapsed. This bit reflects the state of the ACRST_L pin. This bit is cleared to 0 upon entering S3/S4/S5 sleep states, and by SM_RESET_L and RESET_L assertions.
0	<b>GPIIEN. GPI Interrupt Enable.</b> Read-write. This bit controls whether the change in status of any GPI (General Purpose Input/Output, configured as an input) causes an interrupt. 1 = The change on value of a GPI causes an interrupt and sets bit 0 of the Global Status Register. 0 = Bit 0 of the Global Status Register is set, but an interrupt is not generated.

**AC '97 Audio Controller Global Status****AC30**

Default: 0030\_0000h

Attribute: See below.

Bits	Description
31:22	Reserved
21	<b>PCM6CAP. 6 Channel Capability.</b> Read-only. Hardwired to 1. 0 = AC '97 doesn't support 6-channel Audio output. 1 = AC '97 supports 6-channel Audio output.
20	<b>PCM4CAP. 4 Channel Capability.</b> Read-only. Hardwired to 1. 0 = AC '97 doesn't support 4-channel Audio output. 1 = AC '97 supports 4-channel Audio output.
19:18	Reserved
17	<b>MD3. Power down semaphore for Modem.</b> Read-write. This bit is used by software in conjunction with the AD3 bit to coordinate the entry of the two codecs into D3 state. This bit resides in the VDD_COREX well and maintains context across power states. This bit has no hardware function.
16	<b>AD3. Power down semaphore for Audio.</b> Read-write. This bit is used by software in conjunction with the MD3 bit to coordinate the entry of the two codecs into D3 state. This bit resides in the VDD_COREX well and maintains context across power states. This bit has no hardware function.
15	<b>RCSTAT. Read Complete Status.</b> Read-write. This bit indicates the status of Codec read completions. 1 = Codec read failed due to one of the following errors: a time-out, or that ACCLK is detected to not be operating adequately. This bit remains set until being cleared by software. 0 = Codec read completes normally. This bit is cleared by writing a 1 to this bit position.
14	<b>S12BIT3. Bit 3 of slot 12.</b> Display bit 3 of the most recent slot 12 from modem codec.
13	<b>S12BIT2. Bit 2 of slot 12.</b> Display bit 2 of the most recent slot 12 from modem codec.

Bits	Description (Continued)
12	<b>S12BIT1. Bit 1 of slot 12.</b> Display bit 1 of the most recent slot 12 from modem codec.
11	<b>SRINT. Secondary Resume Interrupt.</b> Read-write. This bit indicates that a resume event occurred on ACSDI[1]. 1 = Resume event occurred. Cleared by writing a 1 to this bit position.
10	<b>PRINT. Primary Resume Interrupt.</b> Read-write. This bit indicates that a resume event occurred on ACSDI[0]. 1 = Resume event occurred. Cleared by writing a 1 to this bit position.
9	<b>SCRDY. Secondary Codec Ready.</b> Read-only. Reflects the state of the codec ready bit in ACSDI[1]. Bus masters ignore the condition of the codec ready bits. Software must check this bit before starting the bus masters. This bit is cleared with assertion of GLOB_CNT[SHUTOFF]. This bit is also cleared when ACCLK is detected to not be operating adequately.
8	<b>PCRDY. Primary Codec Ready.</b> Read-only. Reflects the state of the codec ready bit in ACSDI[0]. Bus masters ignore the condition of the codec ready bits. Software must check this bit before starting the bus masters. This bit is cleared with assertion of AC2C[SHUTOFF]. This bit is also cleared when ACCLK is detected to not be operating adequately.
7	<b>MICINT. Mic In Interrupt.</b> Read-only. This bit indicates that one of the Mic in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
6	<b>POINT. PCM Out Interrupt.</b> Read-only. This bit indicates that one of the PCM out channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
5	<b>PIINT. PCM In Interrupt.</b> Read-only. This bit indicates that one of the PCM in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
4	Reserved
3	Reserved
2	<b>MOINT. Modem Out Interrupt.</b> Read-only. This bit indicates that one of the modem out channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
1	<b>MIINT. Modem In Interrupt.</b> Read-only. This bit indicates that one of the modem in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
0	<b>GPIINT. GPI Status Change Interrupt.</b> Read-write. This bit is set whenever bit 0 of slot 12 is set. This happens when the value of any of the GPIOs currently defined as inputs changes. If '1' this bit sets also PM20[AC97_STS]. 1 = Input changed. This bit is cleared by writing a 1 to this bit position.

**AC '97 Audio Controller CODEC Access Semaphore****AC34**

Default: 00h

Attribute: See below.

Bits	Description
7:1	Reserved
0	<b>CAS. Codec Access Semaphore.</b> Read-write. This bit is read by software to check whether a codec access is currently in progress. The act of reading this register sets this bit to 1. The driver that reads this bit can then perform a codec I/O access. Once the access is completed, or if ACCLK is detected to not be operating adequately, hardware automatically clears this bit. 0 = No access in progress. If and only if CAS = 1 and a codec access has not yet been requested (i.e., the PCI transaction has not yet been initiated), CAS may be cleared by writing a 0 to this bit position.

**4.8.2 AC '97 Modem Registers****4.8.2.1 AC '97 Modem Configuration Registers (DevB:6xXX)****AC '97 Modem Controller Vendor and Device Id****DevB:6x00**

Default: 746E\_1022h

Attribute: Read-write Once.

Bits	Description
31:16	<b>AC97ID.</b> Provides the AC '97 device identification.
15:0	<b>AMDID.</b> Provides AMD's PCI vendor identification.

**AC '97 Modem Controller Command and Status****DevB:6x04**

Default: 0200\_0000h

Attribute: See below.

Bits	Description
31:16	<b>STAT.</b> Read-only.
15:0	<b>CMD.</b> CMD[15:3]: Read-only. CMD[2]: <b>BMEN. Bus Master Enable.</b> Read-write. 1 = Enables Bus Master capability (scatter/gather). 0 = Scatter/gather operation is disabled. CMD[1]: <b>Memory Space.</b> Read-only. CMD[0]: <b>IOEN. I/O Space.</b> Read-write. 1 = Enables access to the I/O space for this device, the mixer registers and the bus master controller registers. 0 = Accesses to these registers are disabled.



**AC '97 Modem Controller Revision Id and Class Code****DevB:6x08**

Default: 0703\_00XXh see below.

Attribute: Read-only.

Bits	Description
31:24	<b>BASECLASS.</b> These bits are fixed at 07h indicating a simple communications controller.
23:16	<b>SUBCLASS.</b> These bits are fixed at 03h indicating a generic modem.
15:8	<b>PROGIF.</b>
7:0	<b>REVID. AC '97 Modem Controller silicon revision.</b> The value of this register is revision-dependent.

**AC '97 Modem Controller BIST, Header and Latency****DevB:6x0C**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only.
23:16	<b>HEADER.</b> Read-only.
15:8	<b>LATENCY.</b> Read-write. The latency timer defines the minimum amount of time, in PCI clock cycles, that the bus master can retain ownership of the bus.
7:0	Read-only.

**AC '97 Modem Mixer Base Address****DevB:6x10**

Default: 0000\_0001h

Attribute: See below.

Bits	Description
31:8	<b>MMBA. Base Address.</b> Read-write. These bits are used in the I/O space decode of the modem interface registers. There is a maximum I/O block size of 256 bytes for this base address
7:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is set to one, indicating a request for I/O space.

**AC '97 Modem Controller Bus Master Base Address****DevB:6x14**

Default: 0000\_0001h

Attribute: See below.

Bits	Description
31:7	<b>MBMBA. Base Address.</b> Read-write. These bits are used in the I/O space decode of the modem PCI bus master controller registers. There is a maximum I/O block size of 128 bytes for this base address.
6:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is set to one, indicating a request for I/O space.

**AC '97 Modem Controller Subsystem and Subsystem Vendor Id****DevB:6x2C**

Default: 0000\_0000h

Attribute: Read-write Once.

Bits	Description
31:16	<b>SUBSYSID.</b> This register should be implemented for any function that could be instantiated more than once in a given system. For example, a system with 2 audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot, should have this register implemented. This register, in combination with the Subsystem Vendor ID register, enable the operating environment to distinguish one audio subsystem from the other(s).
15:0	<b>SUBVENID.</b> This register should be implemented for any function that could be instantiated more than once in a given system. For example a system with 2 audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot, should have this register implemented. This register, in combination with the Subsystem ID register, enable the operating environment to distinguish one audio subsystem from the other(s).

**AC '97 Modem Controller Interrupt Line and Interrupt Pin****DevB:6x3C**

Default: 0200h

Attribute: See below.

Bits	Description
15:11	Read-only.
10:8	<b>INTPIN.</b> Indicates which PCI interrupt pin is used for the AC '97 modem interrupt. Read-only. Hardwired to 010b to select PIRQB_L.
7:0	<b>INTLINE.</b> Read-write. This data is not used by hardware. It is used to communicate across software the interrupt line that the interrupt pin is connected to.

**AC '97 Modem Controller Modem Out Descriptor Shadow****DevB:6x40**

Default: 0000\_0000h

Attribute: Read-only.

Bits	Description
31:1	<b>DESCR. Descriptor current value.</b> This is a test register for reading the current descriptor value.
0	Read-only.

**AC '97 Modem Controller Modem In Descriptor Shadow****DevB:6x44**

Default: 0000\_0000h

Attribute: Read-only.

Bits	Description
31:1	<b>DESCR. Descriptor current value.</b> This is a test register for reading the current descriptor value.
0	Read-only.

#### 4.8.2.2 AC '97 Modem Mixer Registers

The modem mixer registers are physically located in the codec. Accesses to these registers are forwarded through the AC link to the codec. Writes to a codec are completed when the data are inserted in the output slot. Reads from a codec cause a PCI retry cycle. For more information about these registers see the AC '97 spec.

In the case of the split codec implementation (i.e., separate primary and secondary codecs), accesses to the modem mixer registers in different codecs are differentiated by the controller by using address offsets 00h–7Fh for the primary codec and address offsets 80h–FFh for the secondary codec.

Table 63 on page 267 shows the register addresses for the modem mixer registers. These registers are exposed in I/O space. The base address register for these registers is DevB:6x10.

**Table 63. Modem Mixer Register**

Address		Modem Mixer Registers
Primary	Secondary <sup>1</sup>	
00h:38h	80h:B8h	RESERVED
3Ch	BCh	Extended Modem ID
3Eh	BEh	Extended Modem Stat/Ctrl
40h	C0h	Line 1 DAC/ADC Rate
42h	C2h	Line 2 DAC/ADC Rate <sup>2</sup>
44h	C4h	Handset DAC/ADC Rate <sup>2</sup>
46h	C6h	Line 1 DAC/ADC Level Mute
48h	C8h	Line 2 DAC/ADC Level Mute <sup>2</sup>
4Ah	CAh	Handset DAC/ADC Level Mute <sup>2</sup>
4Ch	CCh	GPIO Pin Config
4Eh	CEh	GPIO Polarity/Type
50h	D0h	GPIO Pin Sticky
52h	D2h	GPIO Pin Wake Up
54h	D4h	GPIO Pin Status
56h	D6h	Misc. Modem AFE Stat/Ctrl
58h	D8h	Vendor Reserved <sup>3</sup>
7Ah	FAh	Vendor Reserved <sup>3</sup>
7Ch	FCh	Vendor ID1 <sup>3</sup>
7Eh	FEh	Vendor ID2 <sup>3</sup>

**Notes:**

1. The IC supports a modem codec as either primary or secondary codec, but does not support two modem codecs.
2. Registers 42h, 44h, 48h, 4Ah, C2h, C4h, C8h, and CAh are for functions not supported by the IC.
3. Registers 58h, 7Ah, 7Ch, 7Eh, D8h, FAh, FCh, and FEh are multiplexed between audio and modem functions.

#### 4.8.2.3 AC '97 Modem Controller Bus Master Registers

These registers are exposed in I/O space and reside in the AC '97 controller. The two channels, Modem In and Modem Out, each have their own set of Bus Mastering registers. The following register descriptions apply to both channels.

The base address register for these registers is DevB:6x14. See Section 4.1.2 on page 136 for a description of the register naming convention.

**Table 64. Modem Bus Master Registers**

Mnemonic	Name	Default
MC00	Modem In Buffer Descriptor List Base Address	0000_0000h
MC04	Modem In Current Index Value	00h
MC05	Modem In Last Valid Index	00h
MC06	Modem In Status	0001h
MC08	Modem In Position in Current Buffer	0000h
MC0A	Modem Prefetch Index Value	00h
MC0B	Modem In Control	00h
MC10	Modem Out Buffer Descriptor List Base Address	0000_0000h
MC14	Modem Out Current Index Value	00h
MC15	Modem Out Last Valid	00h
MC16	Modem Out Status	0001h
MC18	Modem Out Position in Current Buffer	0000h
MC1A	Modem Out Prefetched Index	00h
MC1B	Modem Out Control	00h
MC3C	Global Control	0000_0000h
MC40	Global Status	0030_0000h
MC44	Codec Access Semaphore	00h
MC48	GPIO Pin Status	0000h

The Global Control (MC3C), Global Status (MC40), Codec Access Semaphore (MC44) registers are aliased to the same global registers in the audio and modem I/O space. Therefore, a Read-write to these registers in either audio or modem I/O space affects the same physical register. The following status and configuration bits are powered by the VDD\_COREX well:

- MC40[MD3]
- MC40[AD3]

#### AC '97 Modem Controller Buffer Descriptor List Base Address

**MC00, MC10**

Default: 0000\_0000h

Attribute: See below.

Bits	Description
31:3	<b>BDLBA. Buffer Descriptor List Base address.</b> Read-write. These bits represent address bits 31:3. The entries should be aligned on 8 byte boundaries.
2:0	Read-only.

**AC '97 Modem Controller Current Index Value****MC04, MC14**

Default: 00h

Attribute: See below.

Bits	Description
7:5	Read-only.
4:0	<b>CIV. Current Index Value.</b> Read-only. These bits represent which buffer descriptor within the list of 32 descriptors is being processed currently. As each descriptor is processed, this value is incremented.

**AC '97 Modem Controller Last Valid Index****MC05, MC15**

Default: 00h

Attribute: See below.

Bits	Description
7:5	Read-only.
4:0	<b>LVI.</b> Read-write. These bits indicate the last valid descriptor in the list. This value is updated by the software as it prepares new buffers and adds to the list.

**AC '97 Modem Controller Status****MC06, MC16**

Default: 0001h

Attribute: See below.

Bits	Description
15:5	Reserved.
4	<b>FIFOERR.</b> Read-write. FIFO Error. The hardware sets this bit if an under-run or over-run occurs. This bit is cleared by writing a 1 to this bit position.
3	<b>BCIS.</b> Read-write. Buffer Completion Interrupt Status. This bit is set by the hardware after the last sample of a buffer has been processed, AND if the Interrupt on Completion (IOC) bit is set in the command byte of the buffer descriptor. Remains active until software clears bit. This bit is cleared by writing a 1 to this bit position.
2	<b>LVBCI.</b> Read-write. Last Valid Buffer Completion Interrupt. This bit is set to 1 by hardware when last valid buffer has been processed. It remains active until cleared by software. This bit indicates the occurrence of the event signified by the last valid buffer being processed. Thus, this is an event status bit that can be cleared by software once this event has been recognized. This event causes an interrupt if the enable bit in the Control Register is set. This bit is cleared by writing a 1 to this bit position.
1	<b>CELV.</b> Read-only. Current Equals Last Valid. 1 = Current Index is equal to the value in the Last Valid Index Register, and the buffer pointed to by the CIV has been processed (i.e., after the last valid buffer has been processed). This bit is very similar to bit 2, except this bit reflects the state rather than the event. This bit reflects the state of the controller, and remains set until the controller exits this state. Hardware clears this bit when controller exits state (i.e., until a new value is written to the LVI register).
0	<b>BMCH.</b> Read-only. Bus master controller halted. This bit is set to 1b because of the Run/Pause bit being cleared in which case BMCH may or may not be asserted immediately, depending on whether a PCI transaction had previously been requested. If a PCI transaction had previously been requested, BMCH assertion occurs after completion of the single transaction. Bus master controller halted can also happen once the controller has processed the last valid buffer in which case it sets MCx6[CELV] and halts. This bit is cleared and the bus master controller resumes operation when both the Run/Pause bit is set and the controller has exited the MCx6[CELV] status.

**Notes:**

1. *Modem in: FIFO error indicates a FIFO overrun. The FIFO pointers don't increment, the incoming data is not written into the FIFO, thus is lost.*
2. *Modem out: FIFO error indicates a FIFO under-run. The sample transmitted in this case should be the last valid sample.*

**AC '97 Modem Controller Position in Current Buffer****MC08, MC18**

Default: 0000h

Attribute: Read-only.

Bits	Description
15:0	<b>PICB. Position in Current Buffer.</b> These bits represent the number of samples left to be processed in the current buffer.

**AC '97 Modem Controller Prefetched Index Value****MC0A, MC1A**

Default: 00h

Attribute: Read-only.

Bits	Description
7:5	Read-only.
4:0	<b>PIV.</b> These bits represent which buffer descriptor in the list has been prefetched.

**AC '97 Modem Controller Control Register****MC0B, MC1B**

Default: 00h

Attribute: see below.

Bits	Description
7:5	Reserved
4	<b>IOCEN. Interrupt On Completion Enable.</b> Read-write. This bit controls whether or not an interrupt occurs when a buffer completes with the IOC bit set in its descriptor. 1 = Enable. 0 = Disable; bit 3 in the Status register is still set, but the interrupt does not occur.

Bits	Description
3	<b>FEIEN. FIFO Error Interrupt Enable.</b> Read-write. This bit controls whether the occurrence of a FIFO error causes an interrupt or not. 1 = Enable. 0 = Disable; bit 4 in the Status Register is set, but the interrupt does not occur.
2	<b>LVBIEN. Last Valid Buffer Interrupt Enable.</b> Read-write. This bit controls whether the completion of the last valid buffer causes an interrupt or not. 1 = Enable. 0 = Disable; bit 2 in the Status register is still set, but the interrupt does not occur.
1	<b>REGRST. Register Reset.</b> Read-write. 1 = Contents of all registers of the bus master engine including the FIFOs are reset, except the interrupt enable bits (bits 4,3,2 of this register). Software needs to set this bit. It must be set only when the Run/Pause bit RUNBM is cleared and MCx6[BMCH] is asserted. Setting it when either RUNBM is asserted (set) or when MCx6[BMCH] is de-asserted (clear) causes undefined consequences. This bit is self-clearing (software need not clear it). 0 = Removes reset condition.
0	<b>RUNBM. Run/Pause Bus Master.</b> Read-write. 1 = Enable bus master operation. 0 = Disable bus master; this results in all state information being retained (i.e., master mode operation can be stopped and then resumed).

**AC '97 Modem Controller Global Control****MC3C**

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31:22	Reserved
21:20	<b>PCM46EN. PCM 4/6 Enable.</b> Read-write. Configures PCM Output for 2, 4 or 6 channel mode 00 = 2-channel mode (default) 01 = 4 channel mode 10 = 6 channel mode 11 = Reserved
19:6	Reserved
5	<b>SRIEN. Secondary Resume Interrupt Enable.</b> Read-write. 1 = Enable an interrupt to occur when the secondary codec causes a resume event on the AC link. 0 = Disable interrupt.
4	<b>PRIEN. Primary Resume Interrupt Enable.</b> Read-write. 1 = Enable an interrupt to occur when the primary codec causes a resume event on the AC-link. 0 = Disable interrupt.

3	<b>SHUTOFF. ACLINK Shut Off.</b> Read-write. 1 = Disable the AC-link signals (drive all AC '97 outputs Low and ignore all AC '97 input buffer enables). 0 = Enable the AC-link signals. Wake-up event detection is independent of this bit.
2	<b>WRST. AC '97 Warm Reset.</b> Read-write. Writing a 1 to this bit causes a warm reset to occur on the AC-link. The warm reset awakens a suspended codec without clearing its internal registers. If software attempts to perform a warm reset while ACCLK is running, the write is ignored and the bit is not changed. A warm reset can only occur in the absence of ACCLK. This bit is self-clearing (it clears itself after the reset has occurred and ACCLK has started).
1	<b>CRST_L. AC '97 Cold Reset, active Low.</b> Read-write. Writing to 0 to this bit causes a cold reset to occur on the AC-link. All data in the codec is lost. Software needs to set this bit no sooner than after 1 $\mu$ s has elapsed. This bit reflects the state of the ACRST_L pin. This bit is cleared to 0 upon entering S3/S4/S5 sleep states, and by SM_RESET_L and RESET_L assertions.
0	<b>GPIIEN. GPI Interrupt Enable.</b> Read-write. This bit controls whether the change in status of any GPI (General Purpose Input/Output, configured as an input) causes an interrupt. 1 = The change on value of a GPI causes an interrupt and sets bit 0 of the Global Status Register. 0 = Bit 0 of the Global Status Register is set, but an interrupt is not generated.

**AC '97 Modem Controller Global Status****MC40**

Default: 0030\_0000h

Attribute: see below.

Bits	Description
31:22	Reserved
21	<b>PCM6CAP. 6 Channel Capability.</b> Read-only. Hardwired to 1. 0 = AC '97 doesn't support 6-channel Audio output. 1 = AC '97 supports 6-channel Audio output.
20	<b>PCM4CAP. 4 Channel Capability.</b> Read-only. Hardwired to 1. 0 = AC '97 doesn't support 4-channel Audio output. 1 = AC '97 supports 4-channel Audio output.
19:18	Reserved
17	<b>MD3. Power down semaphore for Modem.</b> Read-write. This bit is used by software in conjunction with the AD3 bit to coordinate the entry of the two codecs into D3 state. This bit resides in the VDD_COREX well and maintains context across power states. This bit has no hardware function.
16	<b>AD3. Power down semaphore for Audio.</b> Read-write. This bit is used by software in conjunction with the MD3 bit to coordinate the entry of the two codecs into D3 state. This bit resides in the VDD_COREX well and maintains context across power states. This bit has no hardware function.
15	<b>RCSTAT. Read Complete Status.</b> Read-write. This bit indicates the status of Codec read completions. 1 = Codec read failed due to one of the following errors: a time-out, or that ACCLK is detected to not be operating adequately. This bit remains set until being cleared by software. 0 = Codec read completes normally. This bit is cleared by writing a 1 to this bit position.
14	<b>S12BIT3. Bit 3 of slot 12: Display bit 3 of the most recent slot 12 from modem codec.</b>
13	<b>S12BIT2. Bit 2 of slot 12: Display bit 2 of the most recent slot 12 from modem codec.</b>
12	<b>S12BIT1. Bit 1 of slot 12: Display bit 1 of the most recent slot 12 from modem codec.</b>



Bits	Description (Continued)
11	<b>SRINT. Secondary Resume Interrupt.</b> Read-write. This bit indicates that a resume event occurred on ACSDI[1]. 1 = Resume event occurred. Cleared by writing a 1 to this bit position.
10	<b>PRINT. Primary Resume Interrupt.</b> Read-write. This bit indicates that a resume event occurred on ACSDI[0]. 1 = Resume event occurred. Cleared by writing a 1 to this bit position.
9	<b>SCRDY. Secondary Codec Ready.</b> Read-only. Reflects the state of the codec ready bit in ACSDI[1]. Bus masters ignore the condition of the codec ready bits. Software must check this bit before starting the bus masters. This bit is cleared with assertion of MC3C[SHUTOFF]. This bit is also cleared when ACCLK is detected to not be operating adequately.
8	<b>PCRDY. Primary Codec Ready.</b> Read-only. Reflects the state of the codec ready bit in ACSDI[0]. Bus masters ignore the condition of the codec ready bits. Software must check this bit before starting the bus masters. This bit is cleared with assertion of MC3C[SHUTOFF]. This bit is also cleared when ACCLK is detected to not be operating adequately.
7	<b>MICINT. Mic In Interrupt.</b> Read-only. This bit indicates that one of the Mic in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
6	<b>POINT. PCM Out Interrupt.</b> Read-only. This bit indicates that one of the PCM out channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
5	<b>PIINT. PCM In Interrupt.</b> Read-only. This bit indicates that one of the PCM in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
4	Reserved
3	Reserved
2	<b>MOINT. Modem Out Interrupt.</b> Read-only. This bit indicates that one of the modem out channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
1	<b>MIINT. Modem In Interrupt.</b> Read-only. This bit indicates that one of the modem in channel interrupts occurred. 1 = Interrupt occurred. When the specific interrupt is cleared, this bit is cleared automatically.
0	<b>GPIINT. GPI Status Change Interrupt.</b> Read-write. This bit is set whenever bit 0 of slot 12 is set. This happens when the value of any of the GPIOs currently defined as inputs changes. If '1' this bit sets also PM20[AC97_STS]. 1 = input changed. This bit is cleared by writing a 1 to this bit position.

**AC '97 Modem Controller CODEC Access Semaphore****MC44**

Default: 00h

Attribute: see below.

Bits	Description
7:1	Reserved
0	<b>CAS. Codec Access Semaphore.</b> Read-write. This bit is read by software to check whether a codec access is currently in progress. The act of reading this register sets this bit to 1. The driver that reads this bit can then perform a codec I/O access. Once the access is completed, or if ACCLK is detected to not be operating adequately, hardware automatically clears this bit. 0 = No access in progress. If and only if CAS = 1 and a codec access has not yet been requested (i.e., the PCI transaction has not yet been initiated), CAS may be cleared by writing a 0 to this bit position.

AC '97 Modem GPIO StatusMC48Default:00h

Attribute: Read-only.

Bits	Description
15:0	<b>GPIO.</b> Mirror register updated automatically with each valid slot 12 input to reflect or mirror the current state of the modem codec GPIO pins.

## 4.9 USB Controller Registers

### 4.9.1 USB Open Host Controller Registers

#### 4.9.1.1 USB Open Host Controller Configuration Registers (Dev0:[1:0]xXX)

These registers are located in PCI configuration space on the secondary PCI interface, device 0 (function 0 and 1 for each of the two OHCI controllers). See Section 4.1.2 on page 136 for a description of the register naming convention.

**USB Controller Vendor And Device ID Register****Dev0:[1:0]x00**

Default: 7464\_1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>USB controller device ID.</b>
15:0	<b>Vendor ID.</b>

**USB Controller Status And Command Register****Dev0:[1:0]x04**

Default: 0280\_0000h.

Attribute: See below.

Bits	Description
31:30	Reserved.
29	<b>Signaled master abort.</b> Read; write 1 to clear. This bit is set High by the hardware when a bus master cycle is terminated with a master abort.
28	<b>Received target abort.</b> Read; write 1 to clear. This bit is set High by the hardware when a target abort command is received during a master cycle.
27	Reserved.
26:25	<b>DEVSEL timing.</b> Read-only. These bits are fixed at 01b and specifies "medium" timing as defined by the PCI specification.
24:9	Read-only. These bits are fixed at their default values.
8	<b>SERR_L detection enable.</b> Read-write. This bit has no effect on the hardware.
7:5	Reserved.
4	<b>Memory write and invalidate enable.</b> Read-write. 1=Enables bus master to issue memory write and invalidate cycles.
3	Reserved.
2	<b>Bus master enable.</b> Read-write. 1=Enables bus master capability.
1	<b>Memory space enable.</b> Read-write. 1=Enables access to the memory space for this device (the host controller registers).
0	<b>I/O space enable.</b> Read-write. 1=Enables access to the I/O space for this device (the keyboard/mouse controller ports for legacy emulation).

**USB Revision ID, Programming Interface, Sub Class and Base Registers****Dev0:[1:0]x08**

Default: 0C03\_10XXh see below.

Attribute: Read-only.

Bits	Description
31:24	<b>BASECLASS.</b>
23:16	<b>SUBCLASS.</b>
15:8	<b>PROGIF.</b>
7:0	<b>REVISIONID.</b> The value of this register is revision-dependent.

**USB Controller BIST, Header, Latency, and Cacheline Size Register****Dev0:0x0C**

Default: 0080\_0000h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. These bits fixed at their default values.
23:16	<b>HEADER.</b> Read-only. These bits fixed at their default values.
15:8	<b>LATENCY.</b> Bits[7:3] are read-write; bits[2:0] are read-only at their default value.
7:0	<b>CACHE.</b> Read-write. 08h=32 Bytes, 00h=0 Bytes. No other values are allowed.

