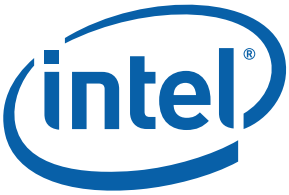


Intel® 6300ESB I/O Controller Hub

Datasheet

November 2007

Notice: The Intel® 6300ESB I/O Controller Hub may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Order Number: 300641-004US

Intel® 6300ESB I/O Controller Hub Product Features

- 8-Bit Hub Interface
  - — 266 Mbyte/s maximum throughput
  - — Parallel Termination scheme for longer trace lengths
  - — Supports Lower Voltages as per Hub Interface 1.5 spec
- PCI-X Bus I/F
  - — Supports PCI-X Rev 1.0 Specification at 66 MHz
  - — Supports PCI Rev 2.2 Specification at 33 MHz
  - — Support external master devices on PCI
  - — 4 @ PCI 33 MHz
  - — 2 @ PCI 64/66 MHz
  - — 4 @ PCI-X 64/66 MHz (two slots and two soldered down devices)
  - — Support for 64-bit addressing on PCI-X using DAC protocol
- PCI Bus I/F
  - — Supports PCI 32b/33 MHz
  - — 120 Mbyte/s throughput
  - — Supports PCI Rev 2.2 Specification at 33 MHz
  - — Supports 4 external master devices @ 33 MHz
  - — Support for 44-bit addressing on PCI using DAC protocol.
  - — 4 slots supported
- Integrated IDE Controller
  - — Supports “Native Mode” Register and Interrupts
  - — Supports faster PIO timings for non-data cycles
  - — Independent timing of up to four drives, with separate Primary and Secondary IDE cable connections
  - — Supports Ultra 100 DMA Mode Transfers up to 100 Mbytes/s for reads from disk; 88.88 Mbytes/s for writes to disk, as well as Ultra66 and Ultra33 DMA modes.
  - — PIO Mode four transfers up to 14 Mbytes/s
- Integrated Serial ATA Host Controllers
  - — Independent DMA operation on two ports
  - — Data transfer rates up to 150 Mbyte/s
  - — Alternate Device ID and RAID Class Code option for support of Soft RAID
  - —
- Power Management Logic
  - — ACPI 1.0 compliant
  - — ACPI-defined power states S1 (Stop Grant), S3 (STR), S4 (STD), S5 (SOFF)
  - — ACPI Power Management Timer
  - — SMI# Generation
  - — PCI PME#
  - — Supports THRMTRIP# input, SYS_RESER# input and SLP_S4# output
  - — Support for APM-based legacy power management for non-ACPI implementations
- External Glue Integration
  - — Integrated Pull-up, Pull-down and Series Termination resistors on IDE, CPU I/F
  - — Integrated Pull-down and Series resistors on USB
- Enhanced Hub I/F buffers improve routing flexibility (Not available with all Memory Controller Hubs)
- Firmware Hub (FWH) I/F supports BIOS Memory size up to 8 Mbytes
- Low Pin Count (LPC) I/F
  - — New: No ISA/X-Bus support
  - — Allows connections of devices such as Super I/O, microcontrollers, customers ASICs
  - — Supports two Master/DMA devices
  - — Memory size up to 8 Mbytes
- Enhanced DMA Controller
  - — Two cascaded 8237 DMA controllers
  - — Supports LPC DMA
  - — Supports DMA Collection Buffer to provide Type-F DMA performance for all DMA channels
- Real-Time Clock
  - — 256-byte battery-backed CMOS RAM
- System TCO Reduction Circuits
  - — Timers to generate SMI# and Reset upon detection of system hang
  - — Interrupt capability to OS-specific manageability extension and OS capability to call TCO BIOS Timers to detect improper CPU reset
  - — Alert On Lan (AOL) to enable heartbeats and system event reporting via LAN controller
  - — Supports CPU BIST
  - — Supports ability to disable external devices

- USB
  - — Includes one EHCI USB2 host controllers, a total of four ports (shared with the UHCI ports)
  - — Two UHCI Host Controllers for a total of four ports (shared with EHCI ports)
  - — New: supports a USB 2.0 High-speed Debug Port
  - — Supports wake-up from sleeping states S1-S4
  - — Supports legacy Keyboard/Mouse software with USB-based keyboard and mouse
- AC'97 Link for Audio and Telephony CODECs
  - — New: Third AC_SDATA_IN Line for three codec support
  - — AC'97 2.2 compliant
  - — New: Independent bus master logic for 8 channels (PCM In/Out, Mic 1 Input, Mic 2 Input, Modem In/Out, S/PDIF Out)
  - — Separate independent PCI functions for Audio and Modem
  - — Support for up four to six channels of PCM audio output (full AC3 decode)
  - — Support for 20-bit sample
  - — Support for ACPI device states - D0 and D3
- Interrupt Controller
  - — Supports up to 12 PCI interrupt pins; four are not shared
  - — Two cascaded 82C59 with 15 interrupts
  - — Supports PCI scheme for delivering interrupts as write cycles (MSI)
  - — Integrated I/O APIC capability with 24 interrupts
  - — Supports Serial Interrupt Protocol
  - — Supports Front-Side Message Interrupt Delivery
- New: Multimedia Timers based on 82C54
  - — Includes three timer comparators
  - — System timer, Refresh request, Speaker tone output
  - — One-shot and periodic interrupts supported
- New: Watchdog Timer
  - — Two-Stage Watchdog with independent count values for each stage
  - — First stage generates an INT or SMI
  - — Second stage drives external pin active until cleared by a system reset or power cycle
  - — Configuration option for write-once enabling (count values can still change)
  - — Configurable granularity from 1µs to 10 min
- SMBus
  - — Flexible SMBus/SMLink architecture to optimize for ASF and eliminate board requirements of SMBus 2.0 compliance
  - — Supports SMBus 2.0 Specification
  - — Host interface allows CPU to communicate via SMBus
  - — Slave interface allows an external Microcontroller to access system resources
  - — Compatible with most 2-wire components that are also I^2C compatible
- New: Integrated 16550 compatible UARTs
  - — Enable/disable per UARTs
  - — Serial interrupts
  - — Can disable when external SIO used
- New: Port 60/64 Emulation
  - — Programmable interrupt generation on writes
  - — Positive decode to Port 60/64 emulation registers
- GPIO
  - — Four GPOs capable of directly driving LEDs
  - — Two GPOs maintain state during and after reset
- 1.5 V operation with 3.3 V I/O. 5 V tolerance on many buffers, including IDE.
- Package 37.5 x 37.5 mm 689 BGA
- Process P859.6

System Block Diagrams

Figure 1. Workstation / PC Model

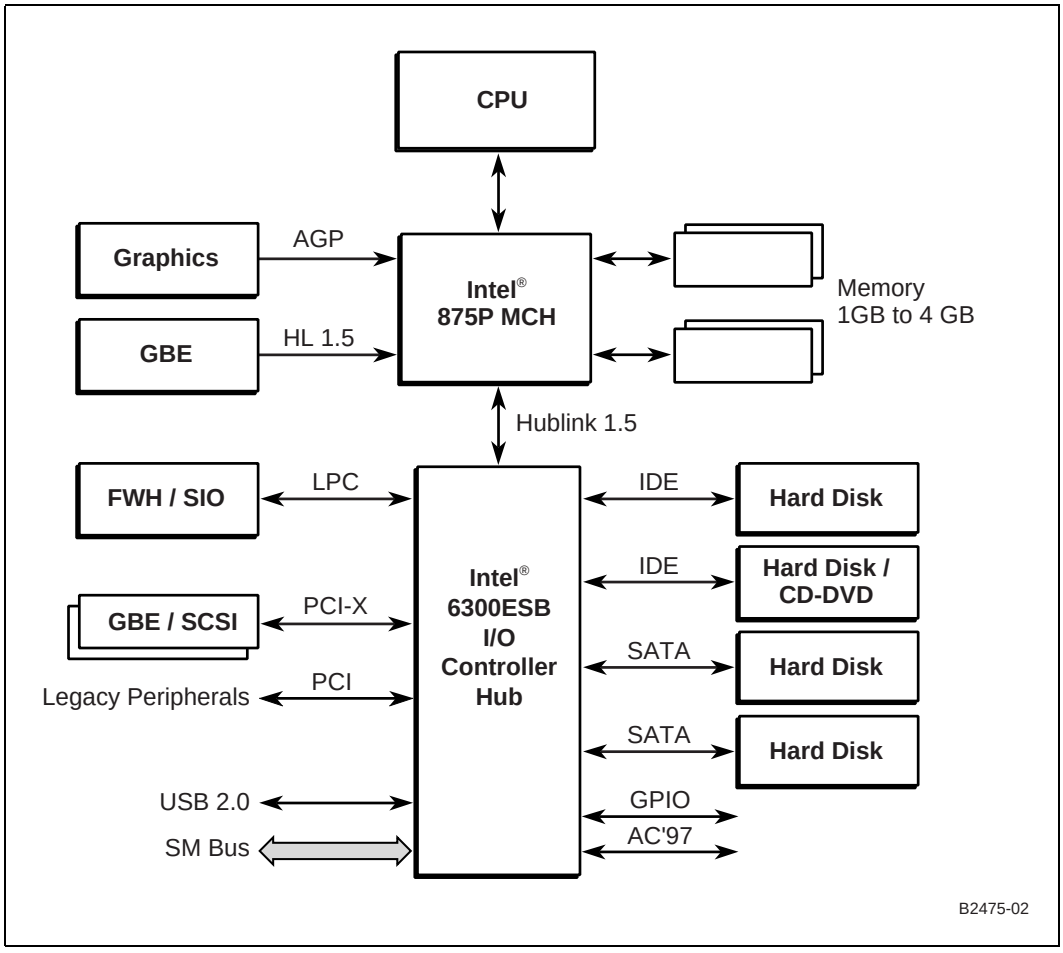


Figure 2. Low to Mid-Range Communication Appliance Model (Diskless)