**USB Controller BIST, Header, Latency, and Cacheline Size Register****Dev0:1x0C**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. These bits fixed at their default values.
23:16	<b>HEADER.</b> Read-only. These bits fixed at their default values.
15:8	<b>LATENCY.</b> Bits[7:3] are read-write; bits[2:0] are read-only at their default value.
7:0	<b>CACHE.</b> Read-write. 08h=32 Bytes, 00h=0 Bytes. No other values are allowed.

**USB Controller Base Address Register****Dev0:[1:0]x10**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:12	<b>BASE[31:12] Base Address.</b> Read-write. These bits specify a 4-kilobyte non-prefetchable space that can be placed anywhere in 32-bit memory.
11:0	Read-only. These bits fixed at their default values.

**USB Subsystem ID and Subsystem Vendor ID Register****Dev0:[1:0]x2C**

Default: 0000\_0000h.

Attribute: Read-only.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> This field is write accessible through Dev0:0x70.
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> This field is write accessible through Dev0:0x70.

**Controller Interrupt Line, Interrupt Pin, Min. Grant, Max Latency Register****Dev0:[1:0]x3C**

Default: 5000\_04FFh.

Attribute: Read-only.

Bits	Description
31:24	<b>MAX LATENCY.</b> Read-only. This register indicates how often the USB controller requires host accesses. The value 50h indicates 20 microseconds.
23:16	<b>MIN GNT.</b> Read-only. These bits are fixed at their default values.
15:8	<b>INTERRUPT PIN.</b> Read-only. The value 04h indicates that USB interrupts are routed through PIRQD_L
7:0	<b>INTERRUPT LINE.</b> Read-write. This field controls no hardware.

**USB Buffer Control Register**

**Dev0:[1:0]x44**

Default: 0000h.

Attribute: Read-write.

Bits	Description
15:9	Reserved.
8	<b>PIPDIS. SIE pipelining disable.</b> Read-write. 1=Transfer descriptors are disabled from being pipelined with USB bus activity.
7:1	Reserved.
0	<b>DB16. Data buffer region 16.</b> Read-write. 1=The size of the region for the data buffer is 16 bytes. 0=The region is 32 bytes.

**USB Device and Subsystem ID Read-Write Register**

**Dev0:[1:0]x70**

Default: 0000\_0000h.

Attribute: Read-write.

Bits	Description
31:16	<b>SSID. Subsystem ID register.</b> The value placed in this register is visible in Dev0:0x2C[31:16].
15:0	<b>SSVENDORID. Subsystem vendor ID register.</b> The value placed in this register is visible in Dev0:0x2C[15:0].

**4.9.1.2 USB Open Host Controller Memory-Mapped Registers**

**4.9.1.2.1 Summary**

For a complete description of these registers, see the OCHI 1.0a specification. The base address is controlled by Dev0:[1:0]x10.

**Table 65. USB OHCI Memory-Mapped Register Summary**

Offset	Register	Offset	Register
00-03	HcRevision	34-37	HcFmInterval
04-07	HcControl	38-3B	HcFrameRemaining
08-0B	HcCommandStatus	3C-3F	HcFmNumber
0C-0F	HcInterruptStatus	40-43	HcPeriodicStart
10-13	HcInterruptEnable	44-47	HcLSThreshold
14-17	HcInterruptDisable	48-4B	HcRhDescriptorA
18-1B	HcHCCA	4C-4F	HcRhDescriptorB
1C-1F	HcPeriodCurrentED	50-53	HcRhStatus
20-23	HcControlHeadED	54-57	HcRhPortStatus[1]
24-27	HcControlCurrentED	58-5B	HcRhPortStatus[2]
28-2B	HcBulkHeadED	5C-5F	HcRhPortStatus[3]
2C-2F	HcBulkCurrentED		
30-33	HcDoneHead	100	HceControl
		104	HceInput
		108	HceOutput
		10C	HceStatus

#### 4.9.1.2.2 Implementation-Specific Items

- *HcRevision[7:0]* is 10h to indicate that it conforms to OHCI 1.0.
- *HcRevision[8]* is 1 to indicate that legacy keyboard & mouse emulation support is present.
- *HcFmInterval\_FSLargestDataPacket* resets to 0.
- *HcRhDescriptorA\_NumberDownstreamPorts* is hardwired to 3.
- *HcRhDescriptorA\_NoPowerSwitching* resets to 1.
- *HcRhDescriptorA\_PowerSwitchingMode* and *HcRhDescriptorA\_OverCurrentProtectionMode* must not be set and reset to 0.
- *HcRhDescriptorA\_PowerOnToPowerGoodTime* resets to 1 (representing 2 ms), and can only hold the values 0 to 3 (0 to 6 ms).
- *HcRhDescriptorA\_NoOverCurrentProtection* resets to 0.
- *HcRhDescriptorB\_DeviceRemovable* resets to 0 and should be set by BIOS if non-removable devices are attached.
- *HcRhDescriptorB\_PortPowerControlMask* resets to 0 and should not be set.
- Because power switching is not implemented, the set and clear power bits in *HcRhStatus* and *HcRhPortStatus[1-3]* are not supported.
- *HceControl* and *HceStatus* reset to 00h.
- *HceInput* and *HceOutput* are not reset.
- *HcDoneHd[31:4]* should not be modified.
- *HcFmInterval\_FSLargestDataPacket* is limited to 14 bits. Bit 15 is read-only 0.
- *HcFmNumber* should not be modified.

## 4.9.2 USB Enhanced Host Controller Registers

### 4.9.2.1 USB Enhanced Host Controller Configuration Registers (Dev0:2xXX)

#### USB Device and Vendor ID

Dev0:2x00

Default: 7463\_1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>DEVID. Device ID.</b> This value is hardwired to 7463h
15:0	<b>VENID. Vendor ID.</b> This is hardwired to AMDID (1022h).

#### USB Status and Command

Dev0:2x04

Default: 0210\_0000h.

Attribute: See below.

Bits	Description
31:27	<b>STAT. PCI Status Bits.</b> Read-only. Hardwired to 0b.
26:25	<b>DEVSEL. DEVSEL timing.</b> Read-only. Hardwired to 01b.
24	<b>MDPED. Master Data Parity Error Detected.</b> Read-only. Hardwired to 0b.
23	<b>FBBC. Fast Back-to-Back Capable.</b> Read-only. Hardwired to 0b.
22	Reserved. Hardwired to 0b.
21	<b>M66EN. 66 MHz Capable.</b> Read-only. Hardwired to 0b.
20	<b>CAPLST. Capabilities List.</b> Read-only. Hardwired to 1b to indicate that there is a Capabilities List (Debug Port).
19:10	Reserved. Hardwired to 000h.
9:3	<b>CMD. PCI Command Bits.</b> Read-only. Hardwired to 00h.
2	<b>BMEN. Bus Master Enable.</b> Read-write. When set to 1b the device is allowed to act as a bus master.
1	<b>MEMEN. Memory Space Enable.</b> Read-write. When set to 1b the device responds to Memory Space accesses.
0	<b>IOEN. I/O Space Enable.</b> Read-only. Hardwired to 0b, i.e., the device never responds to I/O accesses.

**USB Class and Revision ID****Dev0:2x08**

Default: 0C03\_20XXh

Attribute: Read-only

Bits	Description
31:24	<b>BASECLASS. Base Class Code.</b> Hardwired to 0Ch = Serial Bus controller.
23:16	<b>SCC. Sub-Class Code.</b> Hardwired to 03h = Universal Serial Bus Controller.
15:8	<b>PI. Programming Interface.</b> Hardwired to 20h = USB 2.0 Host Controller conforming to EHCI spec.
7:0	<b>REVID. Revision ID.</b> The value of this register is revision-dependent.

**USB BIST, Header, Latency, and Cache Line Size****Dev0:2x0C**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:24	<b>BIST.</b> Read-only. Hardwired to 0b.
23:16	<b>HEADER.</b> Read-only. Hardwired to 00h to indicate a normal device header.
15:8	<b>LATENCY.</b> Read-write. This register controls no internal hardware.
7:0	<b>CLS. Cache Line Size.</b> Read-only. This register controls no internal hardware.

**USB Register Space Base Address****Dev0:2x10**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:8	<b>BAR. Base Address.</b> Read-write. Corresponds to memory address [31:8]. Allows for a register space of 256 bytes.
7:4	Reserved. Hardwired to 0h.
3	<b>PRF. Prefetchable.</b> Read-only. Hardwired to 0b = not prefetchable.
2:1	<b>TPYE. Type.</b> Read-only. Hardwired to 00b = may only be mapped into 32-bit address space.
0	<b>MSI. Memory Space Indicator.</b> Read-only. Hardwired to 0b = memory-mapped.

**USB Subsystem ID and Subsystem Vendor ID****Dev0:2x2C**



Default: 7463\_1022h.

Attribute: Read-only.

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b> This value is writable through Dev0:2x70[31:16].
15:0	<b>SUBVENID. Subvendor ID.</b> This value is writable through Dev0:2x70[15:0].

**USB Capabilities Pointer****Dev0:2x34**

Default: 0000\_0080h.

Attribute: Read-only.

Bits	Description
31:8	Reserved. Hardwired to 000000h.
7:0	<b>CPTR. Capabilities Pointer.</b> Hardwired to 80h, to point to debug port registers.

**USB Latency and Interrupt Control****Dev0:2x3C**

Default: 0000\_04FFh.

Attribute: See below.

Bits	Description
31:24	<b>MAXLAT. Max_Lat.</b> Read-only. Hardwired to 00h to indicate no special requirements for latency timers.
23:16	<b>MINGNT. Min_Gnt.</b> Read-only. Hardwired to 00h to indicate no special requirements for latency timers.
15:8	<b>INTPIN. Interrupt Pin.</b> Read-only. Hardwired to 04h, to indicate usage of PIRQD_L.
7:0	<b>INTLINE. Interrupt Line.</b> Read-write. Only used by SW to determine interrupt routing, not used by EHC hardware.

**USB Legacy Support Extended Capabilities****Dev0:2x40**

Default: 0000\_0001.

Attribute: See below.

Bits	Description
31:25	Reserved. Hardwired to 00h.
24	<b>HCOS. HC OS Owned Semaphore.</b> Read-write. System software sets this bit to request ownership of the EHCI controller. Ownership is obtained when this bit reads as 1 and the HC BIOS Owned Semaphore bit reads as clear. This bit resides in the AUX power domain.
23:17	Reserved. Hardwired to 00h.
16	<b>HCBIOS. HC BIOS Owned Semaphore.</b> Read-write. The BIOS sets this bit to establish ownership of the EHCI controller. System BIOS clears this bit in response to a request for ownership of the EHCI controller by system software. This bit resides in the AUX power domain.
15:8	<b>NEECP. Next EHCI Extended Capability Pointer.</b> Read-only. 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	<b>CAPID. Capability ID.</b> Read-only. 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 co-status information, and this register is located at Dev0:2x44.

This register is used by pre-OS software (BIOS) and the operating system to coordinate ownership of the EHCI host controller.

## USB Legacy Support Control/Status

Dev0:2x44

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31	<b>SMIBAR. SMI on BAR.</b> Read; write 1 to clear. This bit is set to 1b whenever the Base Address Register (BAR) at Dev0:2x10 is written.
30	<b>SMICMD. SMI on PCI Command.</b> Read; write 1 to clear. This bit is set to 1b whenever the PCI Command Register is written.
29	<b>SMIOC. SMI on OS Ownership Change.</b> Read; write 1 to clear. This bit is set to 1b whenever the HC OS Owned Semaphore bit in the Dev0:2x40 register transitions from 1 to a 0 or 0 to a 1.
28:22	Reserved. Hardwired to 00h.
21	<b>SMIAA. SMI on Async Advance.</b> Read-only. Shadow bit of the Interrupt on Async Advance bit in the ECAP34 register.
20	<b>SMIHSE. SMI on Host System Error.</b> Read-only. Shadow bit of Host System Error bit in the ECAP34 register.
19	<b>SMIFLR. SMI on Frame List Rollover.</b> Read-only. Shadow bit of Frame List Rollover bit in the ECAP34 register.
18	<b>SMIPCD. SMI on Port Change Detect.</b> Read-only. Shadow bit of Port Change Detect bit in the ECAP34 register.
17	<b>SMIUERR. SMI on USB Error.</b> Read-only. Shadow bit of USB Error Interrupt (USBERRINT) bit in the ECAP34 register.
16	<b>SMIUC. SMI on USB Complete.</b> Read-only. Shadow bit of USB Interrupt (USBINT) bit in the ECAP34 register.
15	<b>SMIBAREN. SMI on BAR Enable.</b> Read-write. When this bit is 1b and SMI on BAR is 1b, then the host controller issues an SMI. This bit resides in the AUX power domain.
14	<b>SMICMDEN. SMI on PCI Command Enable.</b> Read-write. When this bit is 1b and SMI on PCI Command is 1b, then the host controller issues an SMI. This bit resides in the AUX power domain.
13	<b>SMIOSEN. SMI on OS Ownership Enable.</b> Read-write. When this bit is a one and the OS Ownership Change bit is one, the host controller issues an SMI This bit resides in the AUX power domain.
12:6	Reserved. Hardwired to 00h.
5	<b>SMIAAEN. SMI on Async Advance Enable.</b> Read-write. When this bit is a one, and the SMI on Async Advance bit (above) in this register is a one, the host controller issues an SMI immediately. This bit resides in the AUX power domain.

Bits	Description
4	<p><b>SMIHSEEN. SMI on Host System Error Enable.</b></p> <p>Read-write. When this bit is a one, and the SMI on Host System Error bit (above) in this register is a one, the host controller issues an SMI immediately.</p> <p>This bit resides in the AUX power domain.</p>
3	<p><b>SMIFLREN. SMI on Frame List Rollover Enable.</b></p> <p>Read-write. When this bit is a one, and the SMI on Frame List Rollover bit (above) in this register is a one, the host controller issues an SMI immediately.</p> <p>This bit resides in the AUX power domain.</p>
2	<p><b>SMIPCDEN. SMI on Port Change Enable.</b></p> <p>Read-write. When this bit is a one, and the SMI on Port Change Detect bit (above) in this register is a one, the host controller issues an SMI immediately.</p> <p>This bit resides in the AUX power domain.</p>
1	<p><b>SMIUEEN. SMI on USB Error Enable.</b></p> <p>Read-write. When this bit is a one, and the SMI on USB Error bit (above) in this register is a one, the host controller issues an SMI immediately.</p> <p>This bit resides in the AUX power domain.</p>
0	<p><b>USMIEN. USB SMI Enable.</b></p> <p>Read-write. When this bit is a one, and the SMI on USB Complete bit (above) in this register is a one, the host controller issues an SMI immediately.</p> <p>This bit resides in the AUX power domain.</p>

Pre-OS (BIOS) software uses this register to enable SMIs for every EHCI/USB event it needs to track. Bits [21:16] of this register are simply shadow bit of ECAP34 register [5:0].

**USB Port Wake Capabilities, Frame Length Adjustment, and Serial Bus Release Number**

**Dev0:2x60**

Default: 0001\_2020h.

Attribute: See below.

Bits	Description
31:16	<b>PORTWAKECAP. Port Wake Up Capability Mask.</b> Read-write. A 1b in bit position [16] indicates that the register is implemented. Bit positions [17] through [31] correspond to a physical port. These bits reside in the AUX power domain.
15:14	Reserved bits. Hardwired to 0b.
13:8	<b>FLADJ. Frame Length Adjustment Register.</b> Read-write. The SOF cycle time is equal to 59488 + (16 * this value). The default of 20h gives a SOF cycle time of 60000. Frame Length (HS bit times)    FLADJ value 59488            00h 59504            01h 59520            02h ...                ... 59984            1Fh 60000            20h ...                ... 60480            3Eh 60496            3Fh These bits reside in the AUX power domain.
7:0	<b>Serial Bus Specification Release Number (SBRN).</b> Read-only. Hardwired to 20h = release 2.0

**USB Subsystem ID and Subsystem vendor ID RW Alias**

**Dev0:2x70**

Default: 7463\_1022h.

Attribute: Read-write.

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b> This value is read-only at Dev0:2x2C[31:16]. Reset default: 7463h
15:0	<b>SUBVENID. Subvendor ID.</b> This value is read-only at Dev0:2x2C[15:0]. Reset default: 1022h

**USB Debug Port Capability Register**

**Dev0:2x80**

Default: 20C0\_880Ah.

Attribute: Read-only.

Bits	Description
31:29	<b>DPBAR. Debug Port BAR.</b> A 3-bit field, which indicates which one of the possible 6 Base Address Register offsets contains the Debug Port registers. Hardwired to 1h to indicate the use of the BAR at Dev0:2x14
28:16	<b>DPOFF. Debug Port Offset.</b> Hardwired to C0h
15:8	<b>NXT_PTR. Next Pointer.</b> Pointer to the next item in the capabilities list. Hardwired to 88h (due to Power Management Capability).
7:0	<b>CAP_ID.</b> The value of 0Ah in this field identifies that the function supports a Debug Port.

**USB Power Management Capability Register****Dev0:2x88**

Default: FE82\_0001h.

Attribute: Read-only.

Bits	Description
31:27	<b>PMECAP. PME_Support.</b> This 5-bit field indicates the power states in which the function may assert PME_L. A value of 0b for any bit indicates that the function is not capable of asserting the PME_L signal while in that power state.
26	<b>D2CAP. D2 Support.</b> If this bit is a 1, this function supports the D2 Power Management State.
25	<b>D1CAP. D1 Support.</b> If this bit is a 1, this function supports the D1 Power Management State.
24:22	<b>AUXCUR. Aux_Current.</b> This 3 bit field reports the 3.3Vaux auxiliary current requirements for the PCI function. [24:22] Max. Current Required 1 1 1 375 mA 1 1 0 320 mA 1 0 1 270 mA 1 0 0 220 mA 0 1 1 160 mA 0 1 0 100 mA 0 0 1 55 mA 0 0 0 0 (self powered)
21	<b>DSI.</b> The Device Specific Initialization bit indicates whether special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it.
20	Reserved. Hardwired to 0b.

Bits	Description
19	<b>PMECLK. PME Clock.</b> When this bit is a 1, it indicates that the function relies on the presence of the PCI clock for PME_L operation. When this bit is a 0, it indicates that no PCI clock is required for the function to generate PME_L.
18:16	<b>VER. Version.</b> A value of 010b indicates that this function complies with Revision 1.1 of the PCI Power Management Interface Specification.
15:8	<b>NXT_PTR.Next pointer.</b> Pointer to the next item in the capabilities list. Hardwired to 00h.
7:0	<b>CAP_ID.Capability ID.</b> The value of 01h identifies the linked list item as being the PCI Power Management registers.

**USB Power Management Control/Status & Data Register****Dev0:2x8C**

Default: 0000\_0100h.

Attribute: See below.

Bits	Description
31:24	<b>DATA. Data.</b> Read-only. This register is used to report the state dependent data requested by the DATA_SELECT field.
23:16	<b>P2PCAP. PCI to PCI Bridge Support Extensions.</b> Read-only. Not used, hardwired to 00h.
15	<b>PMESTAT. PME_Status.</b> Read; write 1 to clear. This bit is set when the function would normally assert the PME_L signal independent of the state of the PME_EN bit. This bit resides in the AUX power domain.
14:13	<b>DATA_SCALE. Data_Scale.</b> Read-only. This 2-bit read-only field indicates the scaling factor to be used when interpreting the value of the DATA register. Not used, hardwired to 00b.
12:9	<b>DATA_SELECT. Data_Select.</b> Read-only. This 4-bit field is used to select which data is to be reported through the Data register and DATA_SCALE field. Not used, hardwired to 0h.
8	<b>PME_EN.</b> An 1b enables the function to assert PME_L. When 0b, PME_L assertion is disabled. This bit resides in the AUX power domain.
7:2	Reserved. Hardwired to 00h.
1:0	<b>PWRSTAT. PowerState.</b> This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below. 00b            D0 01b            D1 10b            D2 11b            D3hot These bits reside in the AUX power domain.

**USB Remote Wake-up Mode****Dev0:2xA0**



default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:1	Reserved. Hardwired to 00000000h.
0	<p><b>RWM. Remote Wake-up Mode.</b></p> <p>Read-write. This bit controls the behavior of the host controller when not in power state D0. When set to 1b the host controller immediately drives resume on all its suspended ports. When cleared to 0b the host controller only latches the wake-up event in the respective port status register (ECAP[88:74]) and postpones the resume until the system is back to D0 (S0).</p> <p>This bit resides in the AUX power domain.</p>

#### 4.9.2.1.1 Memory-mapped Registers

#### 4.9.2.1.2 Host Controller Capability Registers

These registers are located in memory-mapped IO space. The base address register for these registers is Dev0:2

#### 4.9.2.2 USB Enhanced Host Controller Memory-Mapped Registers

##### USB HC Version and Capability Length

ECAP00

Default: 0100\_0030h.

Attribute: Read-only.

Bits	Description
31:16	<b>HCVERSION. Interface Version Number.</b> Default = 0100h = Rev. 0.96 in BCD.
15:8	Reserved. Hardwired to 00h.
7:0	<b>CAPLENGTH. Capability Registers Length.</b> Offset to add to Dev0:2x10 to find operational registers.

##### USB HC Structural Parameters

ECAP04

Default: 0010\_2306h.

Attribute: Read-only.

Bits	Description
31:24	Reserved. Hardwired to 00h.
23:20	<b>DPNUM. Debug Port Number.</b> 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.
19:17	Reserved. Hardwired to 000b.
16	<b>P_INDICATOR. Port Indicators.</b> This bit indicates whether the ports support port indicator control. Hardwired to 0b, since there is no such support.
15:12	<b>N_CC. Number of Companion Controller.</b> This field indicates the number of companion controllers associated with this USB 2.0 host controller. Hardwired to 2h.
11:8	<b>N_PCC. Number of Ports per Companion Controller.</b> This field indicates the number of ports supported per companion host controller. It is used to indicate the port routing configuration to system software. Hardwired to 3h.

Bits	Description
7	<b>PRR. Port Routing Rules.</b> This field indicates the method used by this implementation for how all ports are mapped to companion controllers. Hardwired to 0b = 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.
6:5	Reserved. Hardwired to 00b.
4	<b>PPC. Port Power Control.</b> This field indicates whether the host controller implementation includes port power control. Hardwired to 0b, since there is no power control.
3:0	<b>N_PORTS.Number of Ports.</b> 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. Hardwired to 6h.

**USB HC Capability Parameters****ECAP08**

Default: 0000\_4082h.

Attribute: Read-only.

Bits	Description
31:16	Reserved. Hardwired to 0b.
15:8	<b>EECP. EHCI Extended Capabilities Pointer.</b> 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 40h or greater if implemented to maintain the consistency of the PCI header defined for this class of device.
7:4	<b>IST. Isochronous Scheduling Threshold.</b> 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	Reserved. Hardwired to 0b.
2	<b>ASPC. Asynchronous Schedule Park Capability.</b> Hardwired to 0b = the host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule. The 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 ECAP30 register.
1	<b>PFLF. Programmable Frame List Flag.</b> Hardwired to 1b = system software can specify and use a smaller frame list and configure the host controller through the ECAP30 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.
0	<b>AC64. 64-bit Addressing Capability.</b> Hardwired to 0b = data structures using 32-bit address memory pointers.

**USB Command Register****ECAP30**

Default: 0008\_0000h.

Attribute: See below.

Bits	Description																		
31:24	Reserved. Hardwired to 00h.																		
23:16	<p><b>ITC. Interrupt Threshold Control.</b></p> <p>Read-write. This field is used by system software to select the maximum rate at which the host controller issues interrupts. The only valid values are defined below. If software writes an invalid value to this register, the results are undefined.</p> <table> <tr> <td>Value</td> <td>Maximum Interrupt Interval</td> </tr> <tr> <td>00h</td> <td>Reserved</td> </tr> <tr> <td>01h</td> <td>1 micro-frame</td> </tr> <tr> <td>02h</td> <td>2 micro-frames</td> </tr> <tr> <td>04h</td> <td>4 micro-frames</td> </tr> <tr> <td>08h</td> <td>8 micro-frames (default, equates to 1 ms)</td> </tr> <tr> <td>10h</td> <td>16 micro-frames (2 ms)</td> </tr> <tr> <td>20h</td> <td>32 micro-frames (4 ms)</td> </tr> <tr> <td>40h</td> <td>64 micro-frames (8 ms)</td> </tr> </table> <p>Software modifications to this bit while HCHalted bit is equal to zero results in undefined behavior.</p>	Value	Maximum Interrupt Interval	00h	Reserved	01h	1 micro-frame	02h	2 micro-frames	04h	4 micro-frames	08h	8 micro-frames (default, equates to 1 ms)	10h	16 micro-frames (2 ms)	20h	32 micro-frames (4 ms)	40h	64 micro-frames (8 ms)
Value	Maximum Interrupt Interval																		
00h	Reserved																		
01h	1 micro-frame																		
02h	2 micro-frames																		
04h	4 micro-frames																		
08h	8 micro-frames (default, equates to 1 ms)																		
10h	16 micro-frames (2 ms)																		
20h	32 micro-frames (4 ms)																		
40h	64 micro-frames (8 ms)																		
15:12	Reserved. Hardwired to 0000b.																		
11	<p><b>ASPME. Asynchronous Schedule Park Mode Enable.</b></p> <p>Read-write. Hard wired to =b, 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.</p>																		
10	Reserved. Hardwired to 0b.																		
9:8	<p><b>ASPMC. Asynchronous Schedule Park Mode Count.</b></p> <p>Read-write. Hard wired to 0h, It contains a count of the number of successive transactions the host controller will execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. See EHCI Spec, section 4.10.3.2 for full operational details. Valid values are 1h to 3h. Software must not write a zero to this bit when Park Mode Enable is a one as this results in undefined behavior.</p>																		
7	<p><b>LITE_HCRESET. Light Host Controller Reset.</b></p> <p>Read-write. It allows the driver to reset the EHCI controller without affecting the state of the ports or the relationship to the companion host controllers. 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.</p>																		
6	<p><b>IAAD. Interrupt on Async Advance Doorbell.</b></p> <p>Read-write. 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 1b to this bit to ring the doorbell.</p>																		
5	<p><b>ASE. Asynchronous Schedule Enable.</b></p> <p>Read-write. This bit controls whether the host controller skips processing the Asynchronous Schedule. 0= Do not process the Asynchronous Schedule. 1= Use the ECAP48 register to access the Asynchronous Schedule.</p>																		

Bits	Description
4	<p><b>PSE. Periodic Schedule Enable.</b></p> <p>Read-write. This bit controls whether the host controller skips processing the Periodic Schedule. 0= Do not process the Periodic Schedule. 1= Use the ECAP44 register to access the Periodic Schedule.</p>
3:2	<p><b>FLS. Frame List Size.</b></p> <p>Read-write. 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.</p> <p>00b            1024 elements (4096 bytes) Default value  01b            512 elements (2048 bytes)  10b            256 elements (1024 bytes) for resource-constrained environments  11b            Reserved</p>
1	<p><b>HCRESET. Host Controller Reset.</b></p> <p>Read-write. This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset.</p> <p>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.</p> <p>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). Software must reinitialize the host controller in order to return the host controller to an operational state.</p> <p>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.</p> <p>Software should not set this bit to a one when the HCHalted bit in the ECAP34 register is a zero. Attempting to reset an actively running host controller results in undefined behavior.</p>
0	<p><b>RS. Run/Stop.</b></p> <p>Read-write. 1= Run. 0= Stop. When set to a 1b, the Host Controller proceeds with execution of the schedule. The Host Controller continues execution as long as this bit is set to a 1b. When this bit is cleared to 0b, the Host Controller completes the current and any actively pipelined transactions on the USB and then halts. The Host Controller must halt within 16 microframes 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 should not write a one to this field unless the host controller is in the Halted state (i.e., HCHalted in the ECAP34 register is a one).</p>

**USB Status Register****ECAP34**

Default: 0000\_1000h.