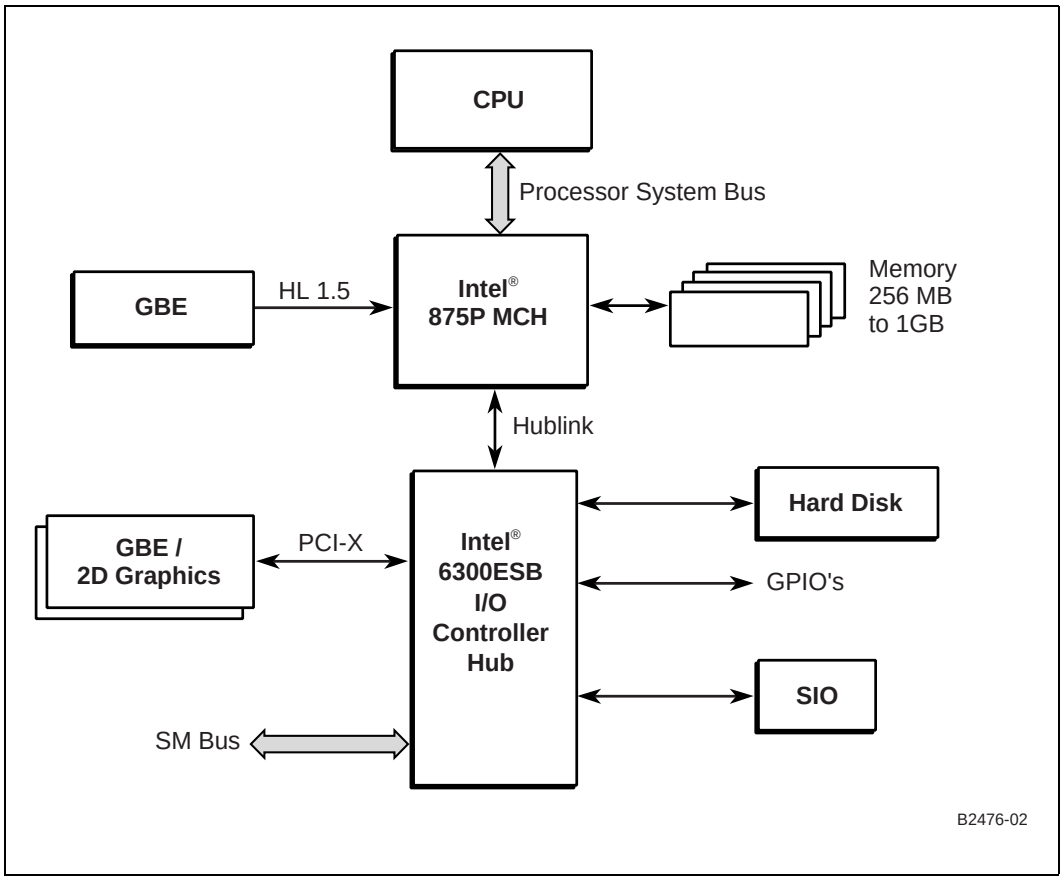
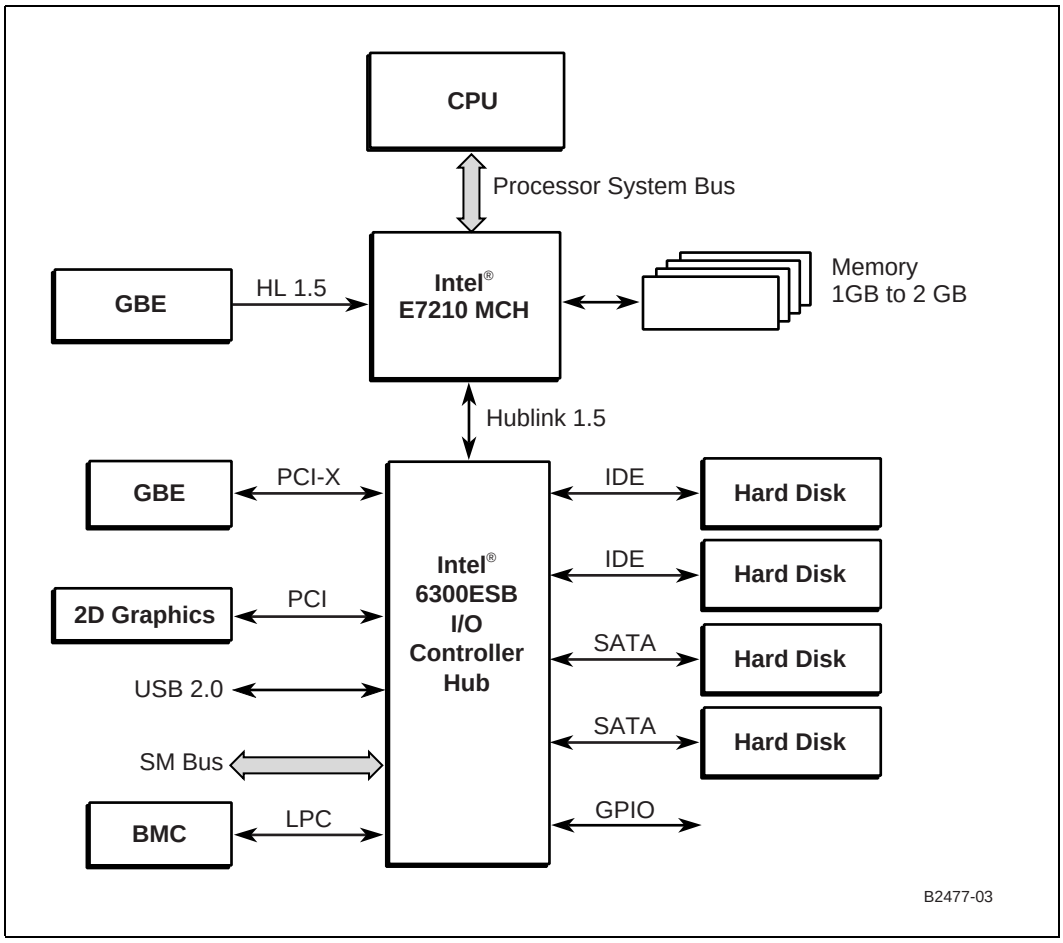
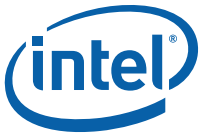
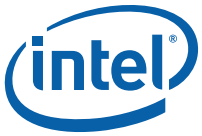


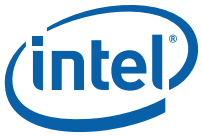
Figure 3. Value Server, Ultra-Dense Server and Low-End Server Blade

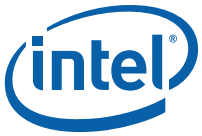


Contents



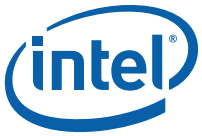


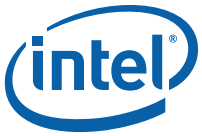


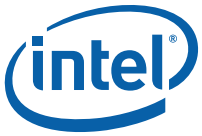


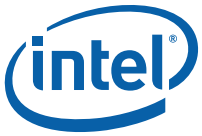
Figures

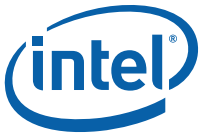
Tables

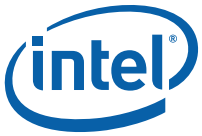












Revision History

Date	Revision	Description
November 2007	004	• Included Specification Changes from specification update, version 011 — Figure 61: Updated to replace V_CPU_IO with VccHI • Included Specifications Clarifications from specification update, version 011 — Table 727 and Table 728: Updated t175, t176 & t184 timing definitions to clarify which Vcc supplies apply to each — Section 5.11.1 and 5.11.6: Clarified wording regarding support of C2 state for dual processors, dual core and processors with HyperThreading Technology. — Table 191: Remove incorrect references to TCO in Note — Section 5.7.1: Removed references to three wire APIC bus. 6300ESB does not support this feature. — Table 31: Moved Note 1 refernces from I/O to Memory cycles — Table 727: Changed Note 2 to require that 3.3V and 1.5V rails must power up or down together — Removed all references to Processor Speed Strapping. Processors used in conjunction with 6300ESB do not use this feature. — Table 319: Modified register defintions to reflect that signals are configured as native functions after a full reset. — Section 19.1: Add Note to indicate that SIUs are not completely 16550 compatible. — Table 635: Added % error rates. — Section 5.10.2.2: Changed to clarify where multiple processor or multiple core configurations can generate Stop Grant cycles in the MCH supports it. • Included Documentation Changes 2 to 13 from specification update, version 011 — Table 568: Revised to allow bit column to align correctly. — Section 22.2: Added Case temperature under Bias value — Changed all references to PCIRST# to PXPCIRST# — Table 317: Changed register deult value to 00000000h — Figure 5: Correct typo in diagram — Table 28: Change GPIO[21] After Reset value to logic ‘1’ — Corrected Product Features section to show that PCI-X Rev 1.0 is supported — Table 22: Correct V5REF definition — Table 573: Correct indexes for Reserved Registers — Table 581: Revised naming and definitions for bits 10:8 and 7:0. Added Note at bottom of table. — Table 728: Revised ‘SLP_S5# inactive to SLP_S4#’ parameter timing values • Table 313: Removed GPO_TTL register listing. This register was not relevant to 6300ESB • Table 29, Figure 61 and Figure 62: Removed references to LAN_RST and RSM_PWROK signals which do not exist in 6300ESB • Table 731, Figure 66: Updated to correct timing requirements for entering test mode. • Section 5.7 and Section 8.5: Removed PCI register references for APIC0. APIC0 registers are purely memory mapped.

Date	Revision	Description
December 2004	003	• Included changes from previous spec updates • Updated Section 22.1 • Updated Section 22.2 • Updated DC Characteristics Section 22.3 • -Updated AC Characteristics Section 22.4
June 2004	002	• Clarified WDT Reload register bit details; listed USB HS reference voltage register bits.
February 2004	001	• Initial release of this document.

Change Bars

A change bar to left of text, a table row or a figure heading indicates this item is either new or modified from the previous version of the document.

Introduction 1

1.1 About This Document

This datasheet is intended for Original Equipment Manufacturers (OEMs) and BIOS vendors creating products based on the Intel® 6300ESB I/O Controller Hub (ICH). This manual assumes a working knowledge of the vocabulary and principles of USB, IDE, AC'97, SMBus, PCI, ACPI, and LPC. Although some details of these features are described herein, refer to the individual industry specifications listed in Table 1 for the complete details.

Table 1. Industry Specifications

Specification	Location
Low Pin Count Interface Specification, Revision 1.0 (LPC)	http://developer.intel.com/design/chipsets/industry/lpc.htm
Audio Codec '97 Component Specification, Version 2.2 (AC'97)	http://www.intel.com/labs/media/audio/index.htm
PCI-X Specification, Revision 1.0	http://www.pcisig.com/specifications/pcix_20/pci_x/
System Management Bus Specification, Version 2.0 (SMBus)	http://www.smbus.org/specs/
PCI Local Bus Specification, Revision 2.2 (PCI)	http://pcisig.com/specs.htm
Universal Serial Bus Revision 2.0/1.0 Specification (USB)	http://www.usb.org
Advanced Configuration and Power Interface, Version 1.0b (ACPI)	http://www.teleport.com/~acpi/
Enhanced Host Controller Interface Specification for Universal Serial Bus, Revision 0.96 (EHCI)	http://developer.intel.com/technology/usb/ehcispec.htm
SATA 1.0 Specification	http://www.serialata.org/collateral/index.shtml
Other Industry Specifications	http://www.intel.com/design/motherbd/bv/bv_industryspecs.htm

This document contains these chapters:

Chapter 1, "Introduction" introduces the Intel® 6300ESB ICH and provides information on manual organization.

Chapter 3, "Signal Description" provides a detailed description of each Intel® 6300ESB ICH signal. Signals are arranged according to interface. Details are provided about the drive characteristics (Input/Output, Open Drain, etc.) of all signals.

Chapter 2, "Intel® 6300ESB ICH and System Clock Domains" provides a list of each clock domain associated with the Intel® 6300ESB ICH in an Intel® 6300ESB ICH-based system.

Chapter 4, "Intel® 6300ESB ICH Power Planes and Pin States" provides a complete list of signals, their associated power well, their logic level in each suspend state, and their logic level before and after reset.

Chapter 5, "Functional Description" provides a detailed description of the functions in the Intel® 6300ESB ICH. All PCI buses, devices, and functions in this manual are abbreviated using the following nomenclature; Bus:Device:Function. This manual abbreviates buses as B0 and B1, devices as D8, D29, D30 and D31 and functions as F0, F1, F2, F3, F4, F5, F6 and F7. For example Device 31 Function 5 is abbreviated as D31:F5, Bus 1 Device 8 Function 0 is abbreviated as B1:D8:F0. Generally, the bus number will not be used, and may be considered to be Bus 0. Note that the Intel® 6300ESB ICH's external PCI bus is typically Bus 1, but may be assigned a different number depending upon system configuration.

Chapter 6, "Register and Memory Mapping" provides an overview of the registers, fixed I/O ranges, variable I/O ranges, and memory ranges decoded by the Intel® 6300ESB ICH.

Chapter 7, "Hub Interface to PCI Bridge Registers (D30:F0)" provides a detailed description of all registers that reside in the Hub Interface to PCI bridge. This bridge resides at Device 30, Function 0 (D30:F0).

Chapter 8, "LPC I/F Bridge Registers (D31:F0)" provides a detailed description of all registers that reside in the LPC bridge. This bridge resides at Device 31, Function 0 (D31:F0). This function contains registers for many different units within the Intel® 6300ESB ICH including DMA, Timers, Interrupts, CPU Interface, GPIO, Power Management, System Management and RTC.

Chapter 9, "IDE Controller Registers (D31:F1)" provides a detailed description of all registers that reside in the IDE controller. This controller resides at Device 31, Function 1 (D31:F1).

Chapter 10, "USB UHCI Controllers Registers" provides a detailed description of all registers that reside in the three UHCI host controllers. These controllers reside at Device 29, Functions 0, 1 and 2 (D29:F0/F1/F2).

Chapter 11, "USB EHCI Controller Registers (D29:F7)" provides a detailed description of all registers that reside in the EHCI host controller. This controller resides at Device 29, Function 7 (D29:F7).

Chapter 12, "SMBUS Controller Registers (D31:F3)" provides a detailed description of all registers that reside in the SMBus controller. This controller resides at Device 31, Function 3 (D31:F3).

Chapter 13, “AC’97 Audio Controller Registers (D31:F5)” provides a detailed description of all registers that reside in the audio controller. This controller resides at Device 31, Function 5 (D31:F5). Note that this section of the EDS does not include the native audio mixer registers. Accesses to the mixer registers are forwarded over the AC-link to the codec where the registers reside.

Chapter 14, “AC’97 Modem Controller Registers (D31:F6)” provides a detailed description of all registers that reside in the modem controller. This controller resides at Device 31, Function 6 (D31:F6). Note that this section of the EDS does not include the modem mixer registers. Accesses to the mixer registers are forwarded over the AC-link to the codec where the registers reside.

Chapter 15, “Multimedia Timer Registers” provides a detailed description of all registers that reside in the multimedia event timer memory mapped register space.

Chapter 16, “Watchdog Timer (WDT) (D29:F4)” provides a detailed description of the configuration registers in the WDT controller. These registers reside at Device 29, Function 4 (D29:F4).

Chapter 17, “APIC1 Configuration Registers (D29:F5)” provides a detailed description of the configuration registers in the APIC1 controller. These registers reside at Device 29,
Function 5 (D29:F5).

Chapter 18, “PCI-X Overview (D28:F0)” provides a detailed description of the configuration registers of the PCI-X controller. These registers reside at Device 28, Function 0 (D28:F0).

Chapter 19, “Serial I/O Unit” describes the SIU, its features, LPC interface, serial ports and Port 60/64 Emulation along with a description of the registers in the SIU. These registers reside at Device 31, Function 0 (D31:F0).

Chapter 20, “Serial ATA Controller Registers (D31:F2)” provides a detailed description of the registers that reside in the SATA controller which encompasses a PCI device. This controller resides at Device 31, Function 2 (D31:F2).

Chapter 21, “Package Information” provides ballout information, signal lists and mechanical drawings.

Chapter 22, “Electrical Characteristics” provides AC and DC characteristics and AC timings.

Chapter 23, “Testability” provides information on test modes and scan chains.

This page intentionally left blank.

Intel® 6300ESB ICH and System Clock Domains 2

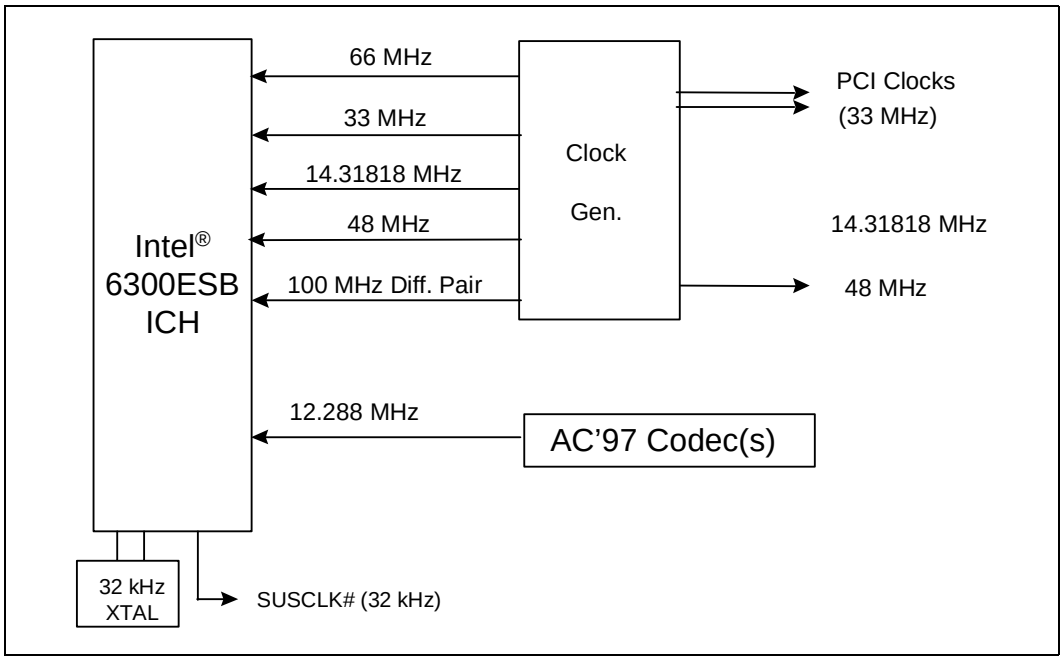
Warning: This section is provided for background purposes and should not be considered by implementers and validators as part of the behavioral definition of the Intel® 6300ESB ICH.

Table 2 presents the Intel® 6300ESB ICH clock domains. Figure 4 shows the assumed connection of the various system components, including the clock generator in desktop systems. For complete details of the system clocking solution, refer to the system's clock generator component specification.

Table 2. Intel® 6300ESB ICH Clock Domains

Clock Domain	Frequency	Source	Usage
Hub Interface	66.66 MHz	Main Clock Generator	Hub Interface, CPU I/F, AGP. Shut off during S3 or below.
PCI	33.33 MHz	Main Clock Generator	The Intel® 6300ESB ICH, PCI, LPC I/F. This remains on during S0-S1 states, and may be shut off during S3-S5. The PCI clock to peripherals may be shut off using the CLKRUN# protocols.
USB/SIO	48.00 MHz	Main Clock Generator	USB 1.0 Controllers in the Intel® 6300ESB ICH, External Super I/O. Shut off in S3 or below.
USB2.0	48.00 MHz	Internal	USB 2.0 logic close to the pins. Shut off when the 48 MHz clock input is shut off (S1 for low power, as well as S3 or below)
OSC	14.31818 MHz	Main Clock Generator	Used by ACPI timer. Also used by the Multimedia Timers Logic. Shut off in S1 for low power, as well as S3 or below.
AC'97	12.288 MHz	AC'97 Codec	AC'97 Link. Generated by AC'97 CODEC. May be shut off by codec in D3, as well as S1 for low power, and S3 or below.
RTC	32.768 KHz	Intel® 6300ESB ICH	RTC, Power Management. The Intel® 6300ESB ICH has its own oscillator. Always running, even in G3 state.
PCI-X	66 MHz	External	
UART	14.7456, 48 MHz	Main Clock Generator and/or discrete clock circuit.	UART clock input. **NOTE:** Some clock chips provide a 14.318x MHz clock output. The Intel® 6300ESB ICH's UART clock must use a 14.7456 MHz frequency; most clock chips do not provide this frequency. An option will be to use the 48.0 MHz clock.
SATA	100 MHz	Main Clock Generator	1.5 GHz clock generated internal to the Intel® 6300ESB ICH for use by SATA phy.

Figure 4. Conceptual System Clock Diagram



Signal Description 3

This section provides a detailed description of each signal. The signals are arranged in functional groups according to their associated interface.

The "#" symbol at the end of the signal name indicates that the active, or asserted state occurs when the signal is at a low voltage level. When "#" is not present, the signal is asserted when at the high voltage level.

The following notations are used to describe the signal type:

I	Input Pin
O	Output Pin
OD	Open Drain Output Pin
I/O	Bi-directional Input/Output Pin

3.1 Hub Interface to Host Controller

Table 3. Hub Interface Signals

Name	Type	Description
HI[11:0]	I/O	**Hub Interface Signals**
HI_STBS	I/O	**Hub Interface Strobe Second:** One of two differential strobe signals used to transmit and receive data through the Hub Interface. Hub Interface 1.5 mode this signal is not differential and is the second of the two strobe signals.
HI_STBF	I/O	**Hub Interface Strobe First:** One of two differential strobe signals used to transmit and receive data through the Hub Interface. Hub Interface 1.5 mode this signal is not differential and is the first of the two strobe signals.
HICOMP	I/O	**Hub Interface Compensation:** Used for Hub Interface buffer compensation. **NOTE:** The Intel® 6300ESB ICH will only support RCOMP, not the ZCOMP mode.
HICLK	I	**Hub Interface Clock:** 66 MHz clock input for Hub Interface. It is also used for some other internal units. This clock will stop during S3-S5 states.

Table 3. Hub Interface Signals

HIREF	I	**Hub Interface Voltage Reference. Analog input, expected voltage 350mV**
HI_VSWING	I	**Hub Interface Voltage Swing:** Analog input used to control the voltage swing and impedance strength of Hub Interface pins. Expected voltage is 800 mV. **NOTES:** 1. Refer to the platform design guide for expected voltages. 2. Refer to the platform design guide for resistor values and routing guidelines for each Hub Interface mode.

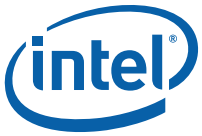
NOTES:
1. The Hub Interface signals are all in a separate power plane, called the Hub Interface plane.
2. During the S3, S4, and S5 states, power to the Hub Interface is assumed to be off. During S0 and S1 states, power to the Hub Interface must be on.

3.2 Firmware Hub Interface

Table 4. Firmware Hub Interface Signals

Name	Type	Description
FWH[3:0] / LAD[3:0]	I/O	**Firmware Hub Signals.** Muxed with LPC address signals. Internal pull-ups are provided.
FWH[4] / LFRAME#	I/O	**Firmware Hub Signals.** Muxed with LPC LFRAME# signal. LFRAME#: Indicates the start of an LPC cycle, or an abort.

NOTE: All LPC/FWH signals are in the core well.