Attribute: See below.

Bits	Description
31:16	Reserved. Hardwired to 0000h.
15	<b>ASS. Asynchronous Schedule Status.</b> Read-only. 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.
14	<b>PSS. Periodic Schedule Status.</b> Read-only. 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.
13	<b>REC. Reclamation.</b> This is a read-only status bit, which is used to detect an empty asynchronous schedule.
12	<b>HCH. HCHalted.</b> Read-only. 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 cleared to 0b, either by software or by the Host Controller hardware (e.g., internal error).
11:6	Reserved. Hardwired to 00h.
5	<b>IAA. Interrupt on Async Advance.</b> Read; write 1 to clear. 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 ECAP30 register. This status bit indicates the assertion of that interrupt source.
4	<b>HSE. Host System Error.</b> Read; write 1 to clear. The Host Controller sets this bit to 1b when a serious error occurs during a host system access involving the Host Controller module. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs.

Bits	Description
3	<p><b>FLR. Frame List Rollover.</b> Read; write 1 to clear. 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.</p>
2	<p><b>PCD. Port Change Detect.</b> Read; write 1 to clear. 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 is also 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 zero to a port's Port Owner bit.</p>
1	<p><b>USBERRINT. USB Error Interrupt.</b> Read; write 1 to clear. The Host Controller sets this bit to 1b 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.</p>
0	<p><b>USBINT. USB Interrupt.</b> Read; write 1 to clear. The Host Controller sets this bit to 1b 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 1b when a short packet is detected (actual number of bytes received was less than the expected number of bytes).</p>



**USB Interrupt Enable Register**

**ECAP38**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:6	Reserved. Hardwired to 0000000h.
5	<b>IAAE. Interrupt on Async Advance Enable.</b> Read-write. When this bit is a one, and the Interrupt on Async Advance bit in the ECAP34 register is a one, the host controller issues an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.
4	<b>HSEE. Host System Error Enable.</b> Read-write. When this bit is a one, and the Host System Error Status bit in the ECAP34 register is a one, the host controller issues an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.
3	<b>FLRE. Frame List Rollover Enable.</b> Read-write. When this bit is a one, and the Frame List Rollover bit in the ECAP34 register is a one, the host controller issues an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit.
2	<b>PCIE. Port Change Interrupt Enable.</b> Read-write. When this bit is a one, and the Port Change Detect bit in the ECAP34 register is a one, the host controller issues an interrupt. The interrupt is acknowledged by software clearing the Port Change Detect bit.
1	<b>UEIE. USB Error Interrupt Enable.</b> Read-write. When this bit is a one, and the USBERRINT bit in the ECAP34 register is a one, the host controller issues an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBERRINT bit.
0	<b>UIE. USB Interrupt Enable.</b> Read-write. When this bit is a one, and the USBINT bit in the ECAP34 register is a one, the host controller issues an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBINT bit.

**USB Frame Index Register**

**ECAP3C**

Default: 0000\_0000h.

Attribute: See below (only doubleword access supported).

Bits	Description
31:14	Reserved. Hardwired to 0000h.
13:0	<b>FIND. Frame Index.</b> Read-write. The value in 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. ECAP30[FLS] Number ElementsN 00b (1024) 12 01b (512) 11 10b (256) 10 11b Reserved

This register cannot be written unless the Host Controller is in the Halted state as indicated by the *HCHalted* bit (ECAP34). A write to this register while the *Run/Stop* bit is set to a one (ECAP30) produces undefined results. Writes to this register also affect the SOF value.

### USB Control Data Structure Segment Register

ECAP40

Default: 0000\_0000h.

Attribute: Read-only.

Bits	Description
31:0	Hardwired to 00000000h, since no 64-bit capability implemented (see ECAP08).

### USB Periodic Frame List Base Address Register

ECAP44

Default: 0000\_0000h.

Attribute: See below (only doubleword access supported).

Bits	Description
31:12	<b>BAR. Base Address.</b> Read-write. These bits correspond to memory address signals [31:12], respectively.
11:0	Reserved. Hardwired to 000h.

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 (ECAP3C) to enable the Host Controller to step through the Periodic Frame List in sequence.

### USB Current Asynchronous List Address Register

ECAP48

Default: 0000\_0000h.

Attribute: See below (only doubleword access supported).

Bits	Description
31:5	<b>LPL. Link Pointer Low.</b> Read-write. These bits correspond to memory address signals [31:5], respectively. This field may only reference a Queue Head (QH).
4:0	Reserved. Hardwired to 00h.

The memory structure referenced by this physical memory pointer is assumed to be 32-byte aligned.

**USB Configure Flag Register****ECAP70**

Default: 0000\_0000h.

Attribute: See below.

Bits	Description
31:1	Reserved. Hardwired to 00000000h.
0	<p><b>CF. Configure Flag.</b></p> <p>Read-write. 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.</p> <p>0b                    Port routing control logic default-routes each port to an implementation dependent classic host controller.</p> <p>1b                    Port routing control logic default-routes all ports to this host controller.</p> <p>This bit resides in the AUX power domain.</p>

**USB Port Status and Control Register [6:1]****ECAP[88:74]**

Default: 0000\_3000h.

Attribute: See below.

Bits	Description														
31:23	Reserved. Hardwired to 000h.														
22	<p><b>WKOC_E. Wake on Over-current Enable.</b> Read-write. Writing this bit to a one enables the port to be sensitive to over-current conditions as wake-up events</p> <p>This bit resides in the AUX power domain.</p>														
21	<p><b>WKDSCNNT_E. Wake on Disconnect Enable.</b></p> <p>Read-write. Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events.</p> <p>This bit resides in the AUX power domain.</p>														
20	<p><b>WKCNT_E. Wake on Connect Enable.</b></p> <p>Read-write. Writing this bit to a one enables the port to be sensitive to device connects as wake-up events.</p> <p>This bit resides in the AUX power domain.</p>														
19:16	<p><b>PTC. Port Test Control.</b></p> <p>Read-write. 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 (0110b–1111b are reserved):</p> <table> <tr> <td>Bits</td> <td>Test Mode</td> </tr> <tr> <td>0000b</td> <td>Test mode not enabled</td> </tr> <tr> <td>0001b</td> <td>Test J_STATE</td> </tr> <tr> <td>0010b</td> <td>Test K_STATE</td> </tr> <tr> <td>0011b</td> <td>Test SE0_NAK</td> </tr> <tr> <td>0100b</td> <td>Test Packet</td> </tr> <tr> <td>0101b</td> <td>Test FORCE_ENABLE</td> </tr> </table> <p>Refer the USB Specification Revision 2.0, Chapter 7 for details on each test mode.</p>	Bits	Test Mode	0000b	Test mode not enabled	0001b	Test J_STATE	0010b	Test K_STATE	0011b	Test SE0_NAK	0100b	Test Packet	0101b	Test FORCE_ENABLE
Bits	Test Mode														
0000b	Test mode not enabled														
0001b	Test J_STATE														
0010b	Test K_STATE														
0011b	Test SE0_NAK														
0100b	Test Packet														
0101b	Test FORCE_ENABLE														
15:14	<p><b>PIC. Port Indicator Control.</b></p> <p>Read-only. Hardwired to 00b, since there is no support for this (see ECAP04).</p>														

Bits	Description															
13	<p><b>POWN. Port Owner.</b></p> <p>Read-write. This bit unconditionally goes to a 0b when the Configured bit in the ECAP70 register makes a 0b to 1b transition. This bit unconditionally goes to 1b 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.</p> <p>This bit resides in the AUX power domain.</p>															
12	<p><b>PP. Port Power.</b> Read-only. Hardwired to 1b, since the host controller does not have port power control switches. Each port is hardwired to power.</p>															
11:10	<p><b>LSTAT. Line Status.</b></p> <p>Read-only. 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:</p> <table border="0" data-bbox="266 821 1448 982"> <tr> <td>Bits[11:10]</td> <td>USB State</td> <td>Interpretation</td> </tr> <tr> <td>00b</td> <td>SE0</td> <td>Not Low-speed device, perform EHCI reset</td> </tr> <tr> <td>10b</td> <td>J-state</td> <td>Not Low-speed device, perform EHCI reset</td> </tr> <tr> <td>01b</td> <td>K-state</td> <td>Low-speed device, release ownership of port</td> </tr> <tr> <td>11b</td> <td>Undefined</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> </table>	Bits[11:10]	USB State	Interpretation	00b	SE0	Not Low-speed device, perform EHCI reset	10b	J-state	Not Low-speed device, perform EHCI reset	01b	K-state	Low-speed device, release ownership of port	11b	Undefined	Not Low-speed device, perform EHCI reset.
Bits[11:10]	USB State	Interpretation														
00b	SE0	Not Low-speed device, perform EHCI reset														
10b	J-state	Not Low-speed device, perform EHCI reset														
01b	K-state	Low-speed device, release ownership of port														
11b	Undefined	Not Low-speed device, perform EHCI reset.														
9	<p>Reserved. Hardwired to 0b.</p>															
8	<p><b>PRES. Port Reset.</b></p> <p>Read-write. 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. Note: when software writes this bit to a one, it must also write a zero to the Port Enable bit.</p> <p>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 does 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 automatically enables this port (e.g., sets the Port Enable bit to a one). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from a one to a zero.</p> <p>The HCHalted bit in the ECAP34 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.</p> <p>This bit resides in the AUX power domain.</p>															

Bits	Description
7	<p><b>SUSP. Suspend.</b></p> <p>Read-write. 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:</p> <p>Bits [Port Enabled, Suspend]Port State</p> <p>0X            Disable</p> <p>10            Enable</p> <p>11            Suspend</p> <p>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.</p> <p>A write of zero to this bit is ignored by the host controller. The host controller unconditionally sets this bit to a zero when:</p> <ul style="list-style-type: none"> <li>• Software sets the Force Port Resume bit to a zero (from a one).</li> <li>• Software sets the Port Reset bit to a one (from a zero).</li> </ul> <p>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.</p> <p>This bit resides in the AUX power domain.</p>
6	<p><b>FPR. Force Port Resume.</b></p> <p>Read-write. 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.</p> <p>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 ECAP34 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.</p> <p>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 remains 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.</p> <p>This bit resides in the AUX power domain.</p>
5	<p><b>OCC. Over-current Change.</b></p> <p>Read; write 1 to clear. This bit gets set to a one when there is a change to Over-current Active.</p> <p>This bit resides in the AUX power domain.</p>
4	<p><b>OCA. Over-current Active.</b></p> <p>Read-only. 1= This port currently has an over-current condition. 0= This port does not have an over-current condition. This bit automatically transitions from a one to a zero when the over current condition is removed.</p>

Bits	Description
3	<p><b>PEC. Port Enable/Disable Change.</b></p> <p>Read; write 1 to clear. 1= Port enabled/disabled status has changed. 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 (See Chapter 11 of the USB Specification for the definition of a Port Error).</p>
2	<p><b>PEN. Port Enabled/Disabled.</b></p> <p>Read-write. 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 only sets this bit to a one when the reset sequence determines that the attached device is a high-speed device.</p> <p>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 (0b) downstream propagation of data is blocked on this port, except for reset.</p> <p>This bit resides in the AUX power domain.</p>
1	<p><b>CSC. Connect Status Change.</b></p> <p>Read; write 1 to clear. 1= Change in Current Connect Status. 0= No change. Indicates a change has occurred in the Current Connect Status of the port. 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.</p>
0	<p><b>CCS. Current Connect Status.</b></p> <p>Read-only. 1= Device is present on port. 0= No device is present. Default = 0. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change bit to be set. This bit resides in the AUX power domain.</p>

#### 4.9.2.2.1 Debug Port Registers

These registers are located in memory-mapped IO space. The base address register for these registers is Dev0:2x14.

## USB Debug Port Control Register

DBG00

Default: 0000\_0000h.

Attribute: See below.

Bits	Description										
31	Reserved. Hardwired to 0b.										
30	<b>OWN. Owner.</b> Read-write. When debug software writes a one to this bit, the ownership of the debug port is forced to the EHCI controller (i.e., immediately taken away from the companion controller). If the port was already owned by the EHCI controller, then setting this bit is has no effect. This bit overrides all of the ownership related bits in the standard EHCI registers. Note that the value in this bit may not effect the value reported in the Port Owner bit in the associated ECAP[88:74] register.										
29	Reserved. Hardwired to 0b.										
28	<b>ENA. Enabled.</b> Read-write. This bit is a one if the debug port is enabled for operation. Software can clear this by writing a zero to it. The controller clears the bit for the same conditions where hardware clears the Port Enable/Disable Change bit (in the ECAP[88:74] register). (Note: this bit is not cleared when System Software clears the Port Enabled/Disabled bit (in the ECAP[88:74] register). Software can directly set this bit, if the port is already enabled in the associated ECAP[88:74] register (this is HW enforced).										
27:17	Reserved. Hardwired to 000b.										
16	<b>DONE. Done.</b> Read; write 1 to clear. This bit is set by HW to indicate that the request is complete. Writing a 1 to this bit clears it. Writing a 0 to this bit has no effect.										
15:11	Reserved. Hardwired to 0h.										
10	<b>USE. In Use.</b> Read-write. Set by software to indicate that the port is in use. Cleared by software to indicate that the port is free and may be used by other software. (This bit has no affect on hardware.)										
9:7	<b>EXCP. Exception.</b> Read-only. This field indicates the exception when Error/Good_L is set. This field cannot be cleared by software. <table border="0"> <tr> <td>Value</td> <td>Meaning</td> </tr> <tr> <td>000b</td> <td>None</td> </tr> <tr> <td>001b</td> <td>Transaction error: indicates the USB2 transaction had an error (CRC, bad PID, time-out, etc.)</td> </tr> <tr> <td>010b</td> <td>HW error. Request was attempted (or in progress) when port was suspended or reset.</td> </tr> <tr> <td>011b-111b</td> <td>Reserved</td> </tr> </table>	Value	Meaning	000b	None	001b	Transaction error: indicates the USB2 transaction had an error (CRC, bad PID, time-out, etc.)	010b	HW error. Request was attempted (or in progress) when port was suspended or reset.	011b-111b	Reserved
Value	Meaning										
000b	None										
001b	Transaction error: indicates the USB2 transaction had an error (CRC, bad PID, time-out, etc.)										
010b	HW error. Request was attempted (or in progress) when port was suspended or reset.										
011b-111b	Reserved										

Bits	Description
6	<b>EGD. Error/Good_L.</b> Read-only. Updated by hardware at the same time it sets the Done bit. When set it indicates that an error occurred. Details of the error are provided in the Exception field. When cleared, it indicates that the request terminated successfully.
5	<b>GO.</b> Read-write. Software sets this bit to cause the hardware to perform a request. Writing this bit to a 1 when the bit is already set may result in undefined behavior. Writing a 0 to this bit has no effect. When set, the hardware clears this bit when the hardware sets the Done bit. (Completion of a request is indicated by the Done bit.)
4	<b>WRD. Write-read_L.</b> Read-write. Software sets this bit to indicate that the current request is a write and clears it to indicate a read.
3:0	<b>DLEN. Data Length.</b> Read-write. For write operations, this field is set by software to indicate to the hardware how many bytes of data in Data Buffer are to be transferred to the console when Write/Read_L is set when software sets Go. A value of 0h indicates that a zero-length packet should be sent. A value of 1–8 indicates 1–8 bytes are to be transferred. Values 9–Fh are illegal and how hardware behaves if used is undefined. For read operations, this field is set by hardware to indicate to software how many bytes in Data Buffer are valid in response to software setting Go when Write/Read_L is cleared. A value of 0h indicates that a zero length packet was returned. (The state of Data Buffer is not defined.) A value of 1–8 indicates 1–8 bytes were received. Hardware is not allowed to return values 9–Fh. The transferring of data always starts with byte 0 in the data area and moves toward byte 7 until the transfer size is reached.



**USB Debug Port PIDs Register****DBG04**

Default: 0000\_C3E1h.

Attribute: See below.

Bits	Description
31:24	Reserved. Hardwired to 00h.
23:16	<b>RPID. Received PID.</b> Read-only. The debug port controller updates this field with the received PID for transactions in either direction. When the controller is sending data (Write/Read_L is asserted), this field is updated with the handshake PID that is received from the device. When the host controller is receiving data (Write/Read_L is not asserted), this field is updated with the data packet PID (if the device sent data), or the handshake PID (if the device NAKs the request). This field is valid when the controller sets the Done bit.
15:8	<b>SPID. Send PID.</b> Read-write. The debug port controller sends this PID to begin the data packet when sending data to USB (i.e., Write/Read_L is asserted). Software typically sets this field to either DATA0 or DATA1 PID values.
7:0	<b>TPID. Token PID.</b> Read-write. The debug port controller sends this PID as the Token PID for each USB transaction. Software typically sets this field to either IN, OUT or SETUP PID values.

**USB Debug Port Data Buffer Low****DBG08**

Default: 0000\_0000h.

Attribute: Read-write.

Bits	Description
31:0	<b>DBL. Data Buffer Low.</b> First 4 bytes of data buffer. The least significant byte is accessed at offset 08h. Each byte in Data Buffer can be individually accessed.

**USB Debug Port Data Buffer High****DBG0C**

Default: 0000\_0000h.

Attribute: Read-write.

Bits	Description
31:0	<b>DBH. Data Buffer High.</b> Second 4 bytes of data buffer. Each byte in Data Buffer can be individually accessed.

Data Buffer must be written with data before software initiates a write request. For a read request, Data Buffer contains valid data when *Done* is set, *Error/Good\_L* is cleared, and *Data Length* specifies the number of bytes that are valid.

**USB Debug Port Device Address Register****DBG10**

Default: 0000\_7F01h.

Attribute: See below.

Bits	Description
31:15	Reserved. Hardwired to 00000h.
14:8	<b>UADR. USB Address.</b> Read-write. 7-bit field that identifies the USB device address used by the controller for all Token PID generation.
7:4	Reserved. Hardwired to 0h.
3:0	<b>UEP. USB Endpoint.</b> Read-write. 4-bit field that identifies the endpoint used by the controller for all Token PID generation.

## 4.10 LAN Ethernet Controller Registers

### 4.10.1 LAN Ethernet Controller User Accessible Registers

The Network Controller requires 4K bytes of memory address space for access to all the various internal registers.

The network controller has three types of user accessible storage—the PCI configuration registers, the memory-mapped registers, and the internal RAMs.

The internal RAMs include the MIB counters, the Pattern Match RAM, and the Autopoll Value RAM. User access to these memories is indirect with a protocol similar to the external PHY access protocol. Associated with each of these memories is an xxx\_ADDR and an xxx\_DATA register, where xxx stands for MIB, PMR, or AP\_VAL. The xxx\_ADDR register contains the address of the memory element to be accessed plus a WR\_CMD bit, a RD\_CMD bit, and a CMD\_ACTIVE bit. The CMD\_ACTIVE bit is a read-only bit that is automatically set when the WR\_CMD or RD\_CMD bit is set and is automatically cleared when the operation is complete. The procedure for writing to one of these internal memories is:

1. Poll the appropriate CMD\_ACTIVE bit until this bit is 0.
2. Write the data into xxx\_DATA.
3. Write the address of the element to be accessed to xxx\_ADDR, with the WR\_CMD bit set.

The procedure for reading from one of these internal memories is:

1. Poll the appropriate CMD\_ACTIVE bit until this bit is 0.
2. Write the address of the element to be accessed to xxx\_ADDR, with the RD\_CMD bit set.
3. Poll the appropriate CMD\_ACTIVE bit until this bit is 0.

#### 4. Read the data from xxx\_DATA.

The PCI configuration registers and the memory-mapped registers can be accessed in any data width up to 32 bits.

The controller's registers can be divided into several functional groups: PCI Configuration Alias registers, PCI Configuration registers, Setup registers, Running registers, and Test registers.

The PCI Configuration Alias registers are typically programmed by the chip set initialization software. In this way, they are initialized before the BIOS accesses the PCI Configuration registers. This group includes the Vendor ID Alias, Device ID Alias, Sub Vendor ID Alias, Sub System ID Alias, MIN\_GNT Alias, MAX\_LAT Alias, and PMC Alias registers at offsets C0h to DFh in PCI Configuration Space.

The PCI Configuration registers are accessed by the system BIOS software to configure the network controller. These registers include the Memory Base Address register, the Interrupt Line register, the PCI Command register, the PCI Status register and the PMC register. Typically, device information is also read from the SID, SVID and VID registers.

The Setup registers include most of the remaining memory-mapped registers. The programming of these is typically divided between the chip set initialization software and the driver software. Many of these registers are optional and need not be initialized unless the associated function is used.

Typically, the SRAM\_SIZE, SRAM\_BND and PADR registers are initialized by the chip set initialization software. The driver initializes the BADR, BADX, RCV\_RING\_LEN, XMT\_RING\_LEN and LADRF registers. The CMD2, CMD3, CTRL1 and CTRL2 registers are typically partially initialized by the chip set initialization software and partially by the driver. The CMD7 register is initialized by the driver. The driver reads the CHIPID register at initialization time. The PHY\_ACCESS register may be used at this time to initialize the external PHY.

Running registers are accessed by the driver software. These include the CMD0, INT0, INTEN0, STAT0, LADRF and MIB registers. The driver may access other registers during operation depending on the features being used.

## 4.10.2 LAN Ethernet Controller Configuration Registers (Dev1:0xXX)

### LAN Ethernet Controller Vendor and Device Id Dev1:0x00

Default: 7462\_1022h

Attribute: Read-only

Bits	Description
31:16	<b>ENCID.</b> Provides the LAN Ethernet controller device identification.
15:0	<b>AMDID.</b> Provides AMD's PCI vendor identification.

### LAN Ethernet Controller Command and Status Dev1:0x04

Default: 0210\_0000h

Attribute: see below field.

Bits	Description
31:16	<b>STAT.</b> Read-only.
15:0	<b>CMD.</b> CMD[15:3]: Read-only. CMD[2]: <b>BMEN. Bus master enable.</b> Read-write. 1 = Enables bus master capability. The host must set BMEN before setting the RUN bit in CMD0 of the Ethernet controller. 0 = Disables bus master capability. BMEN is reset by H_RESET. CMD[1]: <b>MEMEN. Memory space enable.</b> Read-write. 1 = Access to memory space for this device is enabled. 0 = Access to memory space for this device is disabled. MEMEN is reset by H_RESET. CMD[0]: Reserved.

### LAN Ethernet Controller Revision Id and Class Code Dev1:0x08

Default: 0200\_00XXh see below.

Attribute: Read-only

Bits	Description
31:24	<b>BASECLASS.</b> These bits are fixed at 02h indicating a network controller.
23:16	<b>SUBCLASS.</b> These bits are fixed at 00h indicating an Ethernet controller.
15:8	<b>PROGIF.</b>
7:0	<b>REVID. LAN Ethernet Controller silicon revision.</b> The value of this register is revision-dependent.

### LAN Ethernet Controller BIST, Header and Latency Dev1:0x0C

Default: 0000\_0010h

Attribute: see below.

Bits	Description
------	-------------

31:24	<b>BIST.</b> Read-only.
23:16	<b>HEADER.</b> Read-only.
15:8	<b>LATENCY.</b> Read-write. The latency timer defines the minimum amount of time, in PCI clock cycles, that the bus master can retain ownership of the bus. The two least significant bits are fixed to 0b.
7:0	Read-only.

**LAN Ethernet Controller Command Interface Base Address****Dev1:0x10**

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31:12	<b>MEMBASE. Memory-mapped register base address.</b> Read-write. These bits are used in the memory space decode of the memory-mapped registers. There is a maximum block size of 4k bytes for this base address. This register is reset by H_RESET.
11:1	Read-only.
0	<b>RTE. Resource Type Indicator.</b> This bit is fixed to 0b, indicating memory address space.

**LAN Ethernet Controller Subsystem and Subsystem Vendor Id****Dev1:0x2C**

Default: 0000\_0000h

Attribute: Read only.

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b>
15:0	<b>SUBVENID. Subsystem Vendor ID.</b>

*Note:* This register is aliased to Dev1:0xC8 for modifications.

**LAN Ethernet Controller Capabilities Pointer****Dev1:0x34**

Default: 40h

Attribute: Read only.

Bits	Description
7:0	<b>CAPPTR. Capabilities Pointer.</b>

**LAN Ethernet Controller Interrupt Line and Interrupt Pin****Dev1:0x3C**

Default: 0100h

Attribute: see below.

Bits	Description
15:11	Reserved
10:8	<b>INTPIN.</b> Read-only. Indicates which PCI interrupt pin is used for the LAN Ethernet Controller interrupt. Hardwired to 001b to select PIRQA_L.
7:0	<b>INTLINE.</b> Read-write. This data is not used by hardware. It is used to communicate across software the interrupt line that the interrupt pin is connected to. This register is reset by H_RESET.

**LAN Ethernet Controller Min Grant and Max Latency****Dev1:0x3E**

Default: 0000\_0000h

Attribute: Read only.

Bits	Description
15:8	<b>MAXLAT. Maximum latency.</b> This register specifies the maximum arbitration latency the controller can sustain without causing problems to the network activity. The register value specifies the time in units of 1/4 $\mu$ s. It is recommended that the alias register be programmed to a value of 18h, which corresponds to 6 $\mu$ s.
7:0	<b>MINGNT. Minimum Grant.</b> This register specifies the minimum length of a burst period that the device needs to keep up with the network activity. The length of the burst period is calculated assuming a clock rate of 33 MHz. The register value specifies the time in units of 1/4 $\mu$ s. It is recommended that the alias register be programmed to a value of 18h, which corresponds to 6 $\mu$ s.

**Note:** This register is aliased to Dev1:0xCE for modifications.

**LAN Ethernet Controller Power Management Capabilities****Dev1:0x40**

Default: FE02\_0001h

Attribute: Read only.

Bits	Description
31:16	<b>PMC. Power management capabilities.</b>
15:8	<b>NCPTR. Next capability pointer.</b>
7:0	<b>PMCID. Power management capability ID.</b>

**LAN Ethernet Controller Power Management Control and Status****Dev1:0x44**

Default: 0000h

Attribute: see below.

Bits	Description
15	<b>PMESTS. PME status.</b> Read, write 1b to clear. This bit is set when the function would normally assert the $\overline{\text{PME}}$ signal independent of the state of the PME_EN bit. Writing 1b to this bit clears it and causes the function to stop asserting the PME_L signal (if enabled). Writing a 0 has no effect.  If the function supports PME from D3cold then this bit is sticky and must be explicitly cleared by the operating system each time the operating system is initially loaded. This bit is reset by POR. If the function does not support PME signal assertion from D3cold, either because bit 15 of the PMC Alias register is 0b or DevB:3x64[L7_S3EN] is 0b, PME_STATUS is reset following H_RESET.
14:9	Reserved.
8	<b>PMEEN. PME enable.</b> Read-write. When 1b, PME_EN enables the function to assert the PME_L signal. When 0b, PME assertion is disabled. This bit defaults to 0b if the function does not support PME_L signal generation from D3cold. If the function supports PME_L signal generation from D3cold, then this bit is sticky and must be explicitly cleared by the operating system each time the operating system is initially loaded.  This bit is reset by POR. If the function does not support PME_L signal assertion from D3cold, either because bit 15 of the PMC Alias register is 0b or DevB:3x64[L7_S3EN] is 0b, PME_STATUS is reset following H_RESET.
7:2	Reserved.
1:0	<b>PWRSTAT. Power state.</b> Read-write. This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is as follows:  <div style="margin-left: 40px;">           00b—D0            01b—D1            10b—D2            11b—D3         </div> These bits can be written and read, but their contents have no effect on the operation of the device, except that when Power State is not 0, the device only responds to PCI configuration space accesses, and when Power State transitions from 3 to 0, the read-write bits of the PCI configuration space are reset. This reset does not clear PME_EN or PME_STATUS and also does not change read-only bits initialized by the BIOS.

**LAN Ethernet Controller Subsystem and Subsystem Vendor Id Alias****Dev1:0xC8**

Default: 0000\_0000h

Attribute: Read/write

Bits	Description
31:16	<b>SUBSYSID. Subsystem ID.</b>
15:0	<b>SUBVENID. Subsystem Vendor ID.</b>

*Note:* This register is aliased to Dev1:0x2C for modifications.