3.3 PCI Interface

Table 5. PCI Interface Signals (Sheet 1 of 3)

Signal Name	Type	Description
AD[31:0]	I/O	**PCI Address/Data:** AD[31:0] signals are multiplexed. During the first clock of a transaction, AD[31:0] contain the physical address (32 bits). After the first clock, AD[31:0] contain data.
C/BE[3:0]#	I/O	**Bus Command and Byte Enables:** The command and byte enable signals are multiplexed. During the address phase of a transaction, C/BE[3:0]# define the bus command. During the data phase C/BE[3:0]# are used as Byte Enables. All command encoding not shown are reserved. C/BE[3:0]# Command Type Comment 0 0 0 0 Interrupt Acknowledge 0 0 0 1 Special Cycle 0 0 1 0 I/O Read 0 0 1 1 I/O Write 0 1 1 0 Memory Read 0 1 1 1 Memory Write 1 0 1 0 Configuration Read 1 0 1 1 Configuration Write 1 1 0 0 Memory Read Multiple 1 1 0 1 DAC Mode Address to be latched (target only) 1 1 1 0 Memory Read Line 1 1 1 1 Memory Write and Invalidate The Intel® 6300ESB ICH will not use reserved values, and will not respond if a PCI master generates a cycle using a reserved value. See PCI section for details on how these commands are supported depending on the Intel® 6300ESB ICH's role in the PCI cycle (target or initiator). As a target, the Intel® 6300ESB ICH can support DAC mode addressing for 44 bits.
DEVSEL#	I/O	**Device Select:** The Intel® 6300ESB ICH asserts DEVSEL# to claim a PCI transaction. As an output, the Intel® 6300ESB ICH asserts DEVSEL# when a PCI master peripheral attempts an access to an internal Intel® 6300ESB ICH address or an address destined for Hub Interface (main memory or AGP). As an input, DEVSEL# indicates the response to an Intel® 6300ESB ICH-initiated transaction on the PCI bus. DEVSEL# is tri-stated from the leading edge of PXPCIRST#. DEVSEL# remains tri-stated by the Intel® 6300ESB ICH until driven as a target.
FRAME#	I/O	**Cycle Frame:** FRAME# is driven by the current Initiator to indicate the beginning and duration of an access. While FRAME# is asserted data transfers continue. When FRAME# is negated the transaction is in the final data phase. FRAME# is an input to the Intel® 6300ESB ICH when it is the Target. FRAME# is an output when the Intel® 6300ESB ICH is the initiator. FRAME# remains tri-stated by the Intel® 6300ESB ICH until driven as an initiator.
IRDY#	I/O	**Initiator Ready:** IRDY# indicates the Intel® 6300ESB ICH's ability, as an Initiator, to complete the current data phase of the transaction. It is used in conjunction with TRDY#. A data phase is completed on any clock both IRDY# and TRDY# are sampled asserted. During a write, IRDY# indicates the Intel® 6300ESB ICH has valid data present on AD[31:0]. During a read, it indicates the Intel® 6300ESB ICH is prepared to latch data. IRDY# is an input to the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the Target and an output when the Intel® 6300ESB ICH is an Initiator. IRDY# remains tri-stated by the Intel® 6300ESB ICH until driven as an initiator.

Table 5. PCI Interface Signals (Sheet 2 of 3)

Signal Name	Type	Description
TRDY#	I/O	**Target Ready:** TRDY# indicates the Intel® 6300ESB ICH's ability to complete the current data phase of the transaction. TRDY# is used in conjunction with IRDY#. A data phase is completed when both TRDY# and IRDY# are sampled asserted. During a read, TRDY# indicates that the Intel® 6300ESB ICH, as a Target, has placed valid data on AD[31:0]. During a write, it indicates the Intel® 6300ESB ICH, as a Target is prepared to latch data. TRDY# is an input to the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the Initiator and an output when the Intel® 6300ESB ICH is a Target. TRDY# is tri-stated from the leading edge of PXPCIRST#. TRDY# remains tri-stated by the Intel® 6300ESB ICH until driven as a target.
STOP#	I/O	**Stop:** STOP# indicates that the Intel® 6300ESB ICH, as a Target, is requesting an initiator to stop the current transaction. As an Initiator, STOP# causes the Intel® 6300ESB ICH to stop the current transaction. STOP# is an output when the Intel® 6300ESB ICH is a Target and an input when the Intel® 6300ESB ICH is an Initiator. STOP# is tri-stated from the leading edge of PXPCIRST#, and remains tri-stated until driven by the Intel® 6300ESB ICH as a slave.
PAR	I/O	**Calculated/Checked Parity:** PAR is "even" parity and is calculated on 36 bits – AD[31:0] plus C/BE[3:0]#. "Even" parity means that the number of "1"s within the 36 bits plus PAR are counted and the sum is always even. PAR is always calculated on 36 bits regardless of the valid byte enables. PAR is generated for address and data phases, and is only ensured to be valid one PCI clock after the corresponding address or data phase. PAR is driven and tri-stated identically to the AD[31:0] lines, except that PAR is delayed by exactly one PCI clock. PAR is an output during the address phase (delayed one clock) for all Intel® 6300ESB ICH initiated transactions. It is also an output during the data phase (delayed one clock) when the Intel® 6300ESB ICH is the Initiator of a PCI write transaction, and when it is the Target of a read transaction. The Intel® 6300ESB ICH checks parity on the data phase when it is the Initiator of PCI read transactions and when it is the Target of PCI write transactions. It also checks parity on the address phase when it is the target of PCI transitions. If a parity error is detected, the Intel® 6300ESB ICH will set the appropriate internal status bits, and has the option to generate an NMI# or SMI#.
PERR#	I/O	**Parity Error:** Driven by an external PCI device when it receives data that has a parity error. Driven by the Intel® 6300ESB ICH when it detects a parity error. The Intel® 6300ESB ICH can either generate an NMI# or SMI# upon detecting a parity error (either detected internally or reported via PERR# signal) when serving as an initiator.
REQ[0:3]#	I	**PCI Requests:** Supports up to 4 external masters on the PCI bus.
GNT[0:3]#	O	**PCI Grants:** Supports up to 4 external masters on the PCI bus.
PCICLK	I	**NOTE: PCI Clock:** 33 MHz clock. PCICLK provides timing for all transactions on the PCI Bus, as well as many units inside the Intel® 6300ESB ICH. This clock can be stopped in S1 or S3, S4, or S5 states. This signal is not 5 V tolerant.

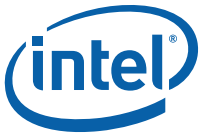


Table 5. PCI Interface Signals (Sheet 3 of 3)

Signal Name	Type	Description
PLOCK#	I/O	**PCI Lock:** Indicates an exclusive bus operation and may require multiple transactions to complete. The Intel® 6300ESB ICH ICH asserts PLOCK# when it is doing non-exclusive transactions on PCI. PLOCK# is ignored when PCI masters are granted the bus.
SERR#	I/OD	**System Error:** SERR# can be pulsed active by any PCI device that detects a system error condition. Upon sampling SERR# active, the Intel® 6300ESB ICH can be programmed to generate an NMI or SMI#. Implemented as I/O open drain. This allows the Intel® 6300ESB ICH to drive these signals due to internal sources.
PME#	I/OD	**PCI Power Management Event:** Driven by PCI peripherals to wake the system from low-power states S1-S5. If can also cause an SCI from the S0 state. Note that in some cases the Intel® 6300ESB ICH may drive PME# active (low) due to an internal wake event. It will not drive PME# high (but it may be pulled up using the internal pull-up resistor). **NOTE:** PME# is in the Resume power plane and has an internal pull-up resistor. **NOTE:** PME# is also used in the PCI-X segment.

3.4 PCI-X Interface

Table 6. PCI-X Interface Signals (Sheet 1 of 4)

Name	Type	Description
PXAD[31:0]	I/O	**PCI-X Address/Data:** These signals are a multiplexed address and data bus. During the address phase or phases of a transaction, the initiator drives a physical address on PXAD[31:0]. During the data phases of a transaction, the initiator drives write data, or the target drives read data. The Intel® 6300ESB ICH will drive all 0's on PXAD[31:0] during the address phase of all PCI-X Special Cycles.
PXAD[63:32]	I/O	**PC-X Address/Data:** These signals are a multiplexed address and data bus. This bus provides an additional 32 bits to the PCI-X bus. During the data phases of a transaction, the initiator drives the upper 32 bits of 64-bit write data, or the target drives the upper 32 bits of 64-bit read data, when PXREQ64# and PXACK64# are both asserted. When not driven PXAD[63:32] are pulled up to a valid logic level through external resistors. **NOTE:** When not driven PXAD[63:32] are pulled up to a valid logic level through external resistors.
PXC/ BE#[3:0]	I/O	**Bus Command and Byte Enables**: The command and byte enable signals are multiplexed on the same PCI-X pins. During the address phase of a transaction, PXC/BE#[3:0] define the bus command. During the data phase PXC/BE[3:0]# define the Byte Enables. **PXC/BE#[3:0] Command Type** 0 0 0 0 Interrupt Acknowledge 0 0 0 1 Special Cycle 0 0 1 0 I/O Read 0 0 1 1 I/O Write 0 1 1 0 Memory Read 0 1 1 1 Memory Write 1 0 1 0 Configuration Read 1 0 1 1 Configuration Write 1 1 0 0 Memory Read Multiple 1 1 1 0 Memory Read Line 1 1 1 1 Memory Write and Invalidate All command encodings not shown are reserved. The Intel® 6300ESB ICH does not decode reserved values, and therefore will not respond when a PCI-X master generates a cycle using one of the reserved values.
PXC/ BE#[7:4]	I/O	**Bus Command and Byte enables upper 4 bits:** These signals are a multiplexed command field and byte enable field. For both reads and write transactions, the initiator will drive byte enables for the PXAD[63:32] data bits on PXC/BE#[7:4] during the data phases when PXREQ64# and PXACK64# are both asserted. When not driven, PXC/BE#[7:4] are pulled up to a valid logic level through external resistors.
PXDEVSEL#	I/O	**Device Select:** The Intel® 6300ESB ICH asserts PXDEVSEL# to claim a PCI transaction. As an output, the Intel® 6300ESB ICH asserts DEVSEL# when a PCI master peripheral attempts an access to an internal Intel® 6300ESB ICH address or an address destined for the Hub Interface (main memory or AGP). As an input, DEVSEL# indicates the response to an Intel® 6300ESB ICH-initiated transaction on the PCI bus. DEVSEL# is tri-stated from the leading edge of PXPCIRST#. PXDEVSEL# remains tri-stated by the Intel® 6300ESB ICH until driven by a Target device.

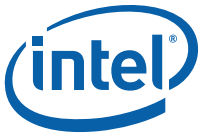


Table 6. PCI-X Interface Signals (Sheet 2 of 4)

Name	Type	Description
PXFRAME#	I/O	**Cycle Frame:** The current Initiator drives PXFRAME# to indicate the beginning and duration of a PCI transaction. While the initiator asserts PXFRAME#, data transfers continue. When the initiator negates PXFRAME#, the transaction is in the final data phase.PXFRAME# is an input to the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the target, and PXFRAME# is an output from the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the Initiator. PXFRAME# remains tri-stated by the Intel® 6300ESB ICH until driven by an Initiator.
PXIRDY#	I/O	**Initiator Ready:** PXIRDY# indicates the Intel® 6300ESB ICH's ability, as an Initiator, to complete the current data phase of the transaction. It is used in conjunction with PXTRDY#. A data phase is completed on any clock both PXIRDY# and PXTRDY# are sampled asserted. During a write, PXIRDY# indicates the Intel® 6300ESB ICH has valid data present on PXAD[31:0]. During a read, it indicates the Intel® 6300ESB ICH is prepared to latch data. PXIRDY# is an input to the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the Target and an output from the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is an Initiator. PXIRDY# remains tri-stated by the Intel® 6300ESB ICH until driven by an Initiator.
PXTRDY#	I/O	**Target Ready:** PXTRDY# indicates the Intel® 6300ESB ICH's ability as a Target to complete the current data phase of the transaction PXTRDY# is used in conjunction with IRDY#. A data phase is completed when both PXTRDY# and PXIRDY# are sampled asserted. During a read, PXTRDY# indicates that the Intel® 6300ESB ICH, as a Target, has placed valid data on PXAD[31:0]. During a write, PXTRDY# indicates the Intel® 6300ESB ICH, as a Target is prepared to latch data PXTRDY# is an input to the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is the Initiator and an output from the Intel® 6300ESB ICH when the Intel® 6300ESB ICH is a Target. PXTRDY# is tri-stated from the leading edge of PXPCIRST#. PXTRDY# remains tri-stated by the Intel® 6300ESB ICH until driven by a target.
PXSTOP#	I/O	**Stop:** PXSTOP# indicates that the Intel® 6300ESB ICH, as a Target, is requesting the Initiator to stop the current transaction. PXSTOP# causes the Intel® 6300ESB ICH, as an Initiator, to stop the current transaction. PXSTOP# is an output when the Intel® 6300ESB ICH is a Target and an input when the Intel® 6300ESB ICH is an Initiator. PXSTOP# is tri-stated from the leading edge of PXPCIRST#. PXSTOP# remains tri-stated until driven by the Intel® 6300ESB ICH.
PXPAR	I/O	**Calculated/Checked Parity:** PXPAR uses “even” parity calculated on 36 bits, PXAD[31:0] plus PXC/BE[3:0]#. “Even” parity means that the Intel® 6300ESB ICH counts the number of “1”s within the 36 bits plus PXPAR and the sum is always even. The Intel® 6300ESB ICH always calculates PXPAR on 36 bits regardless of the valid byte enables. The Intel® 6300ESB ICH generates PXPAR for address and data phases and only ensures PXPAR to be valid one PCI clock after the corresponding address or data phase. The Intel® 6300ESB ICH drives and tri-states PXPAR identically to the PXAD[31:0] lines except that the Intel® 6300ESB ICH delays PAR by exactly one PXPCI clock. PXPAR is an output during the address phase (delayed one clock) for all Intel® 6300ESB ICH initiated transactions. PXPAR is an output during the data phase (delayed one clock) when the Intel® 6300ESB ICH is the Initiator of a PCI-X write transaction, and when it is the Target of a read transaction. The Intel® 6300ESB ICH checks parity when it is the Target of a PC-X write transaction. When a parity error is detected, the Intel® 6300ESB ICH will set the appropriate internal status bits, and has the option to generate an NMI# or SMI#.