### LAN Ethernet Controller Min Grant and Max Latency Alias

Dev1:0xCE

Default: 0000h

Attribute: Read/write

Bits	Description
15:8	<b>MAXLAT. Maximum latency.</b>
7:0	<b>MINGNT. Minimum Grant.</b>

*Note:* This register is aliased to Dev1:0x3E for modifications.

## 4.10.3 Memory-Mapped Registers

The Memory-Mapped Registers give the host CPU access to all programmable features of the device. These registers are mapped directly into PCI memory space so that any programmable feature can be accessed with a single PCI memory read or write transaction. Data in these registers can be accessed as a single byte, a 16-bit word, or a 32-bit double word.

Registers that are logically wider than a double word are shown in the register descriptions as a single register. These may be accessed using multiple smaller accesses or with a single burst access.

Some registers that are smaller than a double word in width (STVAL, PADR[47:32], XMT\_RING\_LEN, RCV\_RING\_LEN) are placed in the memory map in the lower half of a double word. The upper half ignores writes and reads back zeros (STVAL, PADR[47:32]) or ones (XMT\_RING\_LEN, RCV\_RING\_LEN) as appropriate for sign extension. This allows these registers to be accessed with double word reads and writes.

### 4.10.3.1 Command Style Register Access

The command and interrupt enable registers (CMD0, CMD2, CMD3, CMD7, and INTEN0) use a write access technique that in this document is called command style access. Command style access allows the host CPU to write to selected bits of a register without altering bits that are not selected. Command style registers are divided into 4 bytes that can be written independently. The high order bit of each byte is the “value” bit that specifies the value that is written to selected bits of the register. The 7 low order bits of each byte make up a bit map that selects which register bits are altered. If a bit in the bit map is set to 1, the corresponding bit in the register is loaded with the contents of the value bit. If a bit in the bit map is cleared to 0, the corresponding bit in the register is not altered.



For example, if the value 10011010b is written to the least significant byte of a command style register, bits 1, 3, and 4 of the register are set to 1, and the other bits are not altered. If the value 00011010b is written to the same byte, bits 1, 3, and 4 are cleared to 0, and the other bits are not altered.

In the worst case it takes two write accesses to write to all of the bits in a command style register. One access writes to all bits that should be set to 1, and the other access writes to all bits that should be cleared to 0.

#### 4.10.3.2 Memory-Mapped Register Descriptions

In the following register descriptions, the offset listed for each register is the offset relative to the contents of the PCI Memory-Mapped I/O Base Address Register.

In addition to the usual register tables the descriptions below include a write mode identifier with the following meaning:

Write Mode	Description
R	The register can be written any time. (Can be written while the device is running.)
N	The register can be written only when the device is not running (when RUN=0).
SN	The register can be written when the device is suspended or not running.
1	The register can be written only when EN_PMGR and APEP are both 0.
2	The register can be written only when RESET_PHY_PULSE is 0.
3	The register can be written only when PMAT_MODE is 0.

**LAN Ethernet Controller Auto-Poll Value****ENC98**

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>AP_VAL_ACTIVE. Autopoll Value Active.</b> Read-only. This bit is set to 1 while the contents of the selected Autopoll Data Register are being fetched. When this bit is 1, the AP_VAL field of this register does not contain valid data. The host CPU must wait until this bit is 0 before it can write to this register.
30	Reserved.
29	<b>AP_VAL_RD_CMD. Autopoll Read Command.</b> write-only; write mode R. Writing 1 to this bit starts a read access cycle. The contents of the Autopoll Data Register addressed by the AP_VAL_ADDR field of this register are loaded into the AP_VAL field of this register. Writing a 0 to this bit has no effect.
28	<b>AP_RD_ERR. Autopoll Read Error.</b> Read-only. This bit is automatically set to 1 if a read error occurred when the selected auto-poll data register was last read. If this bit is 1, the contents of the AP_VAL field are not valid.
27:19	Reserved.
18:16	<b>AP_VAL_ADDR. Autopoll Data Register Address.</b> Read-write; write mode R. This field contains the address of the Autopoll Data Register (0-5) whose value is loaded into the AP_VAL field of this register during a read cycle.
15:0	<b>AP_VAL. Autopoll Data Register Value.</b> Read-only. This field shows the contents of the selected external Autopoll Data Register register.

This register provides indirect access to registers in the physical layer device.

The access protocol is:

1. Read AP\_VALUE until AP\_VAL\_ACTIVE is 0.
2. Write AP\_VALUE with AP\_VAL\_RD\_CMD = 1 and AP\_VAL\_ADDR set to the address of the desired autopoll value register (0-5).
3. Read AP\_VALUE until AP\_VAL\_ACTIVE is 0. The autopoll value is in the AP\_VAL field.

This register is reset by H\_RESET.

**LAN Ethernet Controller Auto-Poll 0****ENC88**

Default: 8100h

Attribute: Read-only.

Bits	Description
15	<b>AP_REG0_EN. Enable Bit for Autopoll Register 0.</b> This bit is read-only and always has the value 1.
14:13	Reserved.
12:8	<b>AP_REG0_ADDR. AP_REG0 Address.</b> This field is read-only and always has the value 00001.
7:3	Reserved.
4:0	<b>AP_PHY0_ADDR. Auto-Poll PHY0 Address.</b> This field contains the address of the external PHY that contains AP-REG0. The Network Port Manager uses this PHY address for auto-negotiation. The Autopoll State Machine also uses this field as the default PHY address when one or more of the AP_PHYn_DFLT bits are set.

This register controls the automatic polling of the status register of the default external PHY.

This register is reset by H\_RESET.

**LAN Ethernet Controller Auto-Poll 1****ENC8A**

Default: 0000h

Attribute: Read-write; write mode 1.

Bits	Description
15	<b>AP_REG1_EN. Enable Bit for Autopoll Register 1.</b> When this bit and the Auto-Poll External PHY bit (APEP) in CMD3 are both set to 1, the Auto-Poll State Machine periodically reads the external PHY register selected by the AP_PHY1_ADDR and AP_REG1_ADDR fields and sets the APINT1 interrupt bit if it detects a change in the register's contents.
14:13	Reserved.
12:8	<b>AP_REG1_ADDR. AP_REG1 Address.</b> This field contains the register number of an external PHY register that the Auto-Poll State Machine periodically reads if the AP_REG1_EN bit in this register and the APEP bit (CMD3, bit 10) are set.
7	Reserved.
6	<b>AP_PRE_SUP1. Auto-Poll Preamble Suppression.</b> If this bit is set to 1, the Auto-Poll State Machine suppresses the preambles of the MII Management Frames that it uses to periodically read the external PHY register selected by the AP_PHY1_ADDR and AP_REG1_ADDR fields. This bit is ignored when the AP_PHY1_DFLT bit is set.
5	<b>AP_PHY1_DFLT. Auto-Poll PHY1 Default.</b> When this bit is set, the Auto-Poll State Machine ignores the contents of the AP_PHY1_ADDR and AP_PRE_SUP1 fields and uses the AP_PHY0_ADDR field for the address of the PHY device to be polled. If this bit is set, the Auto-Poll State Machine suppresses preambles only if the Port Manager has determined that the default external PHY can accept MII Management Frames without preambles. (The Port Manager examines bit 6 in register 1 of the default PHY to make this determination.)
4:0	<b>AP_PHY1_ADDR. Auto-Poll PHY1 Address.</b> This field contains the address of the external PHY that contains AP_REG1. This bit is ignored when the AP_PHY1_DFLT bit is set.

This register controls the automatic polling of a user-selectable external PHY register.

This register is reset by H\_RESET.

## LAN Ethernet Controller Auto-Poll 2

ENC8C

Default: 0000h

Attribute: Read-write; write mode 1.

Bits	Description
15	<b>AP_REG2_EN. Enable Bit for Autopoll Register 2.</b> When this bit and the Auto-Poll External PHY bit (APEP) in CMD3 are both set to 1, the Auto-Poll State Machine periodically reads the external PHY register selected by the AP_PHY2_ADDR and AP_REG2_ADDR fields and sets the APINT2 interrupt bit if it detects a change in the register's contents.
14:13	Reserved.
12:8	<b>AP_REG2_ADDR. AP_REG2 Address.</b> This field contains the register number of an external PHY register that the Auto-Poll State Machine periodically reads if the AP_REG2_EN bit in this register and the APEP bit (CMD3, bit 10) are set.
7	Reserved.
6	<b>AP_PRE_SUP2. Auto-Poll Preamble Suppression.</b> If this bit is set to 1, the Auto-Poll State Machine suppresses the preambles of the MII Management Frames that it uses to periodically read the external PHY register selected by the AP_PHY2_ADDR and AP_REG2_ADDR fields. This bit is ignored when the AP_PHY2_DFLT bit is set.
5	<b>AP_PHY2_DFLT. Auto-Poll PHY2 Default.</b> When this bit is set, the Auto-Poll State Machine ignores the contents of the AP_PHY2_ADDR and AP_PRE_SUP2 fields and uses the AP_PHY0_ADDR field for the address of the PHY device to be polled. If this bit is set, the Auto-Poll State Machine suppresses preambles only if the Port Manager has determined that the default external PHY can accept MII Management Frames without preambles. (The Port Manager examines bit 6 in register 1 of the default PHY to make this determination.)
4:0	<b>AP_PHY2_ADDR. Auto-Poll PHY2 Address.</b> This field contains the address of the external PHY that contains AP_REG2. This bit is ignored when the AP_PHY2_DFLT bit is set.

This register controls the automatic polling of a user-selectable external PHY register, AP\_REG2.

This register is reset by H\_RESET.

## LAN Ethernet Controller Auto-Poll 3

ENC8E

Default: 0000h

Attribute: Read-write; write mode 1.

Bits	Description
15	<b>AP_REG3_EN. Enable Bit for Autopoll Register 3.</b> When this bit and the Auto-Poll External PHY bit (APEP) in CMD3 are both set to 1, the Auto-Poll State Machine periodically reads the external PHY register selected by the AP_PHY3_ADDR and AP_REG3_ADDR fields and sets the APINT3 interrupt bit if it detects a change in the register's contents.
14:13	Reserved.
12:8	<b>AP_REG3_ADDR. AP_REG3 Address.</b> This field contains the register number of an external PHY register that the Auto-Poll State Machine periodically reads if the AP_REG3_EN bit in this register and the APEP bit (CMD3, bit 10) are set.
7	Reserved.
6	<b>AP_PRE_SUP3. Auto-Poll Preamble Suppression.</b> If this bit is set to 1, the Auto-Poll State Machine suppresses the preambles of the MII Management Frames that it uses to periodically read the external PHY register selected by the AP_PHY3_ADDR and AP_REG3_ADDR fields. This bit is ignored when the AP_PHY3_DFLT bit is set.
5	<b>AP_PHY3_DFLT. Auto-Poll PHY3 Default.</b> When this bit is set, the Auto-Poll State Machine ignores the contents of the AP_PHY3_ADDR and AP_PRE_SUP3 fields and uses the AP_PHY0_ADDR field for the address of the PHY device to be polled. If this bit is set, the Auto-Poll State Machine suppresses preambles only if the Port Manager has determined that the default external PHY can accept MII Management Frames without preambles. (The Port Manager examines bit 6 in register 1 of the default PHY to make this determination.)
4:0	<b>AP_PHY3_ADDR. Auto-Poll PHY3 Address.</b> This field contains the address of the external PHY that contains AP_REG3. This bit is ignored when the AP_PHY3_DFLT bit is set.

This register controls the automatic polling of a user-selectable external PHY register, AP\_REG3.

This register is reset by H\_RESET.

## LAN Ethernet Controller Auto-Poll 5

ENC90

Default: 0000h

Attribute: Read-write; write mode 1.

Bits	Description
15	<b>AP_REG4_EN. Enable Bit for Autopoll Register 4.</b> When this bit and the Auto-Poll External PHY bit (APEP) in CMD3 are both set to 1, the Auto-Poll State Machine periodically reads the external PHY register selected by the AP_PHY4_ADDR and AP_REG4_ADDR fields and sets the APINT4 interrupt bit if it detects a change in the register's contents.
14:13	Reserved.
12:8	<b>AP_REG4_ADDR. AP_REG4 Address.</b> This field contains the register number of an external PHY register that the Auto-Poll State Machine periodically reads if the AP_REG4_EN bit in this register and the APEP bit (CMD3, bit 10) are set.
7	Reserved.
6	<b>AP_PRE_SUP4. Auto-Poll Preamble Suppression.</b> If this bit is set to 1, the Auto-Poll State Machine suppresses the preambles of the MII Management Frames that it uses to periodically read the external PHY register selected by the AP_PHY4_ADDR and AP_REG4_ADDR fields. This bit is ignored when the AP_PHY4_DFLT bit is set.
5	<b>AP_PHY4_DFLT. Auto-Poll PHY4 Default.</b> When this bit is set, the Auto-Poll State Machine ignores the contents of the AP_PHY4_ADDR and AP_PRE_SUP4 fields and uses the AP_PHY0_ADDR field for the address of the PHY device to be polled. If this bit is set, the Auto-Poll State Machine suppresses preambles only if the Port Manager has determined that the default external PHY can accept MII Management Frames without preambles. (The Port Manager examines bit 6 in register 1 of the default PHY to make this determination.)
4:0	<b>AP_PHY4_ADDR. Auto-Poll PHY4 Address.</b> This field contains the address of the external PHY that contains AP_REG4. This bit is ignored when the AP_PHY4_DFLT bit is set.

This register controls the automatic polling of a user-selectable external PHY register, AP\_REG4.

This register is reset by H\_RESET.

**LAN Ethernet Controller Auto-Poll 5****ENC92**

Default: 0000h

Attribute: Read-write; write mode 1.

Bits	Description
15	<b>AP_REG5_EN. Enable Bit for Autopoll Register 5.</b> When this bit and the Auto-Poll External PHY bit (APEP) in CMD3 are both set to 1, the Auto-Poll State Machine periodically reads the external PHY register selected by the AP_PHY5_ADDR and AP_REG5_ADDR fields and sets the APINT5 interrupt bit if it detects a change in the register's contents.
14:13	Reserved
12:8	<b>AP_REG5_ADDR. AP_REG5 Address.</b> This field contains the register number of an external PHY register that the Auto-Poll State Machine periodically reads if the AP_REG5_EN bit in this register and the APEP bit (CMD3, bit 10) are set.
7	Reserved.
6	<b>AP_PRE_SUP5. Auto-Poll Preamble Suppression.</b> If this bit is set to 1, the Auto-Poll State Machine suppresses the preambles of the MII Management Frames that it uses to periodically read the external PHY register selected by the AP_PHY5_ADDR and AP_REG5_ADDR fields. This bit is ignored when the AP_PHY5_DFLT bit is set.
5	<b>AP_PHY5_DFLT. Auto-Poll PHY5 Default.</b> When this bit is set, the Auto-Poll State Machine ignores the contents of the AP_PHY5_ADDR and AP_PRE_SUP5 fields and uses the AP_PHY0_ADDR field for the address of the PHY device to be polled. If this bit is set, the Auto-Poll State Machine suppresses preambles only if the Port Manager has determined that the default external PHY can accept MII Management Frames without preambles. (The Port Manager examines bit 6 in register 1 of the default PHY to make this determination.)
4:0	<b>AP_PHY5_ADDR. Auto-Poll PHY5 Address.</b> This field contains the address of the external PHY that contains AP_REG5. This bit is ignored when the AP_PHY5_DFLT bit is set.

This register controls the automatic polling of a user-selectable external PHY register, AP\_REG5.

This register is reset by H\_RESET.

**LAN Ethernet Controller Receive Ring Base Address****ENC120**

Default: 0000\_0000h

Attribute: Read-write.

Bits	Description
31:0	<b>BADR. Base address of receive descriptor ring.</b>

This 32-bit register allows the Receive Descriptor Ring to be located anywhere in a 32-bit address space. This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 0 Base Address****ENC100**

Default: 0000\_0000h

Attribute: Read-write.

Bits	Description
31:0	<b>BADX0. Base address of transmit descriptor ring 0.</b>

This 32-bit register allows the Transmit Descriptor Rings to be located anywhere in a 32-bit address space. This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 1 Base Address****ENC108**

Default: 0000\_0000h

Attribute: Read-write.

Bits	Description
31:0	<b>BADX1. Base address of transmit descriptor ring 1.</b>

This 32-bit register allows the Transmit Descriptor Rings to be located anywhere in a 32-bit address space. This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 2 Base Address****ENC110**

Default: 0000\_0000h

Attribute: Read-writeRead-write.

Bits	Description
31:0	<b>BADX2. Base address of transmit descriptor ring 2.</b>

This 32-bit register allows the Transmit Descriptor Rings to be located anywhere in a 32-bit address space. This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 3 Base Address****ENC118**

Default: 0000\_0000h

Attribute: Read-write.

Bits	Description
31:0	<b>BADX3. Base address of transmit descriptor ring 3.</b>

This 32-bit register allows the Transmit Descriptor Rings to be located anywhere in a 32-bit address space. This register is reset by RESET\_L.



**LAN Ethernet Controller Chip ID****ENC004**

Default: ???\_?003h

Attribute: Read-only.

Bits	Description
31:28	<b>VER. Version.</b> This 4-bit pattern is silicon-revision dependent.
27:12	<b>PARTID. Part number.</b> The 16-bit codes for the controller are undefined at present. This part number is different from that stored in the Device ID register in the PCI configuration space.
11:1	<b>MANFID. Manufacturer ID.</b> The 11-bit manufacturer code for AMD is 0000000001b. This code is per the JEDEC Publication 106-A. Note that this code is not the same as the Vendor ID in the PCI configuration space.
0	<b>ONE. Always 1.</b>

This register holds a product identification number that system software can use to determine which software driver to load. Note that the part number & Manufacturer ID are not the same as the values in the PCI configuration space.

## LAN Ethernet Controller Command 0 (CMD0)

ENC048

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> Read-write. The value of this bit is written to any bits in the CMD0 register that correspond to bits in the CMD0[30:24] bit map field that are set to 1.
30:24	Reserved.
23	<b>VALBIT2. Value bit for byte 2.</b> Read-write. The value of this bit is written to any bits in the CMD0 register that correspond to bits in the CMD0[22:16] bit map field that are set to 1.
22:17	Reserved.
16	<b>RDMD0. Receive Demand for ring 0.</b> write-only; write mode R. When set causes the Descriptor Management Unit to access the Receive Descriptor Ring if it does not already own the next descriptor. This bit is also reset when the RUN bit is cleared.
15	<b>VALBIT1. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD0 register that correspond to bits in the CMD0[14:8] bit map field that are set to 1.
14:12	Reserved.
11	<b>TDMD3. Transmit Demand for ring 3.</b> write-only; write mode R. Which when set causes the Buffer Management Unit to access the Transmit Descriptor Ring. This bit is also reset when the RUN bit is cleared.
10	<b>TDMD2. Transmit Demand for ring 2.</b> write-only; write mode R. Which when set causes the Buffer Management Unit to access the Transmit Descriptor Ring. This bit is also reset when the RUN bit is cleared.
9	<b>TDMD1. Transmit Demand for ring 1.</b> write-only; write mode R. Which when set causes the Buffer Management Unit to access the Transmit Descriptor Ring. This bit is also reset when the RUN bit is cleared.
8	<b>TDMD0. Transmit Demand for ring 0.</b> write-only; write mode R. Which when set causes the Buffer Management Unit to access the Transmit Descriptor Ring. This bit is also reset when the RUN bit is cleared.
7	<b>VALBIT0. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD0 register that correspond to bits in the CMD0[6:0] bit map field that are set to 1.
6	<b>UINTCMD. User Interrupt Command.</b> write-only; write mode R. UINTCMD can be used by the host to generate an interrupt unrelated to any network activity. Writing a 1 to this bit causes the UINT bit in the Interrupt Register to be set to 1, which in turn causes INTA to be asserted if interrupts are enabled. This bit is also reset when the RUN bit is cleared.
5	<b>RX_FAST_SPND. Receive Fast Suspend.</b> Read-write; write mode R. Setting this bit causes the receiver to suspend its activities as quickly as possible without stopping in the middle of a frame reception. Setting RX_FAST_SPND does not stop the DMA controller from transferring frame data from the receive FIFO to host memory. If a frame is being received at the time that RX_FAST_SPND is set to 1, the reception of that frame is completed, but no more frames are received until RX_FAST_SPND is cleared to 0. After the receiver has suspended its activity, the RX_SUSPENDED bit in the Status register and the Suspend Interrupt (SPNDINT) bit in the Interrupt register are set, which causes an interrupt to occur if interrupts are enabled and the SPNDINTEN bit in the Interrupt Enable register is set. This bit is also reset when the RUN bit is cleared.

Bits	Description
4	<b>TX_FAST_SPND. Transmit Fast Suspend.</b> Read-write; write mode R. Setting this bit causes the transmitter to suspend its activities as quickly as possible without stopping in the middle of a frame transmission. Setting TX_FAST_SPND does not stop the DMA controller from transferring frame data from host memory to the transmit FIFO. If a frame is being transmitted at the time that TX_FAST_SPND is set to 1, the transmission of that frame is completed, but no more frames are transmitted until TX_FAST_SPND is cleared to 0. After the transmitter has suspended its activity, the TX_SUSPENDED bit in the Status register and the Suspend Interrupt (SPNDINT) bit in the INTO register are set, which causes an interrupt to occur if interrupts are enabled and the SPNDINTEN bit in the Interrupt Enable register is set. This bit is also reset when the RUN bit is cleared.
3	<b>RX_SPND. Receive Suspend.</b> Read-write; write mode R. Setting this bit causes the receiver to suspend its activities without stopping in the middle of a frame reception. After the receiver suspends its activities, the DMA controller continues copying data from the Receive FIFO into host system memory until the FIFO is empty. After the Receive FIFO has been emptied, the RX_SUSPENDED bit in the Status register and the Suspend Interrupt (SPNDINT) bit in the Interrupt register are set. Setting the SPNDINT bit causes an interrupt to occur if interrupts are enabled and the SPNDINTEN bit in the Interrupt Enable register is set. This bit is also reset when the RUN bit is cleared.
2	<b>TX_SPND. Transmit Suspend.</b> Read-write; write mode R. Setting this bit causes the transmitter to suspend its activities without stopping in the middle of a frame transmission. The DMA controller suspends after completing the copying of data from host system memory into the Transmit FIFO for the current frame (if any). The transmitter continues operation until all frames in the Transmit FIFO have been transmitted. The TX_SUSPENDED bit in the Status register and the Suspend Interrupt (SPNDINT) bit in the Interrupt register are then set. Setting the SPNDINT bit causes an interrupt to occur if interrupts are enabled and the SPNDINTEN bit in the Interrupt Enable register is set. This bit is also reset when the RUN bit is cleared.
1	<b>INTREN. Interrupt Enable.</b> Read-write; write mode R. This bit allows INTA to be asserted if any bit in the Interrupt register is set. If INTREN is cleared to 0, INTA is not asserted, regardless of the state of the Interrupt register.
0	<b>RUN. Run bit.</b> Read-write; write mode R. Setting the RUN bit enables the controller to start processing descriptors and transmitting and receiving packets. Clearing the RUN bit to 0 abruptly disables the transmitter, receiver, and descriptor processing logic, possibly while a frame is being transmitted or received.  The act of changing the RUN bit from 1 to 0 causes the following bits to be reset to 0: TX_SPND, RX_SPND, TX_FAST_SPND, RX_FAST_SPND, RDMD, all TDMD bits, RINT, all TINT bits, MPINT, and SPNDINT.

CMD0 is a command-style register. This register is reset by RESET\_L. Some bits in this register are also reset when the RUN bit is cleared.

## LAN Ethernet Controller Command 2 (CMD2)

ENC050

Default: 0000\_0000h

Attribute: Read-write; write mode N.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> Read-write. The value of this bit is written to any bits in the CMD2 register that correspond to bits in the CMD2[30:24] bit map field that are set to 1.
30	Reserved.
29	<b>CONDUIT_MODE. Disables backoff and retry.</b> Transmit retry and frame discard is controlled by COL and CRS. See Conduit Mode section for details.
28:24	Reserved.
23	<b>VALBIT2. Value bit for byte 2.</b> Read-write. The value of this bit is written to any bits in the CMD2 register that correspond to bits in the CMD2[22:16] bit map field that are set to 1.
22:20	Reserved.
19	<b>RPA. Runt Packet Accept.</b> This bit forces the controller to accept runt packets (packets shorter than 64 bytes). The minimum packet size that can be received is 12 bytes including the FCS.
18	<b>DRCVPA. Disable Receive Physical Address.</b> When set, the physical address detection (Station or node ID) of the controller is disabled. Frames addressed to the node's individual physical address will not be recognized.
17	<b>DRCVBC. Disable Receive Broadcast.</b> When set, disables the controller from receiving broadcast messages. Used for protocols that do not support broadcast addressing, except as a function of multicast. DRCVBC is cleared by activation of H_RESET (broadcast messages will be received) and is unaffected by the clearing of the RUN bit.
16	<b>PROM. Promiscuous Mode.</b> When PROM = 1, all incoming receive frames are accepted, regardless of their destination addresses.
15	<b>VALBIT1. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD2 register that correspond to bits in the CMD2[14:8] bit map field that are set to 1.
14	Reserved.
13	<b>ASTRIP_RCV. Auto Strip Receive.</b> When set, ASTRIP_RCV enables the automatic pad stripping feature. For any receive frame whose length field has a value less than 46, the pad and FCS fields are stripped and not placed in the FIFO.
12	<b>RCV_DROP0.</b> If this bit is set, receive frames assigned to ring 0 are dropped if no descriptor is available when the frame is ready to be written to system memory. If the bit is 0, these frames are saved until descriptors are available.
11	<b>EMBA. Enable Modified Back-off Algorithm</b> (see Contention Resolution section in Media Access Management section for more details). If EMBA is set, a modified back-off algorithm is implemented.
10	<b>DXMT2PD. Disable Transmit Two Part Deferral</b> (see Medium Allocation section in the Media Access Management section for more details). If DXMT2PD is set, Transmit Two Part Deferral is disabled.
9	<b>LTINTEN. Last Transmit Interrupt Enable.</b> When this bit is set to 1, the LTINT bit in transmit descriptors can be used to determine when transmit interrupts occur. The Transmit Interrupt (TINT) bit is set after a frame has been copied to the Transmit FIFO if the LTINT bit in the frame's last transmit descriptor is set. If the LTINT bit in the frame's last descriptor is 0, TINT is not set after the frame has been copied to the Transmit FIFO.

Bits	Description
8	<b>DXMTFCS. Disable Transmit CRC (FCS).</b> When DXMTFCS is cleared to 0, the transmitter generates and appends an FCS to the transmitted frame. When DXMTFCS is set to 1, no FCS is generated or sent with the transmitted frame. DXMTFCS is overridden when ADD_FCS and ENP bits are set in the transmit descriptor. When the auto padding logic, which is enabled by the APAD_XMT bit (CMD2, bit6), adds padding to a frame, a valid FCS field is appended to the frame, regardless of the state of DXMTFCS. If DXMTFCS is set and ADD_FCS is clear for a particular frame, no FCS is generated. If ADD_FCS is set for a particular frame, the state of DXMTFCS is ignored and a FCS is appended on that frame by the transmit circuitry. See also the ADD_FCS bit in the transmit descriptor. This bit was called DTCR in the Am7990 device.
7	<b>VALBIT0. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD2 register that correspond to bits in the CMD2[6:0] bit map field that are set to 1.
6	<b>APAD_XMT. Auto Pad Transmit.</b> When set, APAD_XMT enables the automatic padding feature. Transmit frames are padded to extend them to 64 bytes including FCS. The FCS is calculated for the entire frame, including pad, and appended after the pad field. When the auto padding logic modifies a frame, a valid FCS field is appended to the frame, regardless of the state of the DXMTFCS bit (CMD2, bit 8) and of the ADD_FCS bit in the transmit descriptor.
5	<b>DRTY. Disable Retry.</b> When DRTY is set to 1, the controller attempts only one transmission. In this mode, the device does not protect the first 64 bytes of frame data in the Transmit FIFO from being overwritten, because automatic retransmission will not be necessary. When DRTY is cleared to 0, the controller attempts 16 transmissions before signaling a retry error.
4	<b>INLOOP. Internal Loopback.</b> When this bit is set, the transmitter is internally connected to the receiver so that the data output and control signals are connected internally to the data input and control signals. The device is forced into full duplex mode so that collisions can not occur. The INLOOP and EXLOOP bits should not be set at the same time.
3	<b>EXLOOP. External Loopback.</b> When this bit is set, the device is forced into full duplex mode so that collisions can not occur during loop back testing. If the PHY interface output signals are connected externally to the PHY interface inputs, then transmitted frames will also be received. This connection can be made by attaching an external jumper or by programming an attached PHY to loopback mode. The INLOOP and EXLOOP bits should not be set at the same time.
2	Reserved.
1	<b>REX_UFLO. Retransmit on Underflow.</b> When this bit is set to 1, if the transmitter is forced to abort a transmission because the transmit FIFO underflows, the transmitter does not discard the frame. Instead, it automatically waits until the entire frame has been loaded into the transmit FIFO and then restarts the transmission process. When this bit is cleared to 0, if the transmitter is forced to abort a transmission because the FIFO underflows, the transmitter discards the frame. In either case, the XmtUnderrunPkts counter is incremented.
0	Reserved.

CMD2 is a command-style register. This register is reset by RESET\_L.

## LAN Ethernet Controller Command 3 (CMD3)

ENC054

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> Read-write. The value of this bit is written to any bits in the CMD3 register that correspond to bits in the CMD3[30:24] bit map field that are set to 1.
30:29	Reserved.
28:24	Reserved.
23	<b>VALBIT2. Value bit for byte 2.</b> Read-write. The value of this bit is written to any bits in the CMD3 register that correspond to bits in the CMD3[22:16] bit map field that are set to 1.
21	<b>JUMBO. Write mode N. Accept Jumbo Frames.</b> Read-write. This bit affects the way the MIB counters count long frames. If JUMBO is 0, only frames that are between 64 and 1518 bytes (or 1522 bytes if VSIZE is set to 1) are counted as valid frames. When JUMBO is 1, any frame between 64 and 9018 bytes (or 9022 bytes if VSIZE is set to 1) with a valid FCS field is counted as a valid frame.
20	<b>VSIZE. Write mode N. VLAN Frame Size.</b> Read-write. This bit determines the maximum frame size used for determining when to increment the XmtPkts1024to1518Octets, XmtExcessiveDefer, RcvPkts1024to1518Octets, and RcvOversizePkts MIB counters and when to assert the Excessive Deferral Interrupt. When this bit is set to 1 the maximum frame size is 1522 bytes (or 9022 if JUMBO is set). When it is cleared to 0, the maximum frame size is 1518 bytes (or 9018 if JUMBO is set).
19	<b>VLONLY. Write mode N. Admit Only VLAN Frames.</b> Read-write. When this bit is set to 1, only frames with a VLAN Tag Header containing a non-zero VLAN ID field are received. All other frames are rejected.
18	<b>VL_TAG_DEL. Write mode N. Delete VLAN Tag.</b> Read-write. If this bit is set, the receiver deletes the 4 bytes of VLAN tag from the frame data. The VLAN tag information is reported in the descriptor. The number of bytes written to system memory and the MCNT field of the receive descriptor is 4 smaller than the actual number of bytes received. The MIB counters are incremented by the actual number of bytes received.
17:16	Reserved.
15	<b>VALBIT1. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD3 register that correspond to bits in the CMD3[14:8] bit map field that are set to 1.
14	<b>EN_PMGR. write mode N. Enables the port manager.</b> Read-write. EN_PMGR is cleared on reset, so the port manager starts only after BIOS is complete. This bit is the inverse of bits called DANAS or DISPM in certain other AMD network controllers.
13	Reserved.
12	<b>FORCE_FULL_DUPLEX. Write mode N. Force Full Duplex.</b> Read-write. This bit is called Full-Duplex Enable (FDEN) in other PCnet™ family devices.) This bit controls whether full-duplex operation is enabled. When FORCE_FD is cleared and the Port Manager is disabled, the controller will always operate in the half-duplex mode. When FORCE_FD is set, the controller operates in full-duplex mode. Do not set this bit when the Port Manager is enabled.
11	<b>FORCE_LINK_STATUS. Write mode N. Force Link Status.</b> Read-write. When this bit is set, the internal link status is forced to the Pass state regardless of the actual state of the PHY device. When this bit is cleared to 0, the internal link status is determined by the Port Manager.
10	<b>APEP. Write mode R. MII Auto-Poll External PHY (APEP).</b> Read-write. When set to 1, the Network Controller polls the MII status register in the external PHY. This feature allows the software driver or upper layers to see any changes in the status of the external PHY. An interrupt, when enabled, is generated when the contents of the new status is different from the previous status.

Bits	Description
9	<b>MPPLBA. Write mode N. Magic Packet™ Physical Logical Broadcast Accept.</b> Read-write; If MPPLBA is at its default value of 0, the controller only detects a Magic Packet frame if the destination address of the packet matches the content of the physical address register (PADR). If MPPLBA is set to 1, the destination address of the Magic Packet frame can be unicast, multicast, or broadcast. Note that the setting of MPPLBA only affects the address detection of the Magic Packet frame. The Magic Packet frame's data sequence must be made up of 16 consecutive physical addresses (PADR[47:0]) regardless of what kind of destination address it has.
8	<b>BKPRS_EN. Write mode N. Back Pressure Enable.</b> Read-write; Back pressure is enabled when (1) this bit is set, (2) the device is operating in half-duplex mode, and (3) the receiver is in the congested state.
7	<b>VALBIT0. Value bit for byte 0.</b> Read-write. The value of this bit is written to any bits in the CMD3 register that correspond to bits in the CMD3[6:0] bit map field that are set to 1.
6:3	Reserved.
2	<b>RESET_PHY_PULSE. Write mode N. External PHY Reset Pulse.</b> write-only. Write-only, always reads 0. Writing a 1 to this bit causes the PHY_RST pin to assert for a duration specified by the RESET_PHY_WIDTH (CTRL1, bits [20:16]) register. RESET_PHY_PULSE and RESET_PHY should not be set at the same time.
1	<b>RESET_PHY. Write mode N. External PHY Reset.</b> Read-write. The PHY_RST pin asserts when RESET_PHY is set to 1 and deasserts when RESET_PHY is cleared to 0. RESET_PHY_PULSE and RESET_PHY should not be set at the same time.
0	<b>PHY_RST_POL. Write mode N. PHY_RST Pin Polarity.</b> Read-write; If 0, PHY_RST is active High. If 1, PHY_RST is active Low.

CMD3 is a command-style register. This register is reset by RESET\_L.

## LAN Ethernet Controller Command 7 (CMD7)

ENC064

Default: 0000\_0000h

Attribute: Read-write.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> Read-write. The value of this bit is written to any bits in the CMD7 register that correspond to bits in the CMD7[30:24] bit map field that are set to 1.
30:24	Reserved.
23	<b>VALBIT2. Value bit for byte 2.</b> Read-write. The value of this bit is written to any bits in the CMD7 register that correspond to bits in the CMD7[22:16] bit map field that are set to 1.
22:16	Reserved.
15	<b>VALBIT1. Value bit for byte 1.</b> Read-write. The value of this bit is written to any bits in the CMD7 register that correspond to bits in the CMD7[14:8] bit map field that are set to 1.
14:8	Reserved.
7	<b>VALBIT0. Value bit for byte 0.</b> Read-write. The value of this bit is written to any bits in the CMD7 register that correspond to bits in the CMD7[6:0] bit map field that are set to 1.
6:5	Reserved.
4	<b>PMAT_SAVE_MATCH. Write mode 3.</b> When this bit is 1, the wake-up frame that caused the PME Status bit to be set is saved in the receive FIFO so that it can be retrieved by the host CPU after it wakes up. When this bit is 0, the wake-up frame is discarded.
3	<b>PMAT_MODE. Pattern Match Mode.</b> Write mode SN. Writing a 1 to this bit enables Pattern Match Mode. This bit should be set only after the Pattern Match RAM has been programmed.
2	Reserved.
1	<b>MPEN_SW. Magic Packet™ Mode Enable.</b> Write mode SN. The controller enters the Magic Packet mode when this bit is set to 1.
0	<b>LCMODE_SW. Link Change Wake-up Mode.</b> Write mode SN. When this bit is set to 1, the LC_DET bit in STAT0 gets set when the MII auto polling logic detects a Link Change.

CMD7 is a command-style register. This register is reset when power is first applied to the device (power-on reset). The contents of this register are not affected by the state of RESET\_L.



## LAN Ethernet Controller Control 1

ENC06C

Default: 0001\_0111h

Attribute: see below.

Bits	Description										
31:22	Reserved.										
20:16	<b>RESET_PHY_WIDTH.</b> Read-write; write mode 2. Duration of PHY reset pulse initiated by setting PHY_RESET_PULSE in units of 1.3 microseconds.										
15:10	Reserved.										
9:8	<b>XMTSP. Transmit Start Point.</b> Read-write; write mode N. XMTSP controls the point at which preamble transmission attempts to commence in relation to the number of bytes written to the MAC Transmit FIFO for the current transmit frame. When the entire frame is in the MAC Transmit FIFO, transmission starts regardless of the value in XMTSP. If REX_UFLO is set, no frame data is overwritten until the frame has been transmitted or discarded. Otherwise, no data is overwritten until at least 64 bytes have been transmitted. Note that when XMTSP = 11b, transmission does not start until the complete frame has been copied into the Transmit FIFO. This mode is useful in a system where high latencies cannot be avoided.  <table border="0"> <tr> <td>XMTSP[1:0]</td> <td>Bytes Written</td> </tr> <tr> <td>00</td> <td>16</td> </tr> <tr> <td>01</td> <td>64</td> </tr> <tr> <td>10</td> <td>128</td> </tr> <tr> <td>11</td> <td>Full Frame</td> </tr> </table>	XMTSP[1:0]	Bytes Written	00	16	01	64	10	128	11	Full Frame
XMTSP[1:0]	Bytes Written										
00	16										
01	64										
10	128										
11	Full Frame										
7:5	Reserved.										
4	<b>CACHE_ALIGN. PCI Bus Burst Align is always set .</b> Read-only. . If a burst transfer starts in the middle of a cache line, the transfer stops at the first cache line boundary.										
3:0	<b>BURST_LIMIT. PCI Bus Burst Limit is fixed at 1.</b> Read-only. All burst transfers end at the first cache line boundary.										

This register contains several miscellaneous control bits. Each byte of this register controls a single function. It is not necessary to do a read-modify-write operation to change a function's settings if only a single byte of the register is written.

This register is reset by RESET\_L.

LAN Ethernet Controller Control 2

ENC070

Default: 0000\_0004h

Attribute: Read-write; write mode 1.

Bits	Description																								
31:10	Reserved.																								
9:8	<b>FMDC. Fast Management Data Clock.</b> When FMDC is set to 2h the MII Management Data Clock runs at 10 MHz max. The Management Data Clock is then no longer IEEE 802.3u-compliant, so setting this bit should be used with care. The accompanying external PHY must also be able to accept management frames at the new clock rate. When FMDC is set to 1h, the MII Management Data Clock runs at 5 MHz max. The Management Data Clock is then no longer be IEEE 802.3u-compliant, so setting this bit should be used with care. The accompanying external PHY must also be able to accept management frames at the new clock rate. When FMDC is cleared to 0h, the MII Management Data Clock runs at 2.5 MHz max and is compliant to IEEE 802.3u standards.																								
7	<b>XPHYRST. External PHY Reset.</b> When XPHYRST is set, the controller after an H_RESET issues an MII management frame that resets the external PHY. This bit is needed when there is no way to ensure the state of the external PHY. This bit must be reprogrammed after every H_RESET. XPHYRST is only valid when the internal Network Port Manager is scanning for a network port.																								
6	<b>XPHYANE. External PHY Auto-Negotiation Enable.</b> This bit forces the external PHY into enabling Auto-Negotiation. When this bit is cleared to 0, the controller sends an MII management frame disabling Auto-Negotiation. XPHYANE is only valid when the internal Network Port Manager is scanning for a network port.																								
5	<b>XPHYFD. External PHY Full Duplex.</b> When set, this bit forces the external PHY into full duplex when Auto-Negotiation is not enabled. XPHYFD is only valid when the internal Network Port Manager is scanning for a network port.																								
4:3	<p><b>XPHYSP. External PHY Speed.</b> When auto-negotiation is disabled, the controller sends a management frame to the external PHY to set the data rate based on the contents of this field.</p> <table border="1"> <thead> <tr> <th>XPHYSP</th> <th>Data Rate</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>10 Mb/s</td> </tr> <tr> <td>01</td> <td>100 Mb/s</td> </tr> <tr> <td>1X</td> <td>Reserved</td> </tr> </tbody> </table>	XPHYSP	Data Rate	00	10 Mb/s	01	100 Mb/s	1X	Reserved																
XPHYSP	Data Rate																								
00	10 Mb/s																								
01	100 Mb/s																								
1X	Reserved																								
2:0	<p><b>APDW. MII Auto-Poll Dwell Time.</b> APDW determines the dwell time (or idle time) between MII Management Frames accesses when Auto-Poll is turned on.</p> <table border="1"> <thead> <tr> <th>APDW</th> <th>MDC Periods Between Frames</th> <th>Polling Period (Start frame to start frame)</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>1</td> <td>27.2 μs @ 2.5 MHz</td> </tr> <tr> <td>001</td> <td>128</td> <td>77.6 μs @ 2.5 MHz</td> </tr> <tr> <td>010</td> <td>256</td> <td>128.8 μs @ 2.5 MHz</td> </tr> <tr> <td>011</td> <td>512</td> <td>231.2 μs @ 2.5 MHz</td> </tr> <tr> <td>100</td> <td>1024</td> <td>436.0 μs @ 2.5 MHz</td> </tr> <tr> <td>101</td> <td>2048</td> <td>845.6 μs @ 2.5 MHz</td> </tr> <tr> <td>110-111</td> <td>Reserved</td> <td>Reserved</td> </tr> </tbody> </table>	APDW	MDC Periods Between Frames	Polling Period (Start frame to start frame)	000	1	27.2 μs @ 2.5 MHz	001	128	77.6 μs @ 2.5 MHz	010	256	128.8 μs @ 2.5 MHz	011	512	231.2 μs @ 2.5 MHz	100	1024	436.0 μs @ 2.5 MHz	101	2048	845.6 μs @ 2.5 MHz	110-111	Reserved	Reserved
APDW	MDC Periods Between Frames	Polling Period (Start frame to start frame)																							
000	1	27.2 μs @ 2.5 MHz																							
001	128	77.6 μs @ 2.5 MHz																							
010	256	128.8 μs @ 2.5 MHz																							
011	512	231.2 μs @ 2.5 MHz																							
100	1024	436.0 μs @ 2.5 MHz																							
101	2048	845.6 μs @ 2.5 MHz																							
110-111	Reserved	Reserved																							

This register contains several miscellaneous control bits. Each byte of this register controls a single function. It is not necessary to do a read-modify-write operation to change a function's settings if only

a single byte of the register is written.

This register is reset by RESET\_L.

### LAN Ethernet Controller Group A Delayed Interrupt

ENC0A8

Default: 0000\_0000h

Attribute: Read-write; write mode R.

Bits	Description
31:29	Reserved.
28	<b>DLY_INT_A_R0.</b> If 1, RINT0 is delayed with EVENT_COUNT_A and MAX_DELAY_TIME_A.
27	<b>DLY_INT_A_T3.</b> If 1, TINT3 is delayed with EVENT_COUNT_A and MAX_DELAY_TIME_A.
26	<b>DLY_INT_A_T2.</b> If 1, TINT2 is delayed with EVENT_COUNT_A and MAX_DELAY_TIME_A.
25	<b>DLY_INT_A_T1.</b> If 1, TINT1 is delayed with EVENT_COUNT_A and MAX_DELAY_TIME_A.
24	<b>DLY_INT_A_T0.</b> If 1, TINT0 is delayed with EVENT_COUNT_A and MAX_DELAY_TIME_A.
23:21	Reserved.
20:16	<b>EVENT_COUNT_A.</b> This field indicates the maximum number of Group A interrupt events that can occur before a delayed interrupt signal occurs.
15:11	Reserved.
10:0	<b>MAX_DELAY_TIME_A.</b> This field indicates the maximum time (measured in units of 10 $\mu$ s) that can elapse after a Group A interrupt event occurs before a delayed interrupt signal occurs.

This register controls Delayed Interrupt Group A. It allows the user to assign interrupt events to Group A and to specify the amount of time or the number of events for which a Group A interrupt can be delayed.

This register is reset by RESET\_L.

**LAN Ethernet Controller Group B Delayed Interrupt****ENC0AC**

Default: 0000\_0000h

Attribute: Read-write; write mode R.

Bits	Description
31:29	Reserved.
28	<b>DLY_INT_B_R0.</b> If 1, RINT0 is delayed with EVENT_COUNT_B and MAX_DELAY_TIME_B.
27	<b>DLY_INT_B_T3.</b> If 1, TINT3 is delayed with EVENT_COUNT_B and MAX_DELAY_TIME_B.
26	<b>DLY_INT_B_T2.</b> If 1, TINT2 is delayed with EVENT_COUNT_B and MAX_DELAY_TIME_B.
25	<b>DLY_INT_B_T1.</b> If 1, TINT1 is delayed with EVENT_COUNT_B and MAX_DELAY_TIME_B.
24	<b>DLY_INT_B_T0.</b> If 1, TINT0 is delayed with EVENT_COUNT_B and MAX_DELAY_TIME_B.
23:21	Reserved.
20:16	<b>EVENT_COUNT_B.</b> This field indicates the maximum number of Group B interrupt events that can occur before a delayed interrupt signal occurs.
15:11	Reserved.
10:0	<b>MAX_DELAY_TIME_B.</b> This field indicates the maximum time (measured in units of 10 $\mu$ s) that can elapse after a Group B interrupt event occurs before a delayed interrupt signal occurs.

This register controls Delayed Interrupt Group B. It allows the user to assign interrupt events to Group B and to specify the amount of time or the number of events for which a Group B interrupt can be delayed.

This register is reset by RESET\_L.

## LAN Ethernet Controller Flow Control

ENC0C8

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> Read-write. The value of this bit is written to any bits in this register that correspond to any bit of this register set to 1 in the range 30:24.
30	<b>PAUSE_LEN_CHG. Write mode R. PAUSE Length Change.</b> Read-write; Before changing the value of PAUSE_LEN, the host CPU must wait until PAUSE_LEN_CHG = 0 and then write to this register setting PAUSE_LEN_CHG to 1 and setting PAUSE_LEN to the new value.
29:24	Reserved.
23	<b>VALBIT2. Value bit for byte 2.</b> Read-write. The value of this bit is written to any bits in this register that correspond to any bit of this register set to 1 in the range 22:16.
22	<b>FTPE. Write mode N. Force Transmit Pause Enable.</b> Read-write; When this bit is set, MAC Control Pause Frames may be transmitted, regardless of the results of autonegotiation.
21	<b>FRPE. Write mode N. Force Receive Pause Enable.</b> Read-write; When this bit is set, MAC Control Pause Frames received are recognized and obeyed, regardless of the results of autonegotiation.
20	<b>NAPA. Write mode N. Negotiate Asymmetric Pause Ability.</b> Read-write; This bit is loaded into the ASM_DIR advertisement bit of PHY register 4 prior to starting autonegotiation.
19	<b>NPA. Write mode R. Negotiate Pause Ability.</b> Read-write; This bit is loaded into the PAUSE advertisement bit of PHY register 4 prior to starting autonegotiation.
18	<b>FIXP. Write mode R. Fixed Length Pause.</b> Read-write; When this bit is set to 1, all MAC Control Pause Frames transmitted from the device contain a Request_operand field that is copied from the PAUSE_LEN field of this register. When this bit is cleared to 0, a Pause Frame with its Request_operand field set to 0FFFFh is sent when FCCMD is set to 1. A Pause Frame with its Request_operand field cleared to 0000h is sent when FCCMD becomes 0.
17	Reserved.
16	<b>FCCMD. Write mode R. Flow Control Command.</b> Read-write; This bit can be used by the host to send Pause Frames or to enable or disable half-duplex back-pressure mode.  In full-duplex mode, if the FIXP bit is 1, FCCMD is a write-only bit. When FCCMD is set to 1, a Pause Frame is sent with Request_operand copied from the PAUSE_LEN field of this register. If the FIXP bit is 0, FCCMD is a read-write bit. When FCCMD is set to 1, a Pause Frame is sent with its Request_operand field filled with all 1s. When FCCMD is cleared to 0, a Pause Frame is sent with its Request_operand field filled with all 0s.  In half-duplex mode, FCCMD is a read-write bit. Setting FCCMD to 1 puts the MAC into back-pressure mode. Clearing FCCMD to 0 disables back-pressure mode.
15:0	<b>PAUSE_LEN. Write mode R. Pause Length.</b> Read-write; The contents of this field are copied into the Request_operand fields of MAC Control Pause Frames that are transmitted while the FIXP pin in this register has the value 1.  The protocol for changing PAUSE_LEN is: 1) Read FLOW until PAUSE_LEN_CHG is 0. 2) Write FLOW with PAUSE_LEN_CHG = 1 and new value of PAUSE_LEN.  PAUSE_LEN_CHG clears when write is complete, but the driver need not poll for this.

The upper 16 bits of the FLOW register is a command-style register while the lower 16 bits are a normal register. This register is reset by RESET\_L.

**LAN Ethernet Controller Inter Frame Spacing****ENC18D**

Default: 60h

Attribute: Read-write; write mode N.

Bits	Description
7:0	<p><b>IPG. Inter Packet Gap.</b> This value indicates the minimum number of network bit times after the end of a frame that the transmitter waits before it starts transmitting another frame. In half-duplex mode the end of the frame is determined by CRS, while in full-duplex mode the end of the frame is determined by TX_EN. The IPG value can be adjusted to compensate for delays through the external PHY device. IPG should be programmed to the nearest nibble. The two least significant bits are ignored. For example, programming IPG to 63h has the same effect as programming it to 60h.</p> <p><b>CAUTION:</b> Use this parameter with care. By lowering the IPG below the IEEE 802.3 standard 96 bit times, the Network Controller can interrupt normal network behavior. IPG must not be set to a value less than 64 (decimal).</p>

This register is reset by RESET\_L.

**LAN Ethernet Controller Interframe Spacing Part 1****ENC18C**

Default: 3Ch

Attribute: Read-write; write mode N.

Bits	Description
7:0	<p><b>IFS1. InterFrameSpacingPart1.</b> Changing IFS1 allows the user to program the value of the InterFrame- SpacePart1 timing. The Network Controller sets the default value at 60 bit times (3ch). See Section 3.10.4.3.1, "Medium Allocation," on page 86 for more details. The equation for setting IFS1 when <math>IPG \geq 96</math> bit times is:</p> $IFS1 = IPG - 36 \text{ bit times}$ <p>IPG should be programmed to the nearest nibble. The two least significant bits are ignored. For example, programming IPG to 63h has the same effect as programming it to 60h.</p>

This register is reset by RESET\_L.

## LAN Ethernet Controller Interrupt 0

ENC038

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>INTR. Interrupt Summary.</b> Read-only. This bit indicates that one or more of the other interrupt bits in this register are set and the associated enable bit or bits in INTEN0 are also set. If INTREN in CMD0 is set to 1 and INTR is set, INTA is active. INTR is read-only. INTR is cleared by clearing all of the active individual interrupt bits that have not been masked out.
30	<b>INTPN. Interrupt Pin Value.</b> Read-only. This bit indicates that the INTA_L pin is asserted, that is, both the INTR bit (INT0[31]) and the INTREN bit (CMD0[1]) are 1.
29:28	Reserved.
27	<b>LCINT. Link Change Interrupt.</b> Read, write 1b to clear; write mode R. This bit is set when the Port Manager detects a change in the link status of the external PHY.
26	<b>APINT5. Auto-Poll Interrupt from Register 5.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 5.
25	<b>APINT4. Auto-Poll Interrupt from Register 4.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 4.
24	<b>APINT3. Auto-Poll Interrupt from Register 3.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 3.
23	Reserved.
22	<b>APINT2. Auto-Poll Interrupt from Register 2.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 2.
21	<b>APINT1. Auto-Poll Interrupt from Register 1.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 1.
20	<b>APINT0. Auto-Poll Interrupt from Register 0.</b> Read, write 1b to clear; write mode R. This bit is set when the Auto-Poll State Machine has detected a change in the external PHY register whose address is stored in Auto-Poll Register 0.
19	<b>MIIPDINT. MII PHY Detect Transition Interrupt.</b> Read, write 1b to clear; write mode R. The MII PHY Detect Transition Interrupt is set by the controller whenever the MIIPD bit in STAT0 transitions from 0 to 1 or vice versa.
18	Reserved.
17	<b>MCCINT. MII Management Command Complete Interrupt.</b> Read, write 1b to clear; write mode R. The MII Management Command Complete Interrupt is set by the controller when a read or write operation to the MII Data Port (PHY Access Register) is complete.
16	<b>MREINT. MII Management Read Error Interrupt.</b> Read, write 1b to clear; write mode R. The MII Read Error interrupt is set by the controller to indicate that the currently read register from the external PHY is invalid. The contents of the PHY Access Register are incorrect and that the operation should be performed again. The indication of an incorrect read comes from the PHY. During the read turnaround time of the MII management frame the external PHY should drive the MDIO pin to a LOW state. If this does not happen, it indicates that the PHY and the controller have lost synchronization.
15	Reserved.
14	<b>SPNDINT. Suspend Interrupt.</b> Read, write 1b to clear; write mode R. This bit is set when a receiver or transmitter suspend operation has finished.

Bits	Description
13	<b>MPINT. Magic Packet™ Interrupt.</b> Read, write 1b to clear; write mode R. Magic Packet Interrupt is set by the controller when the device is in the Magic Packet mode and the controller receives a Magic Packet frame.
12	Reserved.
11	<b>TINT3.</b> Read, write 1b to clear; write mode R. Transmit Interrupt is set by the controller after the OWN bit in the last descriptor of a transmit frame in this particular ring has been cleared to indicate the frame has been copied to the transmit FIFO.
10	<b>TINT2.</b> Read, write 1b to clear; write mode R. Transmit Interrupt is set by the controller after the OWN bit in the last descriptor of a transmit frame in this particular ring has been cleared to indicate the frame has been copied to the transmit FIFO.
9	<b>TINT1.</b> Read, write 1b to clear; write mode R. Transmit Interrupt is set by the controller after the OWN bit in the last descriptor of a transmit frame in this particular ring has been cleared to indicate the frame has been copied to the transmit FIFO.
8	<b>TINT0.</b> Read, write 1b to clear; write mode R. Transmit Interrupt is set by the controller after the OWN bit in the last descriptor of a transmit frame in this particular ring has been cleared to indicate the frame has been copied to the transmit FIFO.
7	<b>UINT. User Interrupt.</b> Read, write 1b to clear; write mode R. UINT is set by the controller after the host has issued a user interrupt command by setting UINTCMD in the CMD0 register.
6:5	Reserved.
4	<b>STINT. Software Timer Interrupt.</b> Read, write 1b to clear; write mode R. The Software Timer interrupt is set by the controller when the Software Timer counts down to 0. The Software Timer immediately loads the contents of the Software Timer Value Register, STVAL, into the Software Timer and begins counting down.
3:1	Reserved.
0	<b>RINT0.</b> Read, write 1b to clear; write mode R. Receive Interrupt is set by the controller after the last descriptor of a receive frame for this ring has been updated by writing a 0 to the OWNership bit.

INT0 identifies the source or sources of an interrupt. With the exception of INTR and INTPN, all bits in this register are “write 1 to clear” so that the CPU can clear the interrupt condition by reading the register and then writing back the same data that it read. Writing a 0 to a bit in this register has no effect.

This register is reset by RESET\_L. In addition, TINTx, RINT0, RINT\_SUM, TINT\_SUM, SPNDINT, and MPINT are reset when the RUN bit is cleared.