Table 6. PCI-X Interface Signals (Sheet 3 of 4)

Name	Type	Description
PXPERR#	I/O	**Parity Error:** An external PCI-X device drives PXPERR# when it receives data that has a parity error. The Intel® 6300ESB ICH drives PXPERR# when it detects a parity error. The Intel® 6300ESB ICH may either generate an NMI# or SMI# upon detecting a parity error (either detected internally or reported via the PXPERR# signal) when serving as an initiator.
PXREQ[1:0]# **PXREQ[2]#/GPIO[0]** **PXREQ[3]#/ GPIO[1]**	I	**PCI-X Requests:** Supports up to four masters on the PCI-X bus. The Intel® 6300ESB ICH accepts four request inputs, PXREQ[3:0]# into its internal bus arbiter. The Intel® 6300ESB ICH request input to the arbiter is an internal signal. **NOTE:** When operating in PCI 64b/66MHz, only two external masters should be used,PXREQ0, PXREQ1. PXREQ[2]# is muxed with GPIO[0] PXREQ[3]# is muxed with GPIO[1]
PXGNT[1:0]# **PXGNT[2]#/** GPIO[16] **PXGNT[3]#/** GPIO[17]	O	**PCI-X Grants:** Supports up to 4 masters on the PCI-X bus. PXGNT[2]# is muxed with GPIO[16] PXGNT[3]# is muxed with GPIO[17] **NOTE:** When operating in PCI 64b/66 MHz, only two external masters should be used: PXGNT0, PXGNT1.
PXPCLKI	I	**PCI-X Clock In:** This signal is connected to an output of the low skew PCI clock buffer tree(PXPCLKO[4]. It is used by the PLL to synchronize the PCI clock driven from PXPCLKO[4] to the clock used for the internal PCI-X logic.
PXPCLKO[4:0]	O	**PCI-X Clock Output:** 33/66 MHz clock for a PCI device. PXPCLKO[4] is connected to the PXPCLKI input. In PCI 64/66 mode PXPCLKO(1:0) are ensured to be driven.
PXPCICLK	I	**PCI-X Clock:** PXPCICLK is the clock for the internal PCI-X circuitry. 66Mhz primary input clock.
PXRCOMP	I/O	**Impedance Compensation:** Used to determine the impedance between the Intel® 6300ESB ICH and the PCI-X slots.
RASERR#	OD	**RAS Error:** This pin indicates that a RAS error has been logged. This is an active low signal that is a logical OR of all the RAS error events. If one of these errors is active, the pin is low. If none are active, then the pin is high.
PXPCIRST#	O	**PCI/PCI-X Reset:** The Intel® 6300ESB ICH asserts PXPCIRST# to reset devices that reside on the PCI-X bus. The Intel® 6300ESB ICH asserts PXPCIRST# during power-up and when S/W initiates a hard reset sequence through the RC (CF9h) register. The Intel® 6300ESB ICH drives PXPCIRST# inactive a minimum of 1 ms after PWROK is driven active. The Intel® 6300ESB ICH drives PXPCIRST# active a minimum of 1 ms when initiated through the RC (CF9h) register. **NOTE:** PXPCIRST# is in the Resume power plane. This signal also causes the legacy PCI bus and external PCI-X bus to reset
PXPLOCK#	I/O	**PCI-X Lock:** Indicates an exclusive bus operation and may require multiple transactions to complete. The Intel® 6300ESB ICH asserts PXPLOCK# when it performs exclusive transactions on the PCI-X bus. PLOCK# is ignored when PCI-X masters are granted the bus. The Intel® 6300ESB ICH does not propagate locked transaction upstream.

Table 6. PCI-X Interface Signals (Sheet 4 of 4)

Name	Type	Description
PXSERR#	I/OD	**System Error:** PXSERR# may be pulsed active by any PC-X device that detects a system error condition except Intel® 6300ESB ICH. The Intel® 6300ESB ICH samples PXSERR# as an input and conditionally forwards it to the Hub Interface. Upon sampling PXSERR# active, the Intel® 6300ESB ICH may be programmed to generate an NMI or SMI#.
PME#	I/OD	**PCI Power Management Event**: PCI-X peripherals drive PME# to wake the system from low-power states S1–S5. PME# assertion may also be enabled to generate a SCI from the S0 state. In some cases the Intel® 6300ESB ICH may drive PME# active due to an internal wake event. The Intel® 6300ESB ICH will not drive PME# high, but it will be pulled up to VccSus3_3 by an internal pull-up resistor. **NOTE:** PME# is in the Resume power plane and has an internal pull-up resistor. PME# control logic is in the primary PCI bus logic and not the PCI-X bridge
PXM66EN	I	**66MHz Enable:** This input signal from the PCI-X Bus indicates the speed of the PCI-X Bus. When it is high, the Bus speed is 66 MHz and when it is low, the bus speed is 33 MHz. This signal will be used to generate appropriate clock (33 or 66MHz) on the PCI-X Bus.
PXPCIXCAP	I	**PCI-X Capable:** Indicates whether all devices on the PCI-X bus are PCI-X devices, so that the Intel® 6300ESB ICH may switch into PCI-X mode
PXPAR64	I/O	**PCI-X interface upper 32-bits parity:** This carries the even parity of the 36 bits of PXAD[63:32] and PXC/BE#[7:4] for both address and data phases. When not driven, PXPAR64 is pulled up to a valid logic level through external resistors
PXREQ64#	I/O	**PCI-X interface request 64-bit transfer:** This is asserted by the initiator to indicate that the initiator is requesting a 64-bit data transfer. It has the same timing as PXFRAME#. When the Intel® 6300ESB ICH is the initiator, this signal is an output. When the Intel® 6300ESB ICH is the target this signal is an input
PXACK64#	I/O	**PCI-X interface acknowledge 64-bit transfer:** This is asserted by the target only when PXREQ64# is asserted by the initiator, to indicate the target's ability to transfer data using 64 bits. It has the same timing as PXDEVSEL#
PCIXSBRST#	O	**PCI-X Secondary Bus Reset:** The Intel® 6300ESB ICH asserts PCIXSBRST# to reset devices that reside on the PCI-X bus. The Intel® 6300ESB ICH asserts PCIXSBRST# when the PXPCIRST# pin is asserted or when SBR bit is set.

3.5 SATA Interface

Table 7. SATA Interface Signals

Name	Type	Description
SATA0TXP SATA0TXN	O	**Serial ATA 0 Differential Transmit Pair:** Outbound high speed differential signals to Port 0.
SATA0RXP SATA0RXN	I	**Serial ATA 0 Differential Receive Pair:** Inbound high speed differential signals from Port 0.
SATA1TXP SATA1TXN	O	**Serial ATA 1 Differential Transmit Pair:** Outbound high speed differential signals to Port 1.
SATA1RXP SATA1RXN	I	**Serial ATA 1 Differential Receive Pair:** Inbound high speed differential signals from Port 1.
SATACLKP, SATACLKN	i	**Differential SATA Clock:** 100 MHz clock input from the Clock Generator
SATALED#	OD	**Serial ATA LED#:** Output indicates Serial ATA Drive activity when it is driven low
SATARBIASP SATARBIASN	I	**Serial ATA Resistor Bias:** Analog connection point for a external resistor to ground.

3.6 IDE Interface

Table 8. IDE Interface Signals (Sheet 1 of 2)

Name	Type	Description
PDCS1#, SDCS1#	O	**Primary and Secondary IDE Device Chip Selects for 100 Range:** For ATA command register block. This output signal is connected to the corresponding signal on the primary or secondary IDE connector.
PDCS3#, SDCS3#	O	**Primary and Secondary IDE Device Chip Select for 300 Range:** For ATA control register block. This output signal is connected to the corresponding signal on the primary or secondary IDE connector.
PDA[2:0], SDA[2:0]	O	**Primary and Secondary IDE Device Address:** These output signals are connected to the corresponding signals on the primary or secondary IDE connectors. They are used to indicate which byte in either the ATA command block or control block is being addressed.
PDD[15:0], SDD[15:0]	I/O	**Primary and Secondary IDE Device Data:** These signals directly drive the corresponding signals on the primary or secondary IDE connector. There is a weak internal pull-down resistor on PDD[7] and SDD[7].
PDDREQ, SDDREQ	I	**Primary and Secondary IDE Device DMA Request:** These input signals are directly driven from the DRQ signals on the primary or secondary IDE connector. It is asserted by the IDE device to request a data transfer, and used in conjunction with the PCI bus master IDE function and are not associated with any AT compatible DMA channel. There is a weak internal pull-down resistor on these signals.

NOTES:
1. The IDE signals are 5V tolerant.
2. The IDE signals have integrated series terminating resistors.
3. All signals may be tri-stated or driven low for mobile swap bays.

Table 8. IDE Interface Signals (Sheet 2 of 2)

Name	Type	Description
PDDACK#, SDDACK#	O	**Primary and Secondary IDE Device DMA Acknowledge:** These signals directly drive the DAK# signals on the primary and secondary IDE connectors. Each is asserted by the Intel® 6300ESB ICH to indicate to IDE DMA slave devices that a given data transfer cycle (assertion of DIOR# or DIOW#) is a DMA data transfer cycle. This signal is used in conjunction with the PCI bus master IDE function and are not associated with any AT-compatible DMA channel.
PDIOR# / (PDWSTB / PRDMARDY#) **SDIOR# /** (SDWSTB / SRDMARDY#)	O	**Primary and Secondary Disk I/O Read (PIO and Non-Ultra DMA):** This is the command to the IDE device that it may drive data onto the PDD or SDD lines. Data is latched by the Intel® 6300ESB ICH on the deassertion edge of PDIOR# or SDIOR#. The IDE device is selected either by the ATA register file chip selects (PDCS1# or SDCS1#, PDCS3# or SDCS3#) and the PDA or SDA lines, or the IDE DMA acknowledge (PDDAK# or SDDAK#). **Primary and Secondary Disk Write Strobe (Ultra DMA Writes to Disk):** This is the data write strobe for writes to disk. When writing to disk, the Intel® 6300ESB ICH drives valid data on rising and falling edges of PDWSTB or SDWSTB. **Primary and Secondary Disk DMA Ready (Ultra DMA Reads from Disk):** This is the DMA ready for reads from disk. When reading from disk, the Intel® 6300ESB ICH deasserts PRDMARDY# or SRDMARDY# to pause burst data transfers.
PDIOW# / (PDSTOP) **SDIOW# /** (SDSTOP)	O	**Primary and Secondary Disk I/O Write (PIO and Non-Ultra DMA):** This is the command to the IDE device that it may latch data from the PDD or SDD lines. Data is latched by the IDE device on the deassertion edge of PDIOW# or SDIOW#. The IDE device is selected either by the ATA register file chip selects (PDCS1# or SDCS1#, PDCS3# or SDCS3#) and the PDA or SDA lines, or the IDE DMA acknowledge (PDDAK# or SDDAK#). **Primary and Secondary Disk Stop (Ultra DMA): the** Intel® 6300ESB ICH asserts this signal to terminate a burst.
PIORDY / (PDRSTB / PWDMARDY#) **SIORDY /** (SDRSTB / SWDMARDY#)	I	**Primary and Secondary I/O Channel Ready (PIO):** This signal will keep the strobe active (PDIOR# or SDIOR# on reads, PDIOW# or SDIOW# on writes) longer than the minimum width. It adds wait states to PIO transfers. **Primary and Secondary Disk Read Strobe (Ultra DMA Reads from Disk):** When reading from disk, the Intel® 6300ESB ICH latches data on rising and falling edges of this signal from the disk. **Primary and Secondary Disk DMA Ready (Ultra DMA Writes to Disk):** When writing to disk, this is deasserted by the disk to pause burst data transfers.

NOTES:
1. The IDE signals are 5V tolerant.
2. The IDE signals have integrated series terminating resistors.
3. All signals may be tri-stated or driven low for mobile swap bays.

3.7 LPC I/F

Table 9. LPC Interface Signals

Name	Type	Description
LAD[3:0] / FWH[3:0]	I/O	**LPC Multiplexed Command, Address, Data:** Internal pull-ups are provided.
LFRAME# / FWH[4]	I/O	**LPC Frame:** Indicates the start of an LPC cycle, or an abort.
LDRQ[1:0]#	I	**LPC Serial DMA/Master Request Inputs:** Used by LPC devices, such as Super I/O chips, to request DMA or bus master access.

NOTE: All LPC/FWH signals are in the core well

3.8 Interrupt Interface

Table 10. Interrupt Signals (Sheet 1 of 2)

Name	Type	Description
SERIRQ	I/O	**Serial Interrupt Request:** This pin implements the serial interrupt protocol.
PIRQ[D:A]#	I	**PCI Interrupt Requests:** In Non-APIC Mode the PIRQx# signals may be routed to interrupts 3, 4, 5, 6, 7, 9, 10, 11, 12, 14 or 15 as described in the Interrupt Steering section. Each PIRQx# line has a separate Route Control Register. In APIC mode, these signals are connected to the internal I/O APIC in the following fashion: PIRQ[A]# IRQ16 PIRQ[B]# IRQ17 PIRQ[C]# IRQ18 PIRQ[D]# IRQ19 This frees the legacy interrupts. These signals are 5V tolerant.

NOTE: The Interrupt signals are 5V tolerant except for PXIRQ [3:0]# / GPIO[36:33]

Table 10. Interrupt Signals (Sheet 2 of 2)

Name	Type	Description
PIRQ[H:E]# / GPIO[5:2]	I	**PCI Interrupt Requests:** In Non-APIC Mode the PIRQx# signals may be routed to interrupts 3, 4, 5, 6, 7, 9, 10, 11, 12, 14 or 15 as described in the Interrupt Steering section. Each PIRQx# line has a separate Route Control Register. These signals are 5 V tolerant. In APIC mode, these signals are connected to the internal I/O APIC in the following fashion: PIRQ[E] IRQ[20] PIRQ[F] IRQ[21] PIRQ[G] IRQ[22] PIRQ[H] IRQ[23] **NOTE:** When not connected as interrupts on the board, these pins must be programmed as GPI. This ensures proper interrupt delivery from internal sources regardless of the board termination on these pins.
IRQ[14-15]	I	**Interrupt Request 14–15:** These interrupt inputs are connected to the IDE drives. IRQ14 is used by the drives connected to the Primary controller and IRQ15 is used by the drives connected to the Secondary controller. These signals are 5 V tolerant.
PXIRQ[3:0]# / GPIO[36:33]	I	**PCI-X Bus Interrupt Request:** The PIRQ# lines from PCI-X interrupts, INTA, INTB, INTC, INTD], may be routed to these interrupt lines. PXIRQ[3:0]# are connected to an I/OxAPIC that resides on the PCI-X bus.

NOTE: The Interrupt signals are 5V tolerant except for PXIRQ [3:0]# / GPIO[36:33]

3.9 USB Interface

Table 11. USB Interface Signals

Name	Type	Description
USBP0P, USBP0N, USBP1P, USBP1N, USBP2P, USBP2N, USBP3P, USBP3N	I/O	**Universal Serial Bus Port 3:0 Differentials:** Bus Data/ Address/Command Bus. **NOTE:** No external resistors are required on these signals. The Intel® 6300ESB ICH integrates 15 kΩ pull-downs and provides an output driver impedance of 45 Ω which requires no external series resistor
OC[3:0]#	I	**Overcurrent Indicators:** These signals set corresponding bits in the USB controllers to indicate that an overcurrent condition has occurred.
USBRBIAS	O	**USB Resistor Bias:** Analog connection point for an external resistor to ground. USBRBIAS should be connected to USBRBIAS# as close to the resistor as possible.
USBRBIAS#	I	**USB Resistor Bias Complement:** Analog connection point for an external resistor to ground. USBRBIAS# should be connected to USBRBIAS as close to the resistor as possible.

NOTES:
1. The USB signals are all in the RESUME well.
2. Since OC[3:0]# are in the 5 V tolerant resume well, the external biasing resistors are not required.
3. All 4 ports support both USB1.0 and USB2.0 signaling.

3.10 Power Management Interface

Table 12. Power Management Interface Signals (Sheet 1 of 2)

Name	Type	Description
THRM#	I	**Thermal Alarm:** Active low signal generated by external hardware to start the Hardware clock throttling mode. May also generate an SMI# or SCI.
THRMTRIP#	I	**Thermal Trip:** When low, indicates that a thermal trip from the processor occurred, and corrective action will be taken. This input buffer has the same characteristics as the FERR# input buffer.
SLP_S3#	O	**S3 Sleep Control:** Power plane control. Shuts off power to all non-critical systems when in the S3 (Suspend To RAM) state.
SLP_S4#	O	**S4 Sleep Control:** Power plane control. Shuts power to non-critical systems when in the S4 (Suspend to Disk) or S5 (Soft Off) state.
SLP_S5#	O	**S5 Sleep Control:** Power plane control. The signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) state.
PWROK	I	**Power OK:** When asserted, PWROK is an indication to the Intel® 6300ESB ICH that core power and PCICLK have been stable for at least 1 ms. PWROK may be driven asynchronously. When PWROK is low, the Intel® 6300ESB ICH asserts PXPCIRST#. Traditional designs have a reset button logically ORed with the PWROK signal from the power supply and the processor's voltage regulator module. When this is done with the Intel® 6300ESB ICH, the PWROK_FLR bit will be set. The Intel® 6300ESB ICH treats this internally as though the RSMRST# signal had gone active. However, it is not treated as a full power failure. When PWROK goes inactive and then active (but RSMRST# stays high), the Intel® 6300ESB ICH will reboot (regardless of the state of the AFTERG3 bit). When RSMRST# also goes low before PWROK goes high, then this is a full power failure and the reboot policy is controlled by the AFTERG3 bit. PWROK must deassert for a minimum of 100 µseconds (simulation and analysis shows 3 RTC clock periods are required) in order to fully reset the core power well and properly generate the PXPCIRST# output.
PWRBTN#	I	**Power Button:** The Power Button will cause SMI# or SCI to indicate a system request to go to a sleep state. When the system is already in a sleep state, this signal will cause a wake event. When PWRBTN# is pressed for more than 4 seconds, this will cause an unconditional transition (power button override) to the S5 state with only the PWRBTN# available as a wake event. Override will occur even when the system is in the S1-S4 states. This signal has an internal pull-up resistor.
RI#	I	**Ring Indicate:** From the modem interface. May be enabled as a wake event, and this is preserved across power failures.
SYS_RESET#	I	**System Reset**: This pin forces an internal reset after being debounced.
RSMRST#	I	**Resume Well Reset:** Used for resetting the resume power plane logic. An external RC circuit is required to ensure that the resume well power is valid prior to RSMRST# going high.

NOTE: These signals are all in the RESUME well, except THRM# which is in the core well; PWROK and RSMRST# which are in the RTC well.

Table 12. Power Management Interface Signals (Sheet 2 of 2)

Name	Type	Description
SUS_STAT# / LPCPD#	O	**Suspend Status:** This signal is asserted by the Intel® 6300ESB ICH to indicate that the system will be entering a low power state soon. This may be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It may also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes. This signal is called LPCPD# on the LPC I/F.
SUSCLK	O	**Suspend Clock:** Output of the RTC generator circuit (32.768 KHz). SUSCLK will have a duty cycle that may be as low as 30% or as high as 70%.
VRMPWRGD	I	**Voltage Regulator Power Good:** Not implemented in the Intel® 6300ESB ICH. Pull this input high to Vcc.

NOTE: These signals are all in the RESUME well, except THRM# which is in the core well; PWROK and RSMRST# which are in the RTC well.

3.11 CPU Interface

Table 13. CPU Interface Signals (Sheet 1 of 2)

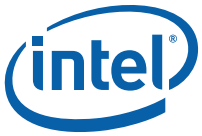
Name	Type	Description
A20M#	O	**Mask A20:** A20M# will go active based on either setting the appropriate bit in the Port 92h register, or based on the A20GATE input being active. Speed Strap: During the reset sequence, the Intel® 6300ESB ICH drives A20M# high when the corresponding bit is set in the FREQ_STRP register.
CPUSLP#	O	**CPU Sleep:** This signal puts the processor into a state that saves substantial power compared to Stop-Grant state. However, during that time, no snoops occur. The Intel® 6300ESB ICH may optionally assert the CPUSLP# signal when going to the S1 state. It will go active for all other sleep states.
FERR#	I	**Numeric Coprocessor Error:** This signal is tied to the coprocessor error signal on the processor. FERR# is only used when the Intel® 6300ESB ICH coprocessor error reporting function is enabled in the General Control Register (D31:F0:Offset D0.bit 5). When FERR# is asserted, the Intel® 6300ESB ICH generates an internal IRQ13 to its interrupt controller unit. It is also used to gate the IGNNE# signal to ensure that IGNNE# is not asserted to the processor unless FERR# is active. FERR# requires an external weak pull-up to ensure a high level when the coprocessor error function is disabled. FERR# may optionally be used in some states for notification by the processor of pending interrupt events.

NOTES:
1. The CPU I/F signals (except RCIN#, A20GATE, and FERR#) are on a separate power well. This saves the external pull-up resistors that were needed on previous chipsets.
2. RCIN# and A20GATE, and FERR# are on in the Core power well.