## LAN Ethernet Controller Interrupt 0 Enable

ENC040

Default: 0000\_0000h

Attribute: Read-write; write mode R.

Bits	Description
31	<b>VALBIT3. Value bit for byte 3.</b> The value of this bit is written to any bits in the INTEN0 register that correspond to bits in the INTEN0[30:24] bit map field that are set to 1.
30:28	Reserved.
27	<b>LCINTEN. Link Change Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the LCINT bit in INT0 is set.
26	<b>APINT5EN. Auto-Poll Interrupt from Register 5 Enable.</b> When this bit is set, the INTR bit will be set when the APINT5 bit in INT0 is set.
25	<b>APINT4EN. Auto-Poll Interrupt from Register 4 Enable.</b> When this bit is set, the INTR bit will be set when the APINT4 bit in INT0 is set.
24	<b>APINT3EN. Auto-Poll Interrupt from Register 3 Enable.</b> When this bit is set, the INTR bit will be set when the APINT3 bit in INT0 is set.
23	<b>VALBIT2. Value bit for byte 2.</b> The value of this bit is written to any bits in the INTEN0 register that correspond to bits in the INTEN0[22:16] bit map field that are set to 1.
22	<b>APINT2EN. Auto-Poll Interrupt from Register 2 Enable.</b> When this bit is set, the INTR bit will be set when the APINT2 bit in INT0 is set.
21	<b>APINT1EN. Auto-Poll Interrupt from Register 1 Enable.</b> When this bit is set, the INTR bit will be set when the APINT1 bit in INT0 is set.
20	<b>APINT0EN. Auto-Poll Interrupt from Register 0 Enable.</b> When this bit is set, the INTR bit will be set when the APINT0 bit in INT0 is set.
19	<b>MIIPDINTEN. MII PHY Detect Transition Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the MIIPDTINT bit in INT0 is set.
18	Reserved.
17	<b>MCCINTEN. MII Management Command Complete Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the MCCINT bit in INT0 is set.
16	<b>MREINTEN. MII Management Read Error Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the MREINT bit in INT0 is set.
15	<b>VALBIT1. Value bit for byte 1.</b> The value of this bit is written to any bits in the INTEN0 register that correspond to bits in the INTEN0[14:8] bit map field that are set to 1.
14	<b>SPNDINTEN. Suspend Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the SPNDINT bit in INT0 is set.
13	<b>MPINTEN. Magic Packet™ Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the MPINT bit in INT0 is set.
12	Reserved.
11	<b>TINTEN3. Transmit Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the TINT bit for this particular ring in INT0 is set.
10	<b>TINTEN2. Transmit Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the TINT bit for this particular ring in INT0 is set.
9	<b>TINTEN1. Transmit Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the TINT bit for this particular ring in INT0 is set.
8	<b>TINTEN0. Transmit Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the TINT bit for this particular ring in INT0 is set.
7	<b>VALBIT0. Value bit for byte 0.</b> The value of this bit is written to any bits in the INTEN0 register that correspond to bits in the INTEN0[6:0] bit map field that are set to 1.

Bits	Description
6:5	Reserved.
4	<b>STINTEN. Software Timer Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the STINT bit in INT0 is set.
3:1	Reserved.
0	<b>RINTEN0. Receive Interrupt Enable.</b> When this bit is set, the INTR bit will be set when the RINT0 bit in INT0 is set.

This register allows the software to specify which types of interrupt events cause the INTR bit in the Interrupt0 register to be set, which in turn causes the PIRQA\_L pin to be asserted if the INTREN bit in CMD0 is set. Each bit in this register corresponds to a bit in the Interrupt0 register. Setting a bit in this register enables the corresponding bit in the Interrupt0 register to cause the INTR bit to be set.

INTEN0 is a command style register. The high order bit of each byte of this register is a ‘value’ bit that specifies the value to be written to selected bits of the register. The seven low order bits of each byte make up a bit map that selects which register bits will be altered.

This register is reset by RESET\_L.

## LAN Ethernet Controller Logical Address Filter

ENC168

Default: 0000\_0000\_0000\_0000h

Attribute: Read-write; write mode R.

Bits	Description
63:0	<p><b>LADRF. Logical Address Filter, LADRF[63:0].</b> This register contains a 64-bit mask that is used to accept incoming logical (or multicast) addresses. If the first bit in the incoming address (as transmitted on the wire) is a 1, the destination address is a logical address.</p> <p>A logical address is passed through the CRC generator to produce a 32-bit result. The high order 6 bits of this result are used to select one of the 64-bit positions in the Logical Address Filter. If the selected filter bit is set, the address is accepted, and the frame is copied into host system memory.</p> <p>The Logical Address Filter is used in multicast addressing schemes. The acceptance of the incoming frame based on the filter value indicates that the message may be intended for the node. It is the responsibility of the host CPU to compare the destination address of the stored message with a list of acceptable multicast addresses to determine whether or not the message is actually intended for the node.</p>

This register is reset by RESET\_L.

**LAN Ethernet Controller MIB Address****ENC14**

Default: 0000h

Attribute: see below.

Bits	Description
15	<b>MIB_CMD_ACTIVE. Command Active.</b> Read-only. A value of 1 indicates that an access cycle is in progress. The host CPU must not write to this register while this bit is 1. A value of 0 indicates that the previous access cycle has completed and a new access cycle may be started.
14	Reserved.
13	<b>MIB_RD_CMD. Read Command.</b> Read, write 1b to clear; write mode R. Setting this bit to 1 starts a read access cycle.
12	<b>MIB_CLEAR.</b> Read, write 1b to clear; write mode R. Setting this bit to 1 causes all MIB locations to be set to all zeros.
11:6	Reserved.
5:0	<b>MIB_ADDRESS.</b> Read-write; write mode R. The address of the desired table entry.

Used for indirect access to the MIB\_DATA register. This register is reset by RESET\_L.

**LAN Ethernet Controller MIB Data Access****ENC10**

Default: ???\_???h

Attribute: Read-only.

Bits	Description
31:0	<b>MIB_DATA. MIB register contents.</b> May be cleared using MIB_CLEAR bit in MIB_ADDR register.

Access to this data register is indirect by way of the MIB\_ADDR register. Protocol is similar to PHY access protocol except no interrupt is available as accesses end within a short time. The access protocol for a read access is:

1. Read MIB\_ADDR until MIB\_CMD\_ACTIVE is 0.
2. Write MIB\_ADDR with MIB\_RD\_CMD = 1 and MIB\_ADDR set to the desired address.
3. Read MIB\_ADDR until MIB\_CMD\_ACTIVE is 0. The value is in the MIB\_DATA register.

Data transfers are 32 bits wide for MIB.

**LAN Ethernet Controller Physical Address****ENC160**

Default: 0000\_0000\_0000h

Attribute: Read-write; write mode R.

Bits	Description
47:0	<p><b>MAC Physical Address, PADR[47:0].</b> This register contains 48-bit, globally unique station address assigned to this device. If the least significant bit of the first byte of a received frame is 0, the destination address of the frame is a unicast address, which is compared with the contents of the PADR. If this bit is 0 and the frame's destination address exactly matches the contents of PADR, the frame is accepted and copied into the host system memory.</p> <p>The byte order is such that PADR[7:0] corresponds to the first address byte transferred over the network.</p> <p>Unicast address matching can be disabled by setting the Disable Receive Physical Address bit (DCRVPA, bit 18 in CMD2). If DRCVPA is set to 1, a match of a frame's destination address with the contents of PADR does not cause the frame to be accepted and copied into the host memory.</p>

This register is reset by RESET\_L.

**LAN Ethernet Controller PHY Access****ENC0D0**

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31	<b>PHY_CMD_ACTIVE. PHY Command Active.</b> Read-only. This bit is automatically set to 1 while the PHY access logic is busy. The content of the PHY_DATA field is invalid while this bit is 1.
30	<b>PHY_WR_CMD. PHY Write Command.</b> Write-only. Setting this bit to 1 starts the process of writing the data in the PHY_DATA field to the external PHY register specified by the PHY_ADDR and PHY_REG_ADDR fields.
29	<b>PHY_RD_CMD. PHY Read Command.</b> Write-only. Setting this bit to 1 starts the process of reading the external PHY register specified by the PHY_ADDR and PHY_REG_ADDR fields.
28	<b>PHY_RD_ERR. PHY Read Error.</b> Read-only. This bit is automatically set if the previous command was PHY_RD_CMD and a read error was detected.
27	Reserved.
26	<b>PHY_PRE_SUP. Preamble Suppression.</b> Read-write. If this bit is set, the MII Management Frame is sent without a preamble. Before setting this bit the host CPU must make sure that the external PHY addressed by the PHY_ADDR field is capable of accepting MII Management Frames without preambles.
25:21	<b>PHY_ADDR. PHY Address.</b> Read-write. The address of the external PHY device to be accessed.
20:16	<b>PHY_REG_ADDR. PHY Register Address.</b> Read-write. The address of the register in the external PHY device to be accessed.
15:0	<b>PHY_DATA. PHY Data.</b> Read-write. Data written to or read from the external PHY register specified by PHY_ADDR and PHY_REG_ADDR.

This register gives the host CPU indirect access to the MII Management Bus (MDC/MDIO). Through this register the host CPU can read or write any external PHY register that is accessible through the

MII Management Bus. When the host CPU sets the PHY Write Command or PHY Read Command bit, the appropriate management frame is sent over the management bus.

The PHY access protocol is:

1. Read PHY\_ACCESS until PHY\_CMD\_ACTIVE is 0.
2. Write PHY\_WR\_CMD or PHY\_RD\_CMD, PHY\_PRE\_SUP, PHY\_ADDR, PHY\_REG\_ADDR and, if write, PHY\_DATA to PHY\_ACCESS register.
3. If read, wait for MCCINT interrupt or read PHY\_ACCESS register until PHY\_CMD\_ACTIVE is 0.

This register is reset by RESET\_L.

### LAN Ethernet Controller OnNow Pattern 0

ENC190

Default: 0000\_000?h

Attribute: see below.

Bits	Description
31	<b>PMR_ACTIVE.</b> Read-only. Read-only copy of PMAT_MODE.
30	<b>PMR_WR_CMD. PMR Write Command.</b> Write-only; write mode 3. Setting this bit to 1 causes the data in the PMAT1 register and the PMR_B4 field of this register to be written to the word in the Pattern Match RAM specified by the PMR_ADDR field of this register.
29	<b>PMR_RD_CMD. PMR Read Command.</b> Write-only; write mode 3. Setting this bit to 1 causes the word in the Pattern Match RAM specified by the PMR_ADDR field of this register to be read into the PMAT1 register and the PMR_B4 field of this register.
28:23	Reserved.
22:16	<b>PMR_ADDR. Pattern Match RAM Address.</b> Read-write; write mode 3. These bits are the Pattern Match RAM address to be written to or read from
15:8	Reserved.
7:0	<b>PMR_B4. Pattern Match RAM Byte 4.</b> Read-write; write mode 3. This byte is written into or read from Byte 4 of the selected word of the Pattern Match RAM.

This register is used to control and indirectly access the Pattern Match RAM (PMR).

Access protocol is:

1. Ensure PMAT\_MODE is 0 either by program logic or by reading CMD7 or PMAT0. The PMR\_ACTIVE bit is a read-only copy of PMAT\_MODE.
2. For write, write data to PMAT1 and then to PMAT0 with PMR\_WR\_CMD set to 1 and PMR\_ADDR set to the desired memory address. Specify the bank with PMR\_BANK.
3. For read, write PMAT0 with PMR\_RD\_CMD set to 1 and PMR\_ADDR set to the desired memory address.
4. For read, read data from PMAT0 and PMAT1.

Bits [31:16] of this register are reset by RESET\_L. Bits [15:0] are never reset.

**LAN Ethernet Controller OnNow Pattern 1****ENC194**

Default: ???\_???h

Attribute: Read-write; write mode 3.

Bits	Description
31:24	<b>PMR_B3. Pattern Match RAM Byte 3.</b> This byte is written into or read from Byte 3 of the selected word of the Pattern Match RAM.
23:16	<b>PMR_B2. Pattern Match RAM Byte 2.</b> This byte is written into or read from Byte 2 of the selected word of the Pattern Match RAM.
15:8	<b>PMR_B1. Pattern Match RAM Byte 1.</b> This byte is written into or read from Byte 1 of the selected word of the Pattern Match RAM.
7:0	<b>PMR_B0. Pattern Match RAM Byte 0.</b> This byte is written into or read from Byte 0 of the selected word of the Pattern Match RAM.

This register is used to indirectly access the Pattern Match RAM (PMR). This register is never reset.

**LAN Ethernet Controller Receive Ring Length****ENC150**

Default: 0000h

Attribute: Read-write; write mode N.

Bits	Description
15:0	<b>RCV_RING_LEN.</b> This value indicates the number of descriptors in the receive descriptor ring.

This register is reset by RESET\_L.

**LAN Ethernet Controller SRAM Boundary****ENC17A**

Default: 0008h

Attribute: Read-write; write mode N.

Bits	Description
15:5	Reserved.
4:0	<b>SRAM_BND.</b> Specifies the size of the transmit buffer portion of the SRAM in units of 512-byte pages. For example, if SRAM_BND is set to 10, then 5120 bytes of the SRAM is allocated for the transmit buffer and the rest is allocated for the receive buffer. The transmit buffer in the SRAM begins at address 0 and ends at the address (SRAM_BND*512)-1. Therefore, the receive buffer always begins on a 512-byte boundary. The minimum allowed number of pages for normal network operation is four, and the maximum is SRAM_SIZE – 4. Values larger than this cause incorrect behavior.

This register is reset by RESET\_L.

**LAN Ethernet Controller SRAM Size****ENC178**

Default: 0010h

Attribute: Read-write; write mode N.

Bits	Description
15:5	Reserved.
4:0	<b>SRAM_SIZE.</b> Specifies the size of the internal SRAM in units of 512-byte pages. SRAM_SIZE should normally be set to 16, since the size of the SRAM is 8192 bytes. SRAM_SIZE can be set to a lower number for debugging purposes.

This register is reset by RESET\_L.

## LAN Ethernet Controller Status 0

ENC030

Default: 0001\_0000h

Attribute: see below.

Bits	Description
31:13	Reserved.
12	<b>PMAT_DET. Pattern Match Detected.</b> Read, write 1b to clear; write mode N. This bit indicates that an OnNow pattern match has occurred while the device was in the OnNow pattern match mode. This bit can be cleared to 0 by writing 1 to STAT0, bit 12.
11	<b>MP_DET. Magic Packet™ Frame Detected.</b> Read, write 1b to clear; write mode N. This bit indicates that a Magic Packet pattern match has occurred while the device was in the Magic Packet mode. This bit can be cleared to 0 by writing 1 to STAT0, bit 11.
10	<b>LC_DET. Link Change Detected.</b> Read, write 1b to clear; write mode N. This bit indicates that a change in the link status of the external PHY device has been detected while the device was in the Link Change Wake-up mode. This bit can be cleared to 0 by writing 1 to STAT0, bit 10.
9:7	<b>SPEED. Speed.</b> Read-only. This field indicates the bit rate at which the network is running. The following encoding is used: 000=Unknown, 001=Reserved, 010=10 Mb/s, 011=100 Mb/s, 100-111 are Reserved.
6	<b>FULL_DUPLEX. Full Duplex.</b> Read-only. This bit is set when the device is operating in full-duplex mode.
5	<b>LINK_STATUS. Link Status.</b> Read-only. This bit is set to the value of the Link Status bit in the status register (R1) of the default external PHY. (The default external PHY is the PHY addressed by the AP_PHY0_ADDR field of the AUTOPOLL0 Register.) This bit is updated each time the external PHY's status register is read, either by the Auto-Poll State Machine, by the Network Port Manager, or by a CPU-initiated read. However, when the Force Link Status bit in CMD3 is set to 1, this bit is forced to 1, regardless of the contents of the external PHY's status register.
4	<b>AUTONEG_COMPLETE. Auto-negotiation Complete.</b> Read-only. This bit is set to the value of the Auto-Negotiation Complete bit in register 1 of the external PHY as determined by the most recent Port Manager polling cycle.
3	<b>MIIPD. MII PHY Detect.</b> Read-only. MIIPD reflects the quiescent state of the MDIO pin. MIIPD is continuously updated whenever there is no management operation in progress on the MII interface. When a management operation begins on the interface, the state of MIIPD is preserved until the operation ends, when the quiescent state is again monitored and continuously updates the MIIPD bit. When the MDIO pin is at a quiescent LOW state, MIIPD is cleared to 0. When the MDIO pin is at a quiescent HIGH state, MIIPD is set to 1. Any transition on the MIIPD bit sets the MIIPDTINT bit in INT0.
2	<b>RX_SUSPENDED. Receiver Suspended.</b> Read-only. This bit has the value 1 while the receiver is suspended, and it has the value 0 when the receiver is not suspended.
1	<b>TX_SUSPENDED. Transmitter Suspended.</b> Read-only. This bit has the value 1 while the transmitter is suspended, and it has the value 0 when the transmitter is not suspended.
0	<b>RUNNING.</b> Read-only. This bit is a read-only alias of the RUN bit in CMD0. When this bit is set, the device is enabled to transmit and receive frames and process descriptors. Note that even though RUNNING is set, the receiver or transmitter might be disabled because RX_SPND or TX_SPND is set (in CMD0).

Bits [12:10] of this register are reset by Power-On reset. Bits [9:0] of this register are reset by RESET\_L.



**LAN Ethernet Controller Spare****ENC000**

Default: 0000\_0000h

Attribute: Read-only.

Bits	Description
31:3	Reserved.
1:0	<b>SPARE.</b> These bits are hardwired to 0h.

**LAN Ethernet Controller Software Timer Value****ENC0D8**

Default: FFFFh

Attribute: Read-write; write mode R.

Bits	Description
15:0	<b>STVAL. Software Timer Value.</b> STVAL controls the maximum time for the Software Timer to count before generating the STINT (INTR, bit 4) interrupt. The Software Timer is a free-running timer that is started upon the first write to STVAL. After the first write, the Software Timer continually counts and sets the STINT interrupt at the STVAL period. The STVAL value is interpreted as an unsigned number with a resolution of 10.24 $\mu$ s. For instance, if STVAL is set to 48,828 (0BEBCh), the Software Timer period is 0.5 s. The default value (0FFFFh) corresponds to 0.6710784 s. Setting STVAL to a value of 0 results in erratic behavior.

This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 0 Length****ENC140**

Default: 0000h

Attribute: Read-write; write mode.

Bits	Description
15:0	<b>XMT_RING0_LEN.</b> This value indicates the number of descriptors in Transmit Descriptor Ring 0.

This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 1 Length****ENC144**

Default: 0000h

Attribute: Read-write; write mode.

Bits	Description
15:0	<b>XMT_RING1_LEN.</b> This value indicates the number of descriptors in Transmit Descriptor Ring 1.

This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 2 Length****ENC148**

Default: 0000h

Attribute: Read-write; write mode.

Bits	Description
15:0	<b>XMT_RING2_LEN.</b> This value indicates the number of descriptors in Transmit Descriptor Ring 2.

This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring 3 Length****ENC14C**

Default: 0000h

Attribute: Read-write; write mode.

Bits	Description
15:0	<b>XMT_RING3_LEN.</b> This value indicates the number of descriptors in Transmit Descriptor Ring 3.

This register is reset by RESET\_L.

**LAN Ethernet Controller Transmit Ring Limit****ENC07C**

Default: 0000\_0000h

Attribute: see below.

Bits	Description
31:24	<b>XMT_RING3_LIMIT. Transmit ring3 limit.</b> Read-only. Always 0. Ring3 is highest priority and cannot be locked out.
23:16	<b>XMT_RING2_LIMIT. Transmit ring2 limit.</b> Read-write; write mode N. No limit if 0.
15:7	<b>XMT_RING1_LIMIT. Transmit ring1 limit.</b> Read-write; write mode N. No limit if 0.
7:0	<b>XMT_RING0_LIMIT. Transmit ring0 limit.</b> Read-write; write mode N. No limit if 0.

Event counters limit the number of consecutive high priority transmissions to prevent the lower priority transmit rings from being locked out by higher priority rings. See Transmitter Fairness Algorithm section. This register is reset by RESET\_L.

**4.10.4 Receive Descriptors**

Figure 32 shows the format of receive descriptors. Figure 33 shows the format of the FLAGS field of a receive descriptor.



Table 66. Receive Descriptor Bit Definitions (Continued)

Offset	Bit	Name	Description
8	31	<b>OWN</b>	This bit indicates whether the descriptor entry is owned by the host (OWN = 0) or by the network controller (OWN = 1). The host sets the OWN bit after it has emptied the buffer pointed to by the descriptor entry. The network controller clears the OWN bit after filling the buffer that the descriptor points to. Both the network controller and the host must not alter a descriptor entry after it has relinquished ownership.
8	30	<b>ERR</b>	<b>Error Summary.</b> ERR is the OR of FRAM, OFLO, CRC, and BUFF. ERR is set by the network controller and cleared by the host.
8	29	<b>FRAM</b>	<b>Framing error</b> indicates that the incoming frame contains a non-integer multiple of eight bits and there was an FCS error. If there was no FCS error on the incoming frame, then FRAM is not set even if there was a non-integer multiple of eight bits in the frame. FRAM is not valid in internal loopback mode. FRAM is valid only when ENP is set and OFLO is not. FRAM is set by the network controller and cleared by the host.
8	28	<b>OFLO</b>	<b>Overflow error</b> indicates that the receiver has lost all or part of the incoming frame, due to an inability to move data from the receive FIFO into a memory buffer before the internal FIFO overflowed. OFLO is set by the network controller and cleared by the host.
8	27	<b>CRC</b>	CRC indicates that the receiver has detected a CRC (FCS) error on the incoming frame. CRC is valid only when ENP is set and OFLO is not. CRC is set by the network controller and cleared by the host. CRC is also set when the controller receives an RX_ER indication from the external PHY through the MII.
8	26	<b>BUFF</b>	<b>Buffer Error.</b> If this bit is set, the end of this frame was lost because the next descriptor was not available when it was needed. The RcvMissPkts MIB counter is also incremented when this error occurs.
8	25	<b>STP</b>	<b>Start of Packet</b> indicates that this is the first buffer used by the network controller for this frame. If STP and ENP are both set to 1, the frame fits into a single buffer. Otherwise, the frame is spread over more than one buffer. STP is set by the controller and cleared by the host.
8	24	<b>ENP</b>	<b>End of Packet</b> indicates that this is the last buffer used by the network controller for this frame. It is used for data chaining buffers. If both STP and ENP are set, the frame fits into one buffer and there is no data chaining. ENP is set by the controller and cleared by the host.
8	23		Reserved
8	22	<b>PAM</b>	<b>Physical Address Match</b> is set by the network controller when it accepts the received frame due to a match of the frame's destination address with the content of the physical address register. PAM is valid only when ENP is set. PAM is set by the network controller and cleared by the host.

**Table 66. Receive Descriptor Bit Definitions (Continued)**

Offset	Bit	Name	Description
8	21	<b>LAFM</b>	<b>Logical Address Filter Match</b> is set by the network controller when it accepts the received frame based on the value in the logical address filter register. LAFM is valid only when ENP is set. LAFM is set by the controller and cleared by the host.  Note that if DRCVBC (CMD2, bit 17) is cleared to 0, only BAM, but not LAFM is set when a Broadcast frame is received, even if the Logical Address Filter is programmed in such a way that a Broadcast frame would pass the hash filter. If DRCVBC is set to 1 and the Logical Address Filter is programmed in such a way that a Broadcast frame would pass the hash filter, LAFM is set on the reception of a Broadcast frame.
8	20	<b>BAM</b>	<b>Broadcast Address Match</b> is set by the network controller when it accepts the received frame, because the frame's destination address is of the type 'Broadcast.' BAM is valid only when ENP is set. BAM is set by the controller and cleared by the host.
8	19:18	<b>TT[1:0]</b>	<b>VLAN Tag Type.</b> Indicates what type of VLAN tag, if any, is included in the received frame.  00 = Reserved 01 = Frame is Untagged 10 = Frame is Priority-tagged 11 = Frame is VLAN-tagged
8	17:16		Reserved.
8	15:0	<b>BCNT</b>	<b>Buffer Byte Count</b> is the length of the buffer pointed to by this descriptor, expressed as an unsigned integer. This field is written by the host and unchanged by the network controller.
0Ch	31:0	<b>RBADR[31:0]</b>	<b>Receive Buffer Address.</b> This field contains the low order bits of the address of the receive buffer that is associated with this descriptor.

### 4.10.5 Transmit Descriptors

Figure 34 shows the format of transmit descriptors. Figure 35 shows the format of the TFLAGS1 field of a transmit descriptor.

Offset	31	30	29	28	27:26	25	24	23	22	21	20	19:18	17	16	15-0	
00h	TFLAGS1										RES		BCNT[15:0]			
04h	RES										TCC[1:0]		TCI[15:0]			
08h	TBADR[31:0]										USER SPACE					
0Ch																

**Figure 34. Transmit Descriptor Format**

The name TFLAGS1 in this figure stands for a collection of bits that are displayed in Figure 35 below.

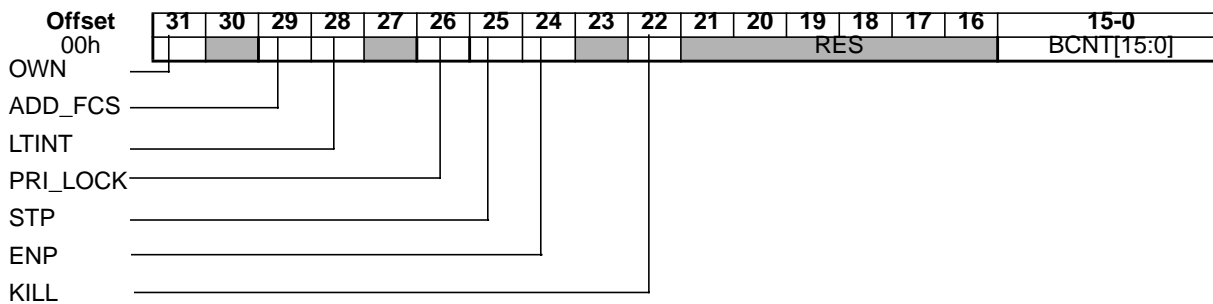


Figure 35. Format of TFLAGS1 in Transmit Descriptor, Offset 00h

Table 67 on page 350 shows the transmit descriptor bit definitions.

Table 67. Transmit Descriptor Bit Definitions

Offset	Bit	Name	Description
0	31	<b>OWN</b>	This bit indicates whether the descriptor entry is owned by the host (OWN = 0) or by the network controller (OWN = 1). The host sets the OWN bit after filling the buffer pointed to by the descriptor entry. The controller clears the OWN bit after transmitting the contents of the buffer. Both the controller and the host must not alter a descriptor entry after it has relinquished ownership.
0	30		Reserved location.
0	29	<b>ADD_FCS</b>	ADD_FCS dynamically controls the generation of FCS on a frame by frame basis. This bit should be set with the ENP bit. However, for backward compatibility, it is recommended that this bit be set for every descriptor of the intended frame. When ADD_FCS is set, the state of DXMTFCS is ignored and transmitter FCS generation is activated. When ADD_FCS is cleared to 0, FCS generation is controlled by DXMTFCS. When APAD_XMT (CMD2, bit 6) is set to 1, the setting of ADD_FCS has no effect. ADD_FCS is set by the host, and is not changed by the network controller.
0	28	<b>LTINT</b>	<b>Last Transmit Interrupt.</b> When enabled by the LTINTEN bit (CMD2, bit 9), LTINT is used to suppress interrupts after selected frames have been copied to the transmit FIFO. When LTINT is cleared to 0 and ENP is set to 1, the network controller does not set TINTx (INT0, bits 8-11) after the corresponding frame has been copied to the transmit FIFO. TINTx is only set when the last descriptor of a frame has both LTINT and ENP set to 1. When LTINTEN is cleared to 0, the LTINT bit is ignored.
0	27		Reserved.
0	26	<b>PRI_LOCK</b>	<b>Priority Lock.</b> If PRI_LOCK is set on the first descriptor of a frame (when STP is set), the transmit ring arbitrator is forced to take the next frame from the same ring as this one. If the first descriptor of that frame also has PRI_LOCK set, the process continues. When PRI_LOCK is not set in the first descriptor of a frame, the next frame comes from the ring chosen by the arbiter. PRI_LOCK is ignored if STP is not set.