Table 13. CPU Interface Signals (Sheet 2 of 2)

Name	Type	Description
IGNNE#	O	**Ignore Numeric Error:** This signal is connected to the ignore error pin on the CPU. IGNNE# is only used when the Intel® 6300ESB ICH coprocessor error reporting function is enabled in the General Control Register (D31:F0:Offset D0.bit 5). When FERR# is active, indicating a coprocessor error, a write to the Coprocessor Error Register (F0h) causes the IGNNE# to be asserted. IGNNE# remains asserted until FERR# is negated. If FERR# is not asserted when the Coprocessor Error Register is written, the IGNNE# signal is not asserted.
INIT#	O	**Initialization:** INIT# is asserted by the Intel® 6300ESB ICH for 16 PCI clocks to reset the processor. The Intel® 6300ESB ICH may be configured to support CPU BIST. In that case, INIT# will be active when PXPCIRST# is active.
INTR	O	**CPU Interrupt:** INTR is asserted by the Intel® 6300ESB ICH to signal the processor that an interrupt request is pending and needs to be serviced. It is an asynchronous output and normally driven low.
NMI	O	**Non-Maskable Interrupt:** NMI is used to force a non-Maskable interrupt to the processor. The Intel® 6300ESB ICH may generate an NMI when either SERR# or IOCHK# is asserted. The processor detects an NMI when it detects a rising edge on NMI. NMI is reset by setting the corresponding NMI source enable/disable bit in the NMI Status and Control Register.
SMI#	O	**System Management Interrupt:** SMI# is an active low output synchronous to PCICLK. It is asserted by the Intel® 6300ESB ICH in response to one of many enabled hardware or software events.
STPCLK#	O	**Stop Clock Request:** STPCLK# is an active low output synchronous to PCICLK. It is asserted by the Intel® 6300ESB ICH in response to one of many hardware or software events. When the processor samples STPCLK# asserted, it responds by stopping its internal clock. This signal will not be connected to the processor in iA64 systems, since the processor has no corresponding input signal.
RCIN#	I	**Keyboard Controller Reset CPU:** The keyboard controller may generate INIT# to the processor. This saves the external OR gate with the Intel® 6300ESB ICH's other sources of INIT#. When the Intel® 6300ESB ICH detects the assertion of this signal, INIT# is generated for 16 PCI clocks. Note that the Intel® 6300ESB ICH will ignore RCIN# assertion during transitions to the S1, S3, S4 and S5 states.
A20GATE	I	**A20 Gate:** From the keyboard controller. Acts as an alternative method to force the A20M# signal active. Saves the external OR gate needed with various other chipsets.

NOTES:
1. The CPU I/F signals (except RCIN#, A20GATE, and FERR#) are on a separate power well. This saves the external pull-up resistors that were needed on previous chipsets.
2. RCIN# and A20GATE, and FERR# are on in the Core power well.



3.12 SMBus Interface

Table 14. SM Bus Interface Signals

Name	Type	Description
SMBDATA	I/OD	**SMBus Data:** External pull-up is required.
SMBCLK	I/OD	**SMBus Clock:** External pull-up is required.
SMBALERT #/ GPIO[11]	I	**SMBus Alert**: This signal is used to wake the system or generate SMI#. When not used for SMBALERT#, it may be used as a GPIO (GPIO[11]).

NOTE: The SMBus I/F signals are all in the RESUME well.

3.13 System Management Interface

Table 15. System Management Interface Signals

Name	Type	Description
INTRUDER #	I	**Intruder Detect:** Detects if the system case has been opened. May be set to disable the system when box is detected open. This signal's status is readable, so it may be used like a GPI when the Intruder switch is not needed.
SMLINK[1:0]	I/OD	**System Management Link:** SMBus link to optional external system management ASIC or LAN Controller. External pull-ups are required. Note that SMLINK[0] corresponds to a SMBus Clock signal, and SMLINK[1] corresponds to a SMBus Data signal.

NOTE: INTRUDER# is in the RTC well. The SMLINK signal is in the RESUME well.

3.14 Real Time Clock Interface

Table 16. Real Time Clock Interface

Name	Type	Description
RTCX1	Special	**Crystal Input 1:** Connected to the 32.768 KHz crystal. When no external crystal is used, then RTCX1 may be driven with the desired clock rate.
RTCX2	Special	**Crystal Input 2:** Connected to the 32.768 KHz crystal. When no external crystal is used, then RTCX2 should be left floating.
Vbias	I	**Bias Voltage for Oscillator:** Sets the proper biasing for the oscillator. Expected voltage 200 mV.

NOTE: An external Crystal/Resistor/Capacitor circuit is required for proper operation of the oscillator.

3.15 Other Clocks

Table 17. Other Clocks

Name	Type	Description
CLK14	I	**Oscillator Clock:** Used for 8254 timers. Runs at 14.31818 MHz. This clock is permitted to stop during S3 (or lower) states in desktop configurations or S1 (or lower) states in mobile configurations.
CLK48	I	**48 MHz Clock:** Used to run the USB controllers. Runs at 48 MHz. This clock is permitted to stop during S3 (or lower) states in desktop configurations or S1 (or lower) states in mobile configurations.
HICLK	I	**66 MHz Clock:** Used to run the Hub Interface. Runs at 66 MHz. This clock is permitted to stop during S3 (or lower) states in desktop configurations or S1 (or lower) states in mobile configurations.

3.16 Miscellaneous Signals

Table 18. Miscellaneous Signals

Name	Type	Description
SPKR	O	**Speaker:** The SPKR signal is the output of counter 2 and is internally "ANDed" with Port 61h bit 1 to provide Speaker Data Enable. This signal drives an external speaker driver device, which in turn drives the system speaker. Upon PXPCIRST#, its output state is 0. **NOTE:** SPKR is sampled at the rising edge of PWROK as a functional strap. See Section 3.21.1, "Functional Straps" for more details. There is a weak integrated pull-down resistor on SPKR pin.
RTCRST#	I	**RTC Reset:** When asserted, this signal resets register bits in the RTC well and sets the RTC_PWR_STS bit (bit 2 in GEN_PMCON3 register). **NOTES:** 1. Clearing CMOS in an Intel® 6300ESB ICH-based platform may be done by using a jumper on RTCRST# or GPI, or using SAFEMODE strap. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low. 2. Unless entering the XOR Chain Test Mode, the RTCRST# input must always be high when all other RTC power planes are on
WDT_TOUT# / GPIO[32]	O	**Watchdog Timer Timeout:** Driven active to indicate the second stage of the WDT has overflowed. This signal will toggle states for each overflow in periodic mode. In non-periodic mode, this signal will go active low and remain in this state until a system reset or power cycle. This signal is muxed with GPIO[32].

3.17 AC'97 Link

Table 19. AC'97 Link Signals

Name	Type	Description
AC_RST#	O	**AC'97 Reset:** Master H/W reset to external Codec(s).
AC_SYNC	O	**AC'97 Sync:** 48 KHz fixed rate sample sync to the Codec(s).
AC_BIT_CLK	I	**AC'97 Bit Clock:** 12.288 MHz serial data clock generated by the external Codec(s). This signal has an integrated pull-down resistor[3].
AC_SDOUT	O	**AC'97 Serial Data Out:** Serial TDM data output to the Codec(s).
AC_SDIN[2:0]	I	**AC'97 Serial Data In 2:0:** Serial TDM data input from the three Codec(s). Integrated pull-down resistors, which are always enabled.

NOTES:
1. These signals are in the RESUME well, except AC_SYNC, AC_BIT_CLK, and AC_SDATA_OUT, which are in the core well.
2. See Section 4.2, "Integrated Pull-Ups and Pull-Downs" for details about when the integrated pull-down resistors are enabled on AC_SYNC, AC_BIT_CLK, and AC_SDATA_OUT.
3. An integrated pull-down resistor on AC_BIT_CLK is enabled when either:
   - The ACLINK Shutoff bit in the AC'97 Global Control Register (See Section 13.2.8, "GLOB_CNT—Global Control Register") is set to '1', or
   - Both Function 5 and Function 6 of Device 31 are disabled. Otherwise, the integrated pull-down resistor is disabled.

3.18 Universal Asynchronous Receive and Transmit (UART0,1)

Table 20. Universal Asynchronous Receive and Transmit (UART0, 1) (Sheet 1 of 2)

Signal Name	Type	Description
UART_CLK	I	**Input clock to the SIU.** This clock is passed to the baud clock generation logic of each UART in the SIU.
SIU0_RXD SIU1_RXD	I	**SERIAL INPUTs for UART0 and UART1:** Serial data input from device pin to the receive port.
SIU0_TXD SIU1_TXD	O	**SERIAL OUTPUT for UART0, 1:** Serial data output to the communication peripheral/modem or data set. Upon reset, the TXD pins will be set to MARKING condition (logic '1' state).
SIU0_CTS SIU1_CTS	I	**CLEAR TO SEND:** Active low, this pin indicates that data may be exchanged between the Intel® 6300ESB ICH and external interface. These pins have no effect on the transmitter. **NOTE:** These pins could be used as Modem Status Input whose condition may be tested by the processor by reading bit 4 (CTS) of the Modem Status register (MSR). Bit 4 is the complement of the CTS# signal. Bit 0 (DCTS) of the MSR indicates whether the CTS# input has changed state since the previous reading of the MSR. When the CTS bit of the MSR changes state, an interrupt is generated when the Modem Status Interrupt is enabled.

Table 20. Universal Asynchronous Receive and Transmit (UART0, 1) (Sheet 2 of 2)

Signal Name	Type	Description
SIU0_DSR **SIU1_DSR**	I	**DATA SET READY for UART 0, 1:** Active low, this pin indicates that the external agent is ready to communicate with the Intel® 6300ESB ICH UARTS. These pins have no effect on the transmitter. **NOTE:** These pins could be used as Modem Status Input whose condition may be tested by the processor by reading bit 5 (DSR) of the Modem Status register. Bit 5 is the complement of the DSR# signal. Bit 1 (DDSR) of the Modem status register (MSR) indicates whether the DSR# input has changed state since the previous reading of the MSR. When the DSR bit of the MSR changes state, an interrupt is generated when the Modem Status Interrupt is enabled.
SIU0_DCD **SIU1_DCD**	I	**DATA CARRIER DETECT for UART 0, 1:** Active low, this pin indicates that data carrier has been detected by the external agent. **NOTE:** These pins are Modem Status Input whose condition may be tested by the processor by reading bit 7 (DCD) of the Modem Status register (MSR). Bit 7 is the complement of the DCD# signal. Bit 3 (DDCD) of the MSR indicates whether the DCD# input has changed state since the previous reading of the MSR. When the DCD bit of the MSR changes state, an interrupt is generated when the Modem Status Interrupt is enabled.
SIU0_RI# **SIU1_RI#**	I	**RING INDICATOR for UART 0, 1:** Active low, this pin indicates that a telephone ringing signal has been received by the external agent. **NOTE:** These pins are Modem Status Input whose condition may be tested by the processor by reading bit 6 (RI) of the Modem Status register (MSR). Bit 6 is the complement of the RI# signal. Bit 2 (TERI) of the MSR indicates whether the RI# input has transitioned back to an inactive state. When the RI bit of the MSR changes from a 1 to 0, an interrupt is generated when the Modem Status Interrupt is enabled.
SIU0_DTR# **SIU1_DTR#**	O	**DATA TERMINAL READY for UART 0, 1:** When low these pins informs the modem or data set that the Intel® 6300ESB ICH UART 0, 1 are ready to establish a communication link. The DTR#x(x=0,1) output signals may be set to an active low by programming the DTRx (x-0,1) (bit0) of the Modem control register to a logic ‘1’. A Reset operation sets this signal to its inactive state (logic ‘1’). LOOP mode operation holds this signal in its inactive state.
SIU0_RTS# **SIU1_RTS#**	O	**REQUEST TO SEND for UART 0, 1:** When low these pins informs the modem or data set that the Intel® 6300ESB ICH UART 0, 1are ready to establish a communication link. The RTS#x, where x = 0,1, output signals may be set to an active low by programming the RTS#x (bit1) of the Modem control register to a logic ‘1’. A Reset operation sets this signal to its inactive state (logic ‘1’). LOOP mode operation holds this signal in its inactive state.