**Table 67. Transmit Descriptor Bit Definitions (Continued)**

Offset	Bit	Name	Description
0	25	<b>STP</b>	Start of Packet indicates that this is the first buffer to be used by the network controller for this frame. It is used for data chaining buffers. The STP bit must be set in the first buffer of the frame, or the controller skips over the descriptor and polls the next descriptor(s) until the OWN and STP bits are set. STP is set by the host and is not changed by the controller.
0	24	<b>ENP</b>	End of Packet indicates that this is the last buffer to be used by the controller for this frame. It is used for data chaining buffers. If both STP and ENP are set, the frame fits into one buffer and there is no data chaining. ENP is set by the host and is not changed by the controller.
0	23		Reserved
0	22	<b>KILL</b>	This bit causes the transmission of the corresponding frame to be aborted. If the transmitter has not started sending the frame at the time that the descriptor processing logic encounters the KILL bit, no portion of the frame is sent. If part of the frame has been sent, the frame is truncated, and an FCS field containing the inverse of the correct CRC is appended.
0	21:16		Reserved.
0	15:0	<b>BCNT</b>	Buffer Byte Count is the usable length of the buffer pointed to by this descriptor, expressed as an unsigned integer. This is the number of bytes from this buffer that are to be transmitted by the network controller. This field is written by the host and is not changed by the network controller. There are no minimum buffer size restrictions.
4	31:18		Reserved.
4	17:16	<b>TCC[1:0]</b>	<b>VLAN Tag Control Command.</b> This field contains a command that causes the transmitter to add, modify, or delete a VLAN tag or to transmit the frame unaltered.  00 = Transmit the data in the buffer unaltered 01 = Delete the VLAN tag (the 13th through 16th bytes of the frame) 10 = Insert a VLAN tag containing the TCI field from the descriptor 11 = Replace the TCI field of the frame with TCI data from the descriptor
4	15:0	<b>TCI[15:0]</b>	<b>Tag Control Information.</b> If the contents of the TCC field are 10 or 11, the controller transmits the contents of the TCI field as bytes 15 and 16 of the outgoing frame.
8	31:0	<b>TBADR[31:0]</b>	<b>Transmit Buffer Address.</b> This field contains the low order bits of the address of the Transmit buffer that is associated with this descriptor.
0Ch	31:0	<b>USER SPACE</b>	<b>User Space.</b> Reserved for user defined data.





## Chapter 5 Electrical Data

### 5.1 Absolute Ratings

The IC is not designed to operate beyond the parameters shown in Table 68.

**Note:** *The absolute ratings in the following table and associated conditions must be adhered to in order to avoid damage to the IC and motherboard. Systems using the IC must be designed to ensure that the power supply and system logic board do not violate these parameters.*

**VIOLATION OF THE ABSOLUTE RATINGS WILL VOID THE PRODUCT WARRANTY.**

**Table 68. Absolute Ratings**

Parameter	Minimum	Maximum	Comments
VDD_IO, VDD_IOX, VDD_RTC, VDD_USB	-0.5 V	3.6 V	
VDD_CORE, VDD_COREX, VDD_USBA	-0.5 V	2.0 V	
VDD_LDT	-0.5 V	1.7 V	
VDD_REF	-0.5 V	5.25 V	
T <sub>CASE</sub>	-65 °C	85 °C	(under BIAS)
T <sub>STORAGE</sub>	-65 °C	150 °C	
<b>Notes:</b>			
1. This table contains preliminary information, which is subject to change.			

## 5.2 Operating Ranges

The IC is designed to provide functional operation if the voltage and temperature parameters are within the limits defined in Table 69.

**Table 69. Operating Ranges**

Parameter	Minimum	Typical	Maximum
VDD_IO, VDD_IOX, VDD_USB	3.135 V	3.3 V	3.465 V
VDD_CORE, VDD_COREX, VDD_USBA	1.71 V	1.8 V	1.89 V
VDD_LDT	1.14 V	1.2 V	1.26 V
VDD_RTC	2.5 V	3.3 V	3.465 V
VDD_REF	4.75 V	5.0 V	5.25 V
T <sub>CASE</sub>	0 °C		70 °C

**Notes:**

1. This table contains preliminary information, which is subject to change.

## 5.3 Current Consumption

Table 70 provides current consumption of the IC while it is operational.

**Table 70. Current Consumption**

Supply	Max Current			
	S0	S1	S3/S4/S5	G3
VDD_IO	60 mA	20 mA	N/A	N/A
VDD_IOX	5 mA	5 mA	5 mA	N/A
VDD_CORE	300 mA	300 mA	N/A	N/A
VDD_COREX	100 mA	100 mA	25 mA	N/A
VDD_USB	150 mA <sup>2</sup>	100 mA	1 mA	N/A
VDD_USBA	250 mA <sup>2</sup>	150 mA	0.5 mA	N/A
VDD_LDT	100 mA	100 mA	N/A	N/A
VDD_RTC	N/A	N/A	N/A	8 μA
VDD_REF	1 mA	1 mA	1 mA	N/A

**Notes:**

1. This table contains preliminary information, which is subject to change.
2. This number assumes 6 USB ports transmitting and receiving high-speed traffic.

## 5.4 Signal Groups

Table 71 defines groups of I/O signals with respect to their electrical behavior.

**Table 71. Signal Groups**

Signal Group	Signals
A	<b>Host HyperTransport™ Technology Interface</b> LDTREQ_L
	<b>Secondary PCI Interface</b> AD[31:0], CBE_L[3:0], DEVSEL_L, FRAME_L, GNT_L[6:0], IRDY_L, PAR, PCLK, PERR_L, PGNT_L, PIRQ[A, B, C, D]_L, PREQ_L, REQ_L[6:0], SERR_L, STOP_L, TRDY_L
	<b>LPC Bus and Legacy Support Signals</b> LAD[3:0], LDRQ_L[1:0], LFRAME_L, SPKR
	<b>Enhanced IDE Interface</b> DADDR[P, S][2:0], DCS1[P, S]_L, DCS3[P, S]_L, DDACK[P, S]_L, DDRQ[P, S], DIOW[P, S]_L, DRST[P, S]_L
	<b>System Management Signals</b> ACAV, AGPSTOP_L, BATLOW_L, C32KHZ, CLKRUN_L, CPUSLEEP_L, CPUSTOP_L, DCSTOP_L, EXTSMI_L, FANCON[1:0], FANRPM, GPIO[31:26, 17, 16, 14, 8, 4], INTIRQ8_L, IRQ1, IRQ6, IRQ12, IRQ14, IRQ15, KA20G, KBRC_L, LID, PCISTOP_L, PME_L, PNPIRQ[2:0], PRDY, PWRBTN_L, PWROK, PWRON_L, RESET_L, RI_L, RPWRON, SERIRQ, SLPBTN_L, SMBALERT[1:0]_L, SMBUSC1, SMBUSD1, SUSPEND_L, THERM_L
	<b>Universal Serial Bus Interface</b> USBOC_L[1:0]
	<b>AC '97 Interface</b> ACCLK, ACRST_L, ACSDI[1:0], ACSDO, ACSYNC
	<b>MII Interface</b> MII_TX_CLK, MII_TXD[3:0], MII_TX_EN, MII_COL, MII_CRS, MII_RX_CLK, MII_RXD[3:0], MII_RX_DV, MII_RX_ERR, MII_PHY_RST, MII_MDC, MII_MDIO
	<b>Test</b> TEST_L
	<b>Miscellaneous</b> STRAPH[3:0], STRAPL[3:0]
B	<b>Host HyperTransport Technology Interface</b> LRXCAD_H/L[7:0], LRXCLK_H/L, LRXCTL_H/L, LTXCAD_H/L[7:0], LTXCLK_H/L, LTXCTL_H/L
C	<b>Enhanced IDE Interface</b> DDATA[P, S][15:0], DIOR[P, S]_L, DRDY[P, S]

Table 71. Signal Groups (Continued)

Signal Group	Signals
D	<b>Universal Serial Bus Interface</b> USB0_H/L[2:0], USB1_H/L[2:0]
E	<b>Host HyperTransport™ Technology Interface</b> LDTSTOP_L, LDTRST_L
	<b>System Management Signals</b> SMBUSC0, SMBUSD0, THERMTRIP_L
F	<b>System Management Signals</b> INTRUDER_L
G	<b>System Management Signals</b> RTCX_IN, RTCX_OUT
H	<b>LPC Bus and Legacy Support Signals</b> OSC
	<b>Universal Serial Bus Interface</b> USBCLK

## 5.5 DC Characteristics

See the HyperTransport™ Technology Electrical Specification for the DC characteristics of the HyperTransport interface signals of signal group B.

See the USB specification for the DC characteristics of the USB interface signals of signal group D.

Table 72. DC Characteristics for Signal Group A

Symbol	Parameter Description	Min	Max	Units
V <sub>IL</sub>	Input Low voltage	-0.3	0.3 × VDD_IO	V
V <sub>IH</sub>	Input High voltage	0.6 × VDD_IO	VDD_REF + 0.5	V
V <sub>OL</sub>	Output Low voltage; I <sub>OUT</sub> = 2 mA <sup>2, 3</sup>		0.1 × VDD_IO	V
V <sub>OH</sub>	Output High voltage; I <sub>OUT</sub> = -2 mA <sup>2, 3</sup>	0.9 × VDD_IO		V
I <sub>LI</sub>	Input leakage current		+/- 10	μA
C <sub>IN</sub>	Input capacitance		8	pF

**Notes:**

1. This table contains preliminary information, which is subject to change.
2. All Secondary PCI, LPC bus, and AC '97 interface signals support an output current of I<sub>OUT</sub> = +/- 6 mA.
3. The SMBus 2.0 interface signals SMBUSC1, SMBUSD1 support an output current of I<sub>OUT</sub> = +/- 4 mA.

**Table 73. DC Characteristics for Signal Group C**

Symbol	Parameter Description	Min	Max	Units
V <sub>IL</sub>	Input Low voltage	-0.3	0.3 × VDD_IO	V
V <sub>IH</sub>	Input High voltage	0.6 × VDD_IO	VDD_REF + 0.5	V
V <sub>OL</sub>	Output Low voltage; I <sub>OUT</sub> = 400 μA		0.1 × VDD_IO	V
V <sub>OH</sub>	Output High voltage; I <sub>OUT</sub> = -4 mA	0.9 × VDD_IO		V
I <sub>LI</sub>	Input leakage current		+/- 10	μA
C <sub>IN</sub>	Input capacitance		8	pF

**Notes:**  
1. This table contains preliminary information, which is subject to change.

**Table 74. DC Characteristics for Signal Group E**

Symbol	Parameter Description	Min	Max	Units
V <sub>IL</sub>	Input Low voltage	-0.3	0.3 × VDD_IO	V
V <sub>IH</sub>	Input High voltage	0.6 × VDD_IO	VDD_IO + 0.3	V
V <sub>OL</sub>	Output Low voltage; I <sub>OUT</sub> = 2 mA		0.1 × VDD_IO	V
V <sub>OH</sub>	Output High voltage; I <sub>OUT</sub> = -2 mA	0.9 × VDD_IO		V
I <sub>LI</sub>	Input leakage current		+/- 10	μA
C <sub>IN</sub>	Input capacitance		8	pF

**Notes:**  
1. This table contains preliminary information, which is subject to change.

**Table 75. DC Characteristics for Signal Group F**

Symbol	Parameter Description	Min	Max	Units
V <sub>IL</sub>	Input Low voltage	-0.1	0.25	V
V <sub>IH</sub>	Input High voltage	1.0	VDD_IO + 0.3	V
I <sub>LI</sub>	Input leakage current <sup>2</sup>		+/- 1	μA
C <sub>IN</sub>	Input capacitance		8	pF

**Notes:**  
1. This table contains preliminary information, which is subject to change.  
2. For V<sub>I</sub> = VSS or V<sub>I</sub> > 2 V.

**Table 76. DC Characteristics for Signal Group G**

Symbol	Parameter Description	Min	Max	Units
I <sub>LI</sub>	Input leakage current	-10.0	1.0	μA

**Table 76. DC Characteristics for Signal Group G**

Symbol	Parameter Description	Min	Max	Units
C <sub>IN</sub>	Input capacitance		4	pF
<b>Notes:</b>				
1. This table contains preliminary information, which is subject to change.				
2. This table applies to RTCX_IN only.				

**Table 77. DC Characteristics for Signal Group H**

Symbol	Parameter Description	Min	Max	Units
V <sub>IL</sub>	Input Low voltage	-0.3	0.3 × VDD_IO	V
V <sub>IH</sub>	Input High voltage	0.6 × VDD_IO	VDD_IO + 0.3	V
I <sub>LI</sub>	Input leakage current		+/- 10	μA
C <sub>IN</sub>	Input capacitance		8	pF
<b>Notes:</b>				
1. This table contains preliminary information, which is subject to change.				

## 5.6 AC Characteristics

See the HyperTransport™ Technology Electrical Specification for the AC characteristics of the HyperTransport interface signals of signal groups A and B.

AC characteristics for the PCI interface signals, LPC bus, legacy support signals, and system management signals of signal group A match those of the PCI specification. See Table 78 on page 359 for the AC characteristics of PCLK. See Table 79 on page 359 for the AC characteristics of OSC.

See the ATA specification for the AC characteristics of the Enhanced IDE signals of signal groups A and C.

See the USB specification for the AC characteristics of the USB interface signals of signal groups D and H. See Table 80 on page 360 for the AC characteristics of USBCLK.

See the AC '97 specification for the AC characteristics of the AC '97 interface signals of signal group A.

See the IEEE 802.3 specification for the AC characteristics of the MII interface signals of signal group A.

See the SMBus specification for the AC characteristics of the SMBus interface signals of signal groups A and F.

**Table 78. AC Characteristics for PCLK**

Symbol	Parameter Description	Min	Typ	Max	Units
f	Frequency		33.333		MHz
t <sub>S</sub>	Slew Rate <sup>2</sup>	1		4	V/ns
t <sub>D</sub>	Duty Cycle <sup>3</sup>	45		55	%
t <sub>JC</sub>	Jitter, Cycle-to-Cycle <sup>4</sup>	0		250	ps
t <sub>JA</sub>	Jitter, Accumulated	-1000		1000	ps

**Notes:**

1. This table contains preliminary information, which is subject to change.
2. Measured between 20% and 60%
3. Measured on rising and falling edge at 1.5 V
4. Measured on rising edge at 1.5V. Maximum difference of cycle time between two adjacent cycles.

**Table 79. AC Characteristics for the OSC**

Symbol	Parameter Description	Min	Typ	Max	Units
f	Frequency		14.318		MHz
t <sub>S</sub>	Slew Rate <sup>2</sup>	0.5		2	V/ns
t <sub>D</sub>	Duty Cycle <sup>3</sup>	45		55	%
t <sub>JC</sub>	Jitter, Cycle-to-Cycle <sup>4</sup>	0	500	1000	ps
t <sub>JA</sub>	Jitter, Accumulated	-1000		1000	ps

**Notes:**

1. This table contains preliminary information, which is subject to change.
2. Measured between 20% and 80%
3. Measured on rising and falling edge at 1.5 V.
4. Measured on rising edge at 1.5 V. Maximum difference of cycle time between two adjacent cycles.

**Table 80. AC Characteristics for USBCLK**

Symbol	Parameter Description	Min	Typ	Max	Units
f	Frequency		48.008		MHz
t <sub>S</sub>	Slew Rate <sup>2</sup>	0.5		2	V/ns
t <sub>D</sub>	Duty Cycle <sup>3</sup>	45		55	%
t <sub>JC</sub>	Jitter, Cycle-to-Cycle <sup>4</sup>	0		200	ps
t <sub>JA</sub>	Jitter, Accumulated	-1000		1000	ps

**Notes:**

1. This table contains preliminary information, which is subject to change.
2. Measured between 20% and 80%.
3. Measured on rising and falling edge at 1.5 V.
4. Measured on rising edge at 1.5V. Maximum difference of cycle time between two adjacent cycles.



# Chapter 6 Pin Designations

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		
A	NC10	NC12	NC19	NC30	NC21	NC28	DAD DRS1	DRD YS	DDAT AS0	DDAT AS12	DDAT AS4	DDAT AS8	DAD DRP2	IRQ14	DDR QP	DDAT AP3	DDAT AP9	DDAT AP3	DDAT AP9	LDRQ_L[0]	LAD2	LAD3	IRQ6	VDD_RTC	LFRA ME_L	KA20 G	VDD_REF	A
B	VSS	VSS	VSS	STRAPH3	STRAPH0	VSS	DAD DRS0	VSS	DDAT AS15	DDAT AS2	VSS	DDAT AS6	DCS1 P_L	VSS	DIOW P_L	DDAT AP1	VSS	DDAT AP5	DRST P_L	VSS	SPKR	LAD0	VSS	IRQ1	VSS	GPIO 26	B	
C	VDD_LDT	VDD_LDT	VDD_LDT	VSS	NC29	NC22	DAD DRS2	IRQ15	DDR QS	DDAT AS13	DDAT AS3	DDAT AS9	DCS3 P_L	DAD DRP0	DIOR P_L	DDAT AP14	DDAT AP2	DDAT AP10	DDAT AP7	OSC	INTIR Q8_L	KBRC_L	SERIR Q	GPIO 8	VSS	VSS	C	
D	LRXC AD_H [1]	LRXC AD_L [0]	LRXC AD_H [0]	VDD_LDT	STRAPH1	VDD_IO	DCS3 S_L	VDD_IO	DIOR S_L	DDAT AS14	VDD_IO	DDAT AS10	DDAT AS7	VDD_IO	DDAC KP_L	DDAT AP15	VDD_IO	DDAT AP11	DDAT AP6	VDD_IO	LDRQ_L[1]	LAD1	VDD_IO	VSS	USB1_H[0]	USB1_H[0]	D	
E	LRXC AD_L [1]	VSS	NC2	STRAPH2	NC20	NC9	DCS1 S_L	DDAC KS_L	DIOW S_L	DDAT AS1	DDAT AS11	DDAT AS5	DRST S_L	DAD DRP1	DRD YP	DDAT AP0	DDAT AP12	DDAT AP4	DDAT AP8	GPIO 29	PRDY	IRQ12	VSS	USB1_L[1]	USB1_H[1]	VSS	E	
F	LRXC AD_H [3]	LRXC AD_L [2]	LRXC AD_H [2]	VSS	NC7	VSS	VDD_CORE	VDD_CORE	VDD_CORE	VDD_CORE							VDD_IO	VDD_IO	VDD_IO	VDD_IO	VSS	TEST_L	GPIO 31	VSS	USB1_L[2]	USB1_H[2]	F	
G	LRXC AD_L [3]	VSS	LDTC OMP0	LDTC OMP1	NC8	VSS															VDD_CORE X	CLKR UN_L	USBO_C_L[0]	NC31	VSS	VDD_USB	G	
H	NC0	LRXC LK_L	LRXC LK_H	VSS	LRXC AD_H [4]	VDD_CORE															VDD_CORE X	NC13	USBO_C_L[1]	NC32	VSS	VDD_USB	H	
J	NC6	VSS	LRXC AD_H [5]	LRXC AD_L [5]	VDD_CORE																VDD_CORE X	VSS_USBA	USB_REXT	VSS	USBO_H[2]	USBO_L[2]	J	
K	LRXC TL_H	NC15	NC17	VSS	LRXC AD_H [6]	VDD_CORE															VDD_CORE X	VDD_USBA	VSS_USBA	USBO_H[1]	USBO_L[1]	VSS	K	
L	LRXC TL_L	VSS	LRXC AD_L [7]	LRXC AD_H [7]						VSS	VSS	VSS	VSS	VSS	VSS							VDD_USBA	VSS	VSS	USBO_H[0]	USBO_L[0]	L	
M	NC11	LTXC TL_H	LTXC TL_L	VSS	LTXC AD_L [7]					VSS	VSS	VSS	VSS	VSS	VSS							VDD_IOX	VDD_IOX	VSS	VSS	VSS	M	
N	NC14	VSS	LTXC AD_L [6]	LTXC AD_H [6]	LTXC AD_H [7]					VSS	VSS	VSS	VSS	VSS	VSS							SMB_USD0	RTXC_OUT	INTRUDER_L	RTXC_IN	S3PL_L_LF	N	
P	LTXC LK_L	NC16	NC18	VSS	LTXC AD_L [5]					VSS	VSS	VSS	VSS	VSS	VSS							SMB_USC0	GPIO 14	LDTR ST_L	S3PL_L_LF_VSS	PWR_OK	P	
R	LTXC LK_H	VSS	LTXC AD_L [4]	LTXC AD_H [4]	LTXC AD_H [5]					VSS	VSS	VSS	VSS	VSS	VSS							VDD_IOX	VDD_IOX	PWR_BTN_L	VSS	EXTS_ML_L	R	
T	LTXC AD_L [2]	LTXC AD_H [3]	LTXC AD_L [3]	VSS	LDTC OMP2					VSS	VSS	VSS	VSS	VSS	VSS							NC4	DCST_OP_L	RPW RON	SMB_ALER_TO_L	SLPB TN_L	T	
U	LTXC AD_H [2]	VSS	NC5	NC1	LDTC OMP3	VDD_CORE																VDD_CORE X	SMB_USD1	LID	PWR_ON_L	ACAV	SMB_ALER_TL_L	U
V	LTXC AD_L [0]	LTXC AD_H [1]	LTXC AD_L [1]	VSS	VSS	VDD_CORE																VDD_CORE X	VDD_IOX	VDD_IOX	SUSP_END_L	VSS	SMB_USC1	V
W	LTXC AD_H [0]	VSS	VSS	VDD_LDT	VSS	VDD_CORE																VDD_CORE X	MIL_C OL	MIL_RS	C32K HZ	RI_L	BATL OW_L	W
Y	VDD_LDT	VDD_LDT	VDD_LDT	PIRQ_C_L	PIRQ_A_L	VDD_CORE																VDD_CORE X	MIL_T_X_CLK	MIL_R_X_ER	MIL_P_HY_RST	PME_L	RESE_T_L	Y
AA	PIRQ_B_L	PIRQ_D_L	AD30	VDD_IO	AD31	VSS	VSS	VDD_CORE	VDD_CORE	VDD_CORE						VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_CORE X	MIL_R_X_DV	VDD_IOX	MIL_T_X_EN	VSS	MIL_R_X_CLK	AA		
AB	AD29	AD28	AD27	AD16	FRA ME_L	AD15	AD13	CBE_L[0]	AD4	AD2	REQ_L[0]	GNT_L[2]	REQ_L[5]	GNT_L[6]	LDTS TOP_L	ACCL K	STRAPL2	LDTR EQ_L	NC24	NC3	STRAPL0	VDD_IOX	MIL_R_XD1	MIL_T_XD1	MIL_R_XD0	MIL_T_XD0	AB	
AC	AD26	VSS	AD25	VDD_IO	IRDY_L	CBE_L[1]	VDD_IO	AD8	AD3	VDD_IO	GNT_L[0]	REQ_L[3]	VDD_IO	PNPI RQ2	FANC ON_0	VDD_IO	GPIO 30	STRAPL3	VDD_IO	NC23	VDD_IO	VSS	VDD_IOX	MIL_T_XD3	MIL_R_XD2	MIL_T_XD2	AC	
AD	AD24	CBE_L[3]	AD20	AD17	TRDY_L	PERR_L	AD11	AD12	AD7	AD0	GNT_L[1]	GNT_L[3]	GNT_L[5]	CLK	PGNT_L	FANC ON_1	THER MTRI P_L	GPIO 27	GPIO 16	STRAPL1	NC25	GPIO 4	ACSD I1	VDD_IOX	VSS	MIL_R_XD3	AD	
AE	AD23	VSS	AD19	VSS	DEVS EL_L	SERR_L	VSS	AD10	AD6	VSS	GNT_L[1]	REQ_L[4]	VSS	PCIST OP_L	PNPI RQ0	VSS	CPUS LEEP_L	AGPS TOP_L	VSS	PREQ_L	GPIO 17	VSS	NC26	VSS	VDD_IOX	MIL_MDC	AE	
AF	AD22	AD21	AD18	CBE_L[2]	STOP_L	PAR	AD14	AD9	AD5	AD1	REQ_L[2]	GNT_L[4]	REQ_L[6]	USBC LK	PNPI RQ1	FANR PM	CPUS TOP_L	THER M_L	ACSD O	ACSY NC	NC27	GPIO 28	ACRS_T_L	ACSD I0	MIL_MDIO	VSS	AF	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		

Figure 36. BGA Designations (Top Side View)

**Table 81. Alphabetical Listing of Signals and Corresponding BGA Designators**

Ball	Signal	Ball	Signal	Ball	Signal
U25	ACAV	F1	LRXCAD_H[3]	Y6	VDD_CORE
AB16	ACCLK	H5	LRXCAD_H[4]	F7	VDD_CORE
AF23	ACRST_L	J3	LRXCAD_H[5]	F8	VDD_CORE
AF24	ACSDI0	K5	LRXCAD_H[6]	G21	VDD_COREX
AD23	ACSDI1	L3	LRXCAD_H[7]	H21	VDD_COREX
AF19	ACSDO	H2	LRXCLK_L	J21	VDD_COREX
AF20	ACSYNC	H3	LRXCLK_H	K21	VDD_COREX
AD10	AD0	L1	LRXCTL_L	U21	VDD_COREX
AF10	AD1	K1	LRXCTL_H	V21	VDD_COREX
AB10	AD2	V1	LTXCAD_L[0]	W21	VDD_COREX
AC9	AD3	V3	LTXCAD_L[1]	Y21	VDD_COREX
AB9	AD4	T1	LTXCAD_L[2]	AA21	VDD_COREX
AF9	AD5	T3	LTXCAD_L[3]	D6	VDD_IO
AE9	AD6	R3	LTXCAD_L[4]	AC4	VDD_IO
AD9	AD7	P5	LTXCAD_L[5]	F17	VDD_IO
AC8	AD8	N3	LTXCAD_L[6]	F18	VDD_IO
AF8	AD9	M5	LTXCAD_L[7]	D20	VDD_IO
AE8	AD10	W1	LTXCAD_H[0]	D17	VDD_IO
AD7	AD11	V2	LTXCAD_H[1]	F19	VDD_IO
AD8	AD12	U1	LTXCAD_H[2]	F20	VDD_IO
AB7	AD13	T2	LTXCAD_H[3]	D23	VDD_IO
AF7	AD14	R4	LTXCAD_H[4]	AA17	VDD_IO
AB6	AD15	R5	LTXCAD_H[5]	AA18	VDD_IO
AB4	AD16	N4	LTXCAD_H[6]	AA19	VDD_IO
AD4	AD17	N5	LTXCAD_H[7]	AA20	VDD_IO
AF3	AD18	P1	LTXCLK_L	AC21	VDD_IO
AE3	AD19	R1	LTXCLK_H	AC16	VDD_IO
AD3	AD20	M3	LTXCTL_L	AC19	VDD_IO
AF2	AD21	M2	LTXCTL_H	AC7	VDD_IO
AF1	AD22	W22	MII_COL	D8	VDD_IO
AE1	AD23	W23	MII_CRS	AC10	VDD_IO
AD1	AD24	AE26	MII_MDC	D11	VDD_IO
AC3	AD25	AF25	MII_MDIO	AC13	VDD_IO
AC1	AD26	Y24	MII_PHY_RST	D14	VDD_IO
AB3	AD27	AA26	MII_RX_CLK	AA4	VDD_IO
AB2	AD28	AA22	MII_RX_DV	M22	VDD_IOX
AB1	AD29	Y23	MII_RX_ER	AD24	VDD_IOX
AA3	AD30	AB25	MII_RXD0	AE25	VDD_IOX
AA5	AD31	AB23	MII_RXD1	R22	VDD_IOX
AE18	AGPSTOP_L	AC25	MII_RXD2	V22	VDD_IOX
W26	BATLOW_L	AD26	MII_RXD3	AB22	VDD_IOX

**Table 81. Alphabetical Listing of Signals and Corresponding BGA Designators (Continued)**

Ball	Signal	Ball	Signal	Ball	Signal
W24	C32KHZ	Y22	MII_TX_CLK	M23	VDD_IOX
AB8	CBE_L[0]	AA24	MII_TX_EN	R23	VDD_IOX
AC6	CBE_L[1]	AB26	MII_TXD0	V23	VDD_IOX
AF4	CBE_L[2]	AB24	MII_TXD1	AA23	VDD_IOX
AD2	CBE_L[3]	AC26	MII_TXD2	AC23	VDD_IOX
G22	CLKRUN_L	AC24	MII_TXD3	Y1	VDD_LDT
AE17	CPUSLEEP_L	H1	NC0	Y2	VDD_LDT
AF17	CPUSTOP_L	U4	NC1	Y3	VDD_LDT
C14	DADDRP0	E3	NC2	W4	VDD_LDT
E14	DADDRP1	AB20	NC3	C1	VDD_LDT
A13	DADDRP2	T22	NC4	C2	VDD_LDT
B7	DADDRS0	U3	NC5	C3	VDD_LDT
A7	DADDRS1	J1	NC6	D4	VDD_LDT
C7	DADDRS2	F5	NC7	A26	VDD_REF
B13	DCS1P_L	G5	NC8	A23	VDD_RTC
E7	DCS1S_L	E6	NC9	G26	VDD_USB
C13	DCS3P_L	A1	NC10	H26	VDD_USB
D7	DCS3S_L	M1	NC11	K22	VDD_USBA
T23	DCSTOP_L	A2	NC12	L22	VDD_USBA
D15	DDACKP_L	H22	NC13	F24	VSS
E8	DDACKS_L	N1	NC14	D24	VSS
E16	DDATAP0	K2	NC15	E23	VSS
B16	DDATAP1	P2	NC16	L24	VSS
C17	DDATAP2	K3	NC17	M26	VSS
A17	DDATAP3	P3	NC18	M25	VSS
E18	DDATAP4	A3	NC19	G25	VSS
B18	DDATAP5	E5	NC20	E26	VSS
D19	DDATAP6	A5	NC21	J24	VSS
C19	DDATAP7	C6	NC22	C26	VSS
E19	DDATAP8	AC20	NC23	B6	VSS
A18	DDATAP9	AB19	NC24	L23	VSS
C18	DDATAP10	AD21	NC25	H25	VSS
D18	DDATAP11	AE23	NC26	M24	VSS
E17	DDATAP12	AF21	NC27	AC22	VSS
A16	DDATAP13	A6	NC28	U2	VSS
C16	DDATAP14	C5	NC29	W2	VSS
D16	DDATAP15	A4	NC30	AC2	VSS
A9	DDATAS0	G24	NC31	AE2	VSS
E10	DDATAS1	H24	NC32	B3	VSS
B10	DDATAS2	C20	OSC	W3	VSS
C11	DDATAS3	AF6	PAR	C4	VSS

**Table 81. Alphabetical Listing of Signals and Corresponding BGA Designators (Continued)**

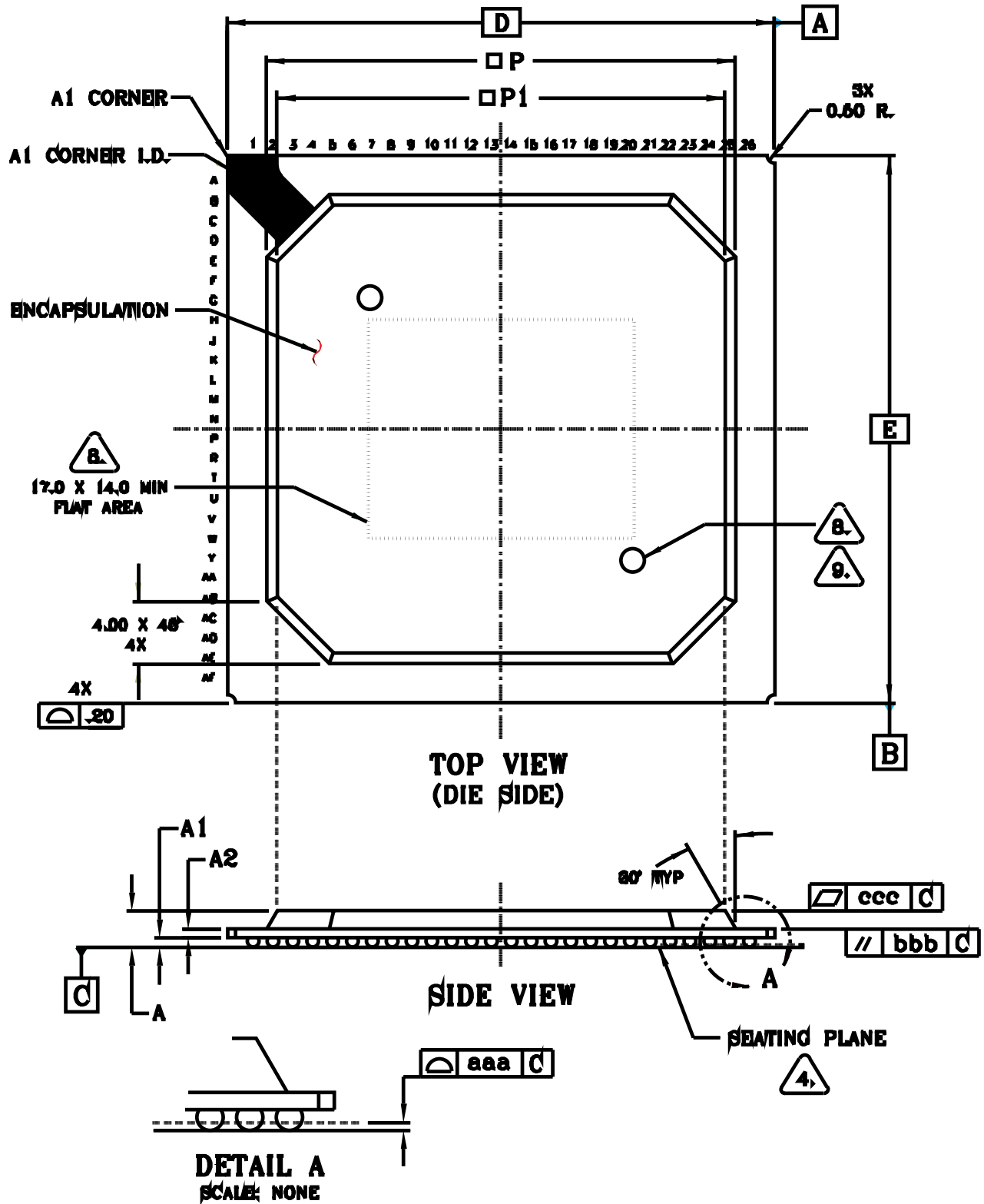
Ball	Signal	Ball	Signal	Ball	Signal
A11	DDATAS4	AE14	PCISTOP_L	F4	VSS
E12	DDATAS5	AD14	PCLK	H4	VSS
B12	DDATAS6	AD6	PERR_L	B1	VSS
D13	DDATAS7	AD15	PGNT_L	K4	VSS
A12	DDATAS8	Y5	PIRQA_L	M4	VSS
C12	DDATAS9	AA1	PIRQB_L	P4	VSS
D12	DDATAS10	Y4	PIRQC_L	T4	VSS
E11	DDATAS11	AA2	PIRQD_L	V4	VSS
A10	DDATAS12	Y25	PME_L	AE4	VSS
C10	DDATAS13	AE15	PNPIRQ0	V5	VSS
D10	DDATAS14	AF15	PNPIRQ1	W5	VSS
B9	DDATAS15	AC14	PNPIRQ2	B2	VSS
A15	DDRQP	E21	PRDY	F6	VSS
C9	DDRQS	AE20	PREQ_L	G6	VSS
AE5	DEVSEL_L	R24	PWRBTN_L	AA6	VSS
C15	DIORP_L	P26	PWROK	AE7	VSS
D9	DIORS_L	U24	PWRON_L	AA7	VSS
B15	DIOWP_L	AB11	REQ_L[0]	B8	VSS
E9	DIOWS_L	AD11	REQ_L[1]	AE10	VSS
E15	DRDYP	AF11	REQ_L[2]	L11	VSS
A8	DRDYS	AC12	REQ_L[3]	M11	VSS
B19	DRSTP_L	AE12	REQ_L[4]	N11	VSS
E13	DRSTS_L	AB13	REQ_L[5]	E2	VSS
R26	EXTSMI_L	AF13	REQ_L[6]	P11	VSS
AC15	FANCON_0	Y26	RESET_L	R11	VSS
AD16	FANCON_1	W25	RI_L	T11	VSS
AF16	FANRPM	T24	RPWRON	B11	VSS
AB5	FRAME_L	N25	RTCX_IN	L12	VSS
AC11	GNT_L[0]	N23	RTCX_OUT	M12	VSS
AE11	GNT_L[1]	N26	S3PLL_LF	N12	VSS
AB12	GNT_L[2]	P25	S3PLL_LF_VSS	P12	VSS
AD12	GNT_L[3]	C23	SERIRQ	R12	VSS
AF12	GNT_L[4]	AE6	SERR_L	T12	VSS
AD13	GNT_L[5]	T26	SLPBTN_L	G2	VSS
AB14	GNT_L[6]	T25	SMBALERT0_L	AE13	VSS
P23	GPIO14	U26	SMBALERT1_L	L13	VSS
AD19	GPIO16	P22	SMBUSC0	M13	VSS
AE21	GPIO17	V26	SMBUSC1	N13	VSS
B26	GPIO26	N22	SMBUSD0	P13	VSS
AD18	GPIO27	U22	SMBUSD1	R13	VSS
AF22	GPIO28	B21	SPKR	T13	VSS

**Table 81. Alphabetical Listing of Signals and Corresponding BGA Designators (Continued)**

Ball	Signal	Ball	Signal	Ball	Signal
E20	GPIO29	AF5	STOP_L	L14	VSS
AC17	GPIO30	B5	STRAPH0	M14	VSS
F23	GPIO31	D5	STRAPH1	N14	VSS
AD22	GPIO4	E4	STRAPH2	J2	VSS
C24	GPIO8	B4	STRAPH3	P14	VSS
C21	INTIRQ8_L	AB21	STRAPL0	R14	VSS
N24	INTRUDER_L	AD20	STRAPL1	T14	VSS
AC5	IRDY_L	AB17	STRAPL2	B14	VSS
B24	IRQ1	AC18	STRAPL3	L15	VSS
E22	IRQ12	V24	SUSPEND_L	M15	VSS
A14	IRQ14	F22	TEST_L	N15	VSS
C8	IRQ15	AF18	THERM_L	P15	VSS
A22	IRQ6	AD17	THERMTRIP_L	R15	VSS
A25	KA20G	AD5	TRDY_L	T15	VSS
C22	KBRC_L	J23	USB_REXT	L2	VSS
B22	LAD0	L26	USB0_L[0]	AE16	VSS
D22	LAD1	K25	USB0_L[1]	L16	VSS
A20	LAD2	J26	USB0_L[2]	M16	VSS
A21	LAD3	L25	USB0_H[0]	N16	VSS
A19	LDRQ_L[0]	K24	USB0_H[1]	P16	VSS
D21	LDRQ_L[1]	J25	USB0_H[2]	R16	VSS
G3	LDTCOMP0	D25	USB1_L[0]	T16	VSS
G4	LDTCOMP1	E24	USB1_L[1]	B17	VSS
T5	LDTCOMP2	F25	USB1_L[2]	B20	VSS
U5	LDTCOMP3	D26	USB1_H[0]	AE19	VSS
AB18	LDTREQ_L	E25	USB1_H[1]	B23	VSS
P24	LDTRST_L	F26	USB1_H[2]	N2	VSS
AB15	LDTSTOP_L	AF14	USBCLK	F21	VSS
A24	LFRAME_L	G23	USBOC_L[0]	AE22	VSS
U23	LID	H23	USBOC_L[1]	B25	VSS
D2	LRXCAD_L[0]	H6	VDD_CORE	AE24	VSS
E1	LRXCAD_L[1]	F9	VDD_CORE	C25	VSS
F2	LRXCAD_L[2]	F10	VDD_CORE	K26	VSS
G1	LRXCAD_L[3]	AA8	VDD_CORE	R25	VSS
J5	LRXCAD_L[4]	AA9	VDD_CORE	V25	VSS
J4	LRXCAD_L[5]	AA10	VDD_CORE	AA25	VSS
L5	LRXCAD_L[6]	J6	VDD_CORE	AD25	VSS
L4	LRXCAD_L[7]	K6	VDD_CORE	R2	VSS
D3	LRXCAD_H[0]	U6	VDD_CORE	AF26	VSS
D1	LRXCAD_H[1]	V6	VDD_CORE	J22	VSS_USBA
F3	LRXCAD_H[2]	W6	VDD_CORE	K23	VSS_USBA









Symbol	Description
1	Dimensions and tolerances conform to ASME Y14.5M–1994.
2	All dimensions are in millimeters.
	Dimension 'b' is measured at the maximum solder ball diameter on a plane parallel to Datum C.
	Datum C and the seating plane are defined by the spherical crowns of the solder balls.
	The number of peripheral rows and columns.
	'S' is measured with respect to Datums A and B, and defines the position of the solder balls nearest the package centerlines.
7	Conforms to JEP-95, MO-151, Issue 9, Variation Bal-2.
	The minimum flat area top side of the package is used for marking and pickup. The flatness specification applied to this area. Any ejector marks must be outside of this area.
	Optional features.

Symbol	Minimum	Nominal	Maximum	Description
A	2.20	2.33	2.46	Overall thickness
A1	0.50	0.60	0.70	Ball height
A2	0.51	0.56	0.61	Body thickness
D		35.00 BSC.		Body size
D1		31.75 BSC.		Ball footprint
E		35.00 BSC.		Body size
E1		31.75 BSC.		Ball footprint
M		26 x 26		Ball matrix size
N		492		Total ball count
MR		5		Number of rows
b	0.60	0.75	0.90	Ball diameter
e		1.27 BSC.		Ball pitch
P	29.9	30.0	30.1	Encapsulation area
P1		28.0 Minimum		Flat encapsulation area
S		0.635 BSC.		Solder ball placement

Symbol	Tolerance	Description
aaa	0.15	Coplanarity
bbb	0.15	Parallelism
ccc	0.15	Flatness



## Chapter 8 Test

### 8.1 Pin Select Options

The device includes a number of test modes which are entered through pin selection. These modes are entered by asserting the following pins:

**Table 82. Pin Select Options**

Mode	Equation to enable mode
High Impedance	$\sim\text{TEST\_L} \& \text{PWROK} \& \sim\text{PREQ\_L} \& \sim\text{SERR\_L}$
NAND Tree	$\sim\text{TEST\_L} \& \text{PWROK} \& \sim\text{PREQ\_L} \& \text{SERR\_L}$

#### 8.1.1 NAND Tree Mode

The device includes a number of NAND trees around the periphery that can be used for continuity testing.

The NAND trees are connected in the following order:

**Table 84. NAND Tree 1: Output GNT\_L[0]**

1	LDTCOMP2	11	LRXCAD_H/L[5]	21	LTXCAD_H[2]	31	LTXCAD_H[7]
2	LDTCOMP3	12	LRXCAD_H/L[6]	22	LTXCAD_L[2]	32	LTXCAD_L[7]
3	LDTCOMP0	13	LRXCAD_H/L[7]	23	LTXCAD_H[3]	33	LTXCTL_H
4	LDTCOMP1	14	LRXCTL_H/L	24	LTXCAD_L[3]	34	LTXCTL_L
5	LRXCLK_H/L	15	LTXCLK_H	25	LTXCAD_H[4]	35	
6	LRXCAD_H/L[0]	16	LTXCLK_L	26	LTXCAD_L[4]	36	
7	LRXCAD_H/L[1]	17	LTXCAD_H[0]	27	LTXCAD_H[5]	37	
8	LRXCAD_H/L[2]	18	LTXCAD_L[0]	28	LTXCAD_L[5]	38	
9	LRXCAD_H/L[3]	19	LTXCAD_H[1]	29	LTXCAD_H[6]	39	
10	LRXCAD_H/L[4]	20	LTXCAD_L[1]	30	LTXCAD_L[6]	40	

*Note:* All LRXCAD inputs to the NAND tree are differential.

**Table 85. NAND Tree 2: Output GNT\_L[1]**

1	PIRQB_L	11	CBE_L[3]	21	AD17	31	CBE_L[0]	41	AD9
2	AD29	12	AD23	22	IRDY_L	32	AD11	42	AD7
3	PIRQD_L	13	AD31	23	AD15	33	STOP_L	43	AD6
4	AD28	14	AD16	24	TRDY_L	34	AD8	44	AD0
5	AD30	15	AD25	25	AD18	35	AD4	45	AD2
6	PIRQC_L	16	AD20	26	CBE_L[1]	36	PAR	46	AD5
7	AD26	17	AD22	27	AD13	37	AD14	47	REQ_L[1]
8	AD24	18	AD21	28	PERR_L	38	AD3	48	AD1
9	AD27	19	AD19	29	DEVSEL_L	39	AD12	49	
10	PIRQA_L	20	FRAME_L	30	CBE_L[2]	40	AD10	50	

**Table 86. NAND Tree 3: Output NC24**

1	REQ_L[2]	10	USBCLK	19	THERM_L	28	ACCLK	37	NC25
2	REQ_L[0]	11	PNPIRQ1	20	CPUSLEEP_L	29	NC27	38	GPIO4
3	GNT_L[2]	12	REQ_L[5]	21	ACSDO	30	GPIO28	39	LDTREQ_L
4	REQ_L[4]	13	PNPIRQ0	22	FANCON_0	31	GPIO30	40	NC
5	GNT_L[4]	14	PGNT_L	23	LDTSTOP_L	32	GPIO17	41	
6	REQ_L[6]	15	FANRPM	24	FANCON_1	33	GPIO16	42	
7	GNT_L[5]	16	PNPIRQ2	25	ACSYNC	34	STRAPL3	43	
8	REQ_L[3]	17	GNT_L[6]	26	THERMTRIP_L	35	STRAPL2	44	
9	PCISTOP_L	18	CPUSTOP_L	27	GPIO27	36	STRAPL1	45	

**Table 87. NAND Tree 4: Output DCSTOP\_L**

1	USB1_H[2]	13	INTRUDER_L	25	MII_TXD1	37	MII_PHY_RST	49	SMBALERT1_L
2	USB1_L[2]	14	AGPSTOP_L	26	MII_RX_DV	38	C32KHZ	50	RPWRON
3	USB1_H[1]	15	ACRST_L	27	MII_RXD2	39	PME_L	51	SMBUSD0
4	USB1_L[1]	16	NC26	28	MII_TXD2	40	RESET_L	52	SMBALERT0_L
5	USB1_H[0]	17	ACSDI1	29	MII_TX_CLK	41	SUSPEND_L	53	PWRBTN_L
6	USB1_L[0]	18	ACSDI0	30	MII_RX_ER	42	RI_L	54	SLPBTN_L
7	USB0_H[2]	19	MII_MDIO	31	MII_TXD0	43	BATLOW_L	55	EXTSMI_L
8	USB0_L[2]	20	STRAPL0	32	MII_TX_EN	44	LID	56	GPIO14
9	USB0_H[1]	21	MII_MDC	33	MII_RXD0	45	SMBUSD1	57	SMBUSC0
10	USB0_L[1]	22	MII_RXD3	34	MII_COL	46	PWRON_L	58	LDTRST_L
11	USB0_H[0]	23	MII_TXD3	35	MII_CRS	47	SMBUSC1	59	
12	USB0_L[0]	24	MII_RXD1	36	MII_RX_CLK	48	ACAV	60	

**Table 88. NAND Tree 5: Output DIOWP\_L**

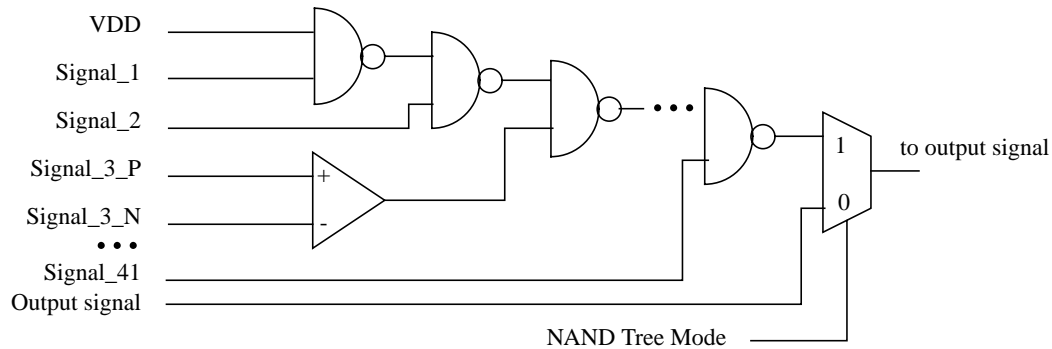
1	USBOC_L[1]	11	IRQ12	21	DDATAP8	31	DDATAP10	41	DDATAP3
2	USBOC_L[0]	12	LAD1	22	DDATAP4	32	DRSTP_L	42	DADDRP1
3	GPIO26	13	KBRC_L	23	LAD3	33	DDATAP0	43	DIORP_L
4	CLKRUN_L	14	LAD0	24	DDATAP6	34	DDATAP15	44	DDATAP1
5	KA20G	15	IRQ6	25	OSC	35	DDATAP9	45	DDATAP13
6	GPIO31	16	PRDY	26	DDATAP7	36	DDATAP2	46	DRSTS_L
7	GPIO8	17	LDRQ_L[1]	27	DDATAP11	37	DDATAP5	47	DADDRP0
8	IRQ1	18	INTIRQ8_L	28	LAD2	38	DRDYP	48	
9	LFRAME_L	19	GPIO29	29	DDATAP12	39	DDACKP_L	49	
10	SERIRQ	20	SPKR	30	LDRQ_L[0]	40	DDATAP14	50	

**Table 89. NAND Tree 6: Output GNT\_L[3]**

1	DDRQP	11	DDATAS6	21	DDATAS15	31	DCS3S_L	41	NC19
2	DDATAS7	12	DDATAS8	22	DRDYS	32	DCS1S_L	42	NC20
3	IRQ14	13	DDATAS3	23	DIORS_L	33	NC28	43	STRAPH2
4	DCS3P_L	14	DDATAS4	24	DDRQS	34	NC22	44	
5	DCS1P_L	15	DDATAS12	25	DIEWS_L	35	NC21	45	
6	DDATAS10	16	DDATAS2	26	IRQ15	36	STRAPH0	46	
7	DDATAS5	17	DDATAS14	27	DADDRS1	37	NC29	47	
8	DDATAS9	18	DDATAS1	28	DADDRS0	38	NC30	48	
9	DDATAS11	19	DDATAS13	29	DADDRS2	39	STRAPH3	49	
10	DADDRP2	20	DDATAS0	30	DDACKS_L	40	STRAH1	50	

The following pins are not part of the NAND tree: RTCX\_IN, RTCX\_OUT, S3PLL\_LF, S3PLL\_LF\_VSS, USB\_REXT, PWROK, PCLK, TEST\_L, PREQ\_L, SERR\_L.

Each of the above columns show the NAND tree, with the last signal used as the output. For example, the first column is connected as follows:



### **8.1.2 High Impedance Mode**

If this mode is entered all bidirectional and open-drain pins are switched into high-z state.

## Appendix A Glossary

---

**Application processor.** In a multiprocessor system, an application processor is any CPU except the boot processor.

**Boot processor.** In a multiprocessor system, the boot processor is the processor that initializes the system.

**FON.** System full on state which occurs when the MAIN and AUX planes are powered.

**Function.** A function is like a separate PCI device within a PCI peripheral. See “3.7.4.1 Configuration Mechanism #1” of the revision 2.1 PCI specification for details on how functions are mapped in configuration address space.

**IOAPIC.** I/O advanced programmable interrupt controller. This term refers to the industry-standard register definitions.

**MOFF.** System mechanical off state which occurs when the AUX planes are not powered.

**MP.** Multiprocessor.

**Offset.** This generally refers to the value that must be added to a base address pointer to address a specific register.

**PIC.** Programmable interrupt controller. This term refers to the dual-8259-based system interrupt control as found in the original AT ISA bus.

**PIT.** Programmable interval timer. This is a legacy ISA-bus device that is incorporated into the logic. It is used to generate interrupts from timers.

**POS.** System power on suspend state. The AUX and MAIN planes are powered in this state; the processor and potentially other system components are not functional when in this state.

**Power button override event.** This event occurs when PWRBTN\_L is held active for at least four seconds. The IC transitions to the SOFF state, if this event occurs.

**SOFF.** System soft off state which occurs when the AUX planes are powered but the MAIN plane is not powered.

**STR.** System suspend to RAM state. In this state, only the AUX planes are powered in the IC; most of the rest of the system is powered down except resume logic and the system DRAM.

**USB.** Universal serial bus.

---

## Appendix B References

---

*Advanced Configuration and Power Interface Specification Revision 1.0.* By Intel, Microsoft, and Toshiba. Copyright 1996, 1997.

*Advanced Configuration and Power Interface Specification Revision 2.0.* By Compaq, Intel, Microsoft, Phoenix, and Toshiba. Copyright 2000.

*Hydra ASIC Design Specification.* Revision 1.2, dated 11-Sep-96, by Compaq.

*HyperTransport™ I/O Link Protocol Specification, Revision 1.02.* From the HyperTransport Technology Consortium, [www.hypertransport.org](http://www.hypertransport.org).

*Low Pin Count (LPC) Interface Specification, revision 1.0.* From Intel Corporation.

*Multiprocessor Specification, Version 1.4, August 1996.* From Intel Corporation.

*OpenHCI for USB.* Release 1.0a, dated 9-Mar-97, by Compaq, Microsoft, and National Semiconductor.

*Mobile PC/PCI DMA Arbitration and Protocols.* Revision 2.2, dated 22-Apr-96, by Intel Corporation.

*PCI IDE Controller Specification, revision 1.0.* Dated 4-Mar-94. By the PCI Special Interest Group, PO Box 14070, Portland, OR, 97214, (800) 433-5177.

*PCI Mobile Design Guide, Version 1.1.* Dated 18-Dec-98. By the PCI Special Interest Group, PO Box 14070, Portland, OR, 97214, (800) 433-5177.

*PCI System Architecture, Third Edition.* By Tom Shanley and Don Anderson. Copyright 1995, by MindShare, Inc., ISBN: 0-201-40993-3.

*Revision 2.1 PCI Specification.* By the PCI Special Interest Group, PO Box 14070, Portland, OR, 97214, (800) 433-5177. Copyright 1992, 1993, and 1995 by the PCI Special Interest Group.

*Serial IRQ Specification Version 1.0.* VESA (Video Electronics Standards Association), 2150 North First Street, Suite 440, San Jose, CA, 95131-2029, Phone: (408) 435-0333.

*System Management Bus Specification Revision 1.0.* Dated 15-Feb-95. Copyright 1996. Written by Benchmarq Microelectronics Inc., Duracell Inc., Energizer Power Systems, Intel Corporation, Linear Technology Corporation, Maxim Integrated Products, Mitsubishi Electric Corporation, National Semiconductor Corporation, Toshiba Battery Co., Varta Batterie AG.

*Universal Serial Bus Specification Revision 1.0.* By Compaq, DEC, IBM, Intel, Microsoft, NEC, and Northern Telecom. Copyright 1995, 1996.



## Appendix C Conventions

---

In this specification, all formulae and number schemes follow Verilog numerical conventions. Here is a brief summary:

- y'hx     'h means that the number that follows it, x, is in hexadecimal format. If there is a number before the 'h, y, specifies the number of bits in x.
- { }       These brackets are used to indicate a group of bits that are concatenated together.
- |         This is the logical OR operator.
- &        This is the logical AND operator.
- ~        This is the logical NOT operator.
- ==      This is the logical "is equal to" operator.
- !=      This is the logical "is not equal to" operator.
- \*        Multiply.

The order in which the operators are applied is: ~ first, & second, and | last.