3.19 General Purpose I/O

Note: *Alternative signal definition is for pin strap selected feature muxing. GPIOs muxing is based on GPIO configurations.*

Table 21. General Purpose I/O Signals (Sheet 1 of 2)

Signal Name	Type	Description
GPIO[0]/ PXREQ[2]#	I	Fixed as Input only. Main power well. May instead be used as PXREQ[2]#.
GPIO[1]/ PXREQ[3]#	I	Fixed as Input only. Main power well. May instead be used as PXREQ[3]#.
GPIO[5:2]/ PIRQ[H:E]#	I	Fixed as Input only. Main power well. May instead be used as PIRQ[H:E]#.
GPIO[6]	I	Fixed as Input only. Main power well. (3.3 V tolerant, not 5 V tolerant)
GPIO[7]	I	Fixed as Input only. Main power well. (3.3 V tolerant, not 5 V tolerant)
GPIO[8]	I	Fixed as Input only. Resume power well. Unmuxed. The GPI_INV bit corresponding to GPIO[8] must be set in order to achieve the correct polarity in the General Purpose Event 0 Status Register.
GPIO[9:10]	I	Reserved. These GPIO are not implemented.
GPIO[11]	I	Fixed as Input only. Resume power well. May instead be used for SMBAlert#.
GPIO[12:13]	I	Fixed as Input only. Resume power well.
GPIO[14:15]	I	Reserved. These GPIO are not implemented.
GPIO[16]	O	Fixed as Output only. Main power well. May instead be used as PXGNT[2]#.
GPIO[17]	O	Fixed as Output only. Main power well. May instead be used as PXGNT[3]#.
GPIO[18]	O	Fixed as Output only. Main power well.
GPIO[19]	O	Fixed as Output only. Main power well.
GPIO[20]	O	Fixed as Output only. Main power well.
GPIO[21]	O	Fixed as Output only. Main power well.
GPIO[23]	O	Fixed as Output only. Main power well.
GPIO[24]	I/O	May be input or output. Resume power well. Unmuxed.
GPIO[25]	I/O	May be input or output. Resume power well. Unmuxed.
GPIO[26]	I/O	Reserved. This GPIO is not implemented.
GPIO[27:28]	I/O	May be input or output. Resume power well. Unmuxed.
GPIO[29:31]	O	Reserved. These GPIO are not implemented.
GPIO[32]	O	Fixed as Output only. Core power well. May instead be used for WDT_TOUT#.
GPIO[33:36]	I/O	May be input or output. Core power well. May instead be used for PXIRQ[0:3]#.
GPIO[37:39]	I/O	May be input or output. Core power well. GPIO[37,39] are unmuxed.

NOTES:
1. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
2. GPIO[8:15] and GPIO[24:31] are in the suspend well.
3. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
4. Resume-well GPIO are not 5 V tolerant.
5. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.

Table 21. General Purpose I/O Signals (Sheet 2 of 2)

Signal Name	Type	Description
GPIO[40:43]	I/O	May be input or output. Core power well. **NOTE:** These GPIOs have High Strength Output Capability (for driving LEDs).
GPIO[44:55]	I/O	Reserved. These GPIO are not implemented.
GPIO[56:57]	OD	Output only. Resume and RTC power wells. Unmuxed.
GPIO[58:63]	I/O	Reserved. These GPIO are not implemented.

NOTES:
1. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
2. GPIO[8:15] and GPIO[24:31] are in the suspend well.
3. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
4. Resume-well GPIO are not 5 V tolerant.
5. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.

3.20 Power and Ground

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

Name	Description
VCC3_3	3.3 V supply for core well I/O buffers. This power may be shut off in S3, S4, or S5 states.
VCC1_5	1.5 V supply for core well logic. This power may be shut off in S3, S4, S5 or S5 states.
VCCHI	1.5 V supply for Hub Interface 1.5 logic. This power may be shut off in S3, S4, and S5 states.
V5REF	Reference for 5 V tolerance on core well inputs. This power may be shut off in S3, S4, S5 or G3 states.
VCCREF	3.3V reference voltage for PCI-X inputs
VCCA	1.5V supply for Hub Interface PLL. This power may be shut off in S3, S4, and S5 states.
HIREF	Analog Input. Expected voltages are 350 mV for the Hub Interface 1.5 (Enhanced Hub Interface) Parallel Termination. This power is shut off in S3, S4, S5 and G3 states.
VccSus3_3	3.3 V supply for resume well I/O buffers. This power is not expected to be shut off unless the system is unplugged in desktop configurations.
VCCSUS1_5	1.5 V supply for resume well logic. This power is not expected to be shut off unless the system is unplugged in desktop configurations.
V5REF_Sus	Reference for 5 V tolerance on resume well inputs. This power is not expected to be shut off unless the system is unplugged in desktop configurations.
VCCRTC	3.3 V (may drop to 2.0 V min. in G3 state) supply for the RTC well. This power is not expected to be shut off unless the RTC battery is removed or completely drained. **NOTE:** Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low. Clearing CMOS in an Intel® 6300ESB ICH-based platform may be done by using a jumper on RTCRST# or GPI, or using SAFEMODE strap.

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

Name	Description
VccPLL	1.5 V supply for core well logic. This power may be shut off in S3, S4, S5 or G3 states.
V_CPU_IO	Powered by the same supply as the CPU I/O voltage. This supply is used to drive the CPU I/F outputs. 0.8 to 1.75 V. The power will be shut in S3, S4, and S5 states.
Vss	Shared Grounds.

3.21 Pin Straps

3.21.1 Functional Straps

The following signals are used for static configuration. They are sampled at various reset points to select configurations, and then revert later to their standard usage. To invoke the associated mode, the signal should be driven at least four PCI clocks prior to the time it is sampled.

Note: The Intel® 6300ESB ICH changes the polarity of the “no reboot” strap in order to avoid an audible click due to the pull-up on the SPKR output.

Table 23. Functional Strap Definitions

Signal	Usage	When Sampled	Comment
AC_SDOUT	SAFE MODE	Rising Edge of PWROK	The signal has a weak internal pull-down. When the signal is sampled high, the Intel® 6300ESB ICH will set the processor speed strap pins for safe mode. Refer to processor specification for speed strapping definition. The status of this strap is readable via the SAFE_MODE bit (bit 2, D31: F0, Offset D4h).
SIU0_DTR#	A16 swap override	Rising Edge of PWROK	This signal has a weak internal pull-up. The Intel® 6300ESB ICH starts driving it when PXPCIRST# goes high. The status of this strap is readable via D31: F0, Offset D5h, bit 5).
SPKR	NO REBOOT	Rising Edge of PWROK	The signal has a weak internal pull-down. When the signal is sampled high, this indicates that the system is strapped to the “No Reboot” mode (The Intel® 6300ESB ICH will disable the TCO Timer system reboot feature). The status of this strap is readable via the NO_REBOOT bit (bit 1, D31: F0, Offset D4h).

3.22 Revision and Device ID Table

Table 24. Revision and Device ID Table

Device Function	Description	Device ID	A0 Rev ID	A1/A2 Rev ID	A3 Rev ID	Comments
D30,F0	Hub to PCI Bridge	244Eh	08h	09h	0Ah	
D31,F0	LPC Bridge	25A1h	00h	01h	02h	
D31,F1	IDE	25A2h	00h	01h	02h	
D31,F2	SATA Controller	25A3h	00h	01h	02h	SATA and RAID are mutually exclusive.
D31, F3	SMBus Controller	25A4h	00h	01h	02h	
D31, F5	AC'97 Audio	25A6h	00h	01h	02h	
D31, F6	AC'97 Modem	25A7h	00h	01h	02h	
D29, F0	USB UHCI #1	25A9h	00h	01h	02h	
D29, F1	USB UHCI #2	25AAh	00h	01h	02h	
D29,F4	WDT	25ABh	00h	01h	02h	
D29:F5	IOxAPIC	25ACh	00h	01h	02h	
D29, F7	USB EHCI	25ADh	00h	01h	02h	
D28:F0	Hub interface to PCI-X Bridge	25AEh	00h	01h	02h	
D31:F2	RAID Controller	25B0h	00h	01h		SATA and RAID are mutually exclusive.

NOTE: Refer to the latest Intel® 6300ESB I/O Controller Hub *Specification Update* for the value of the Revision Identification Registers.

Intel® 6300ESB ICH Power Planes and Pin States 4

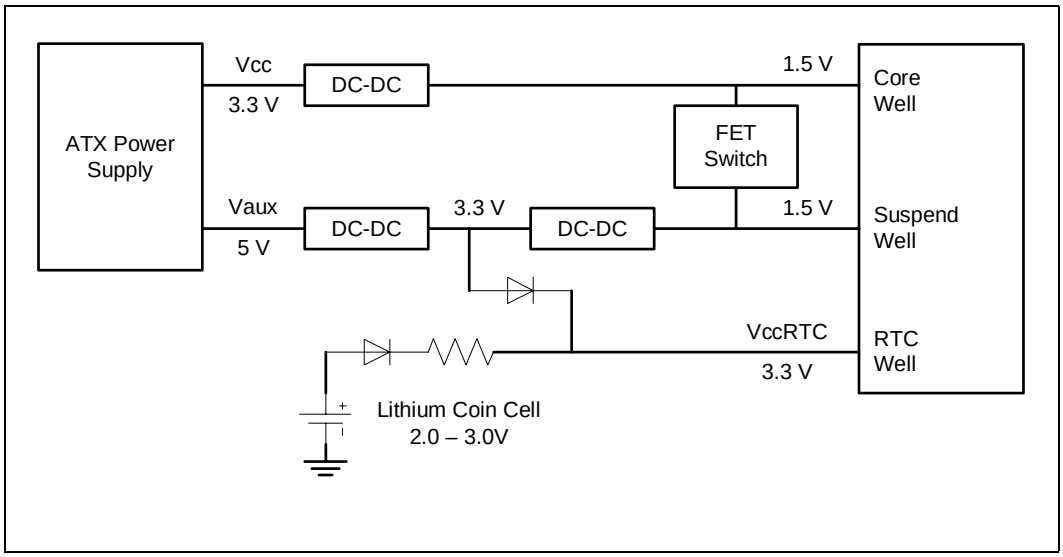
4.1 Power Planes

Table 25. Intel® 6300ESB I/O Controller Hub Power Planes

Plane	Description
Main I/O (3.3 V)	**Vcc3_3:** Powered by the main power supply. When the system is in the S3, S4, S5, or G3 state, this plane is assumed to be shut off.
Main Logic (1.5 V)	**Vcc1_5:** Powered by the main power supply. When the system is in the S3, S4, S5, or G3 state, this plane is assumed to be shut off.
Resume I/O (3.3 V Standby)	**VccSUS3_3:** Powered by the main power supply in S0 - S1 states. Powered by the trickle power supply when the system is in the S3, S4, S5, state. Assumed to be shut off only when in the G3 state (system is unplugged and AC power is not present).
Resume Logic (1.5 V Standby)	**VccSUS1_5:** Powered by the main power supply in S0 - S1 states. Powered by the trickle power supply when the system is in the S3, S4, S5, state. Assumed to be shut off only when in the G3 state (system is unplugged and AC power is not present).
CPU I/F (0.8 ~ 1.75 V)	**V_CPU_IO:** Powered by the main power supply through the processor voltage regulator. When the system is in the S3, S4, S5, or G3 state, this plane is assumed to be shut off.
Hub Interface Logic (1.5 V)	**VccHI:** Powered by the main power supply. Assumed to be 1.5 V. When the system is in the S3, S4, S5, or G3 state, this plane is assumed to be shut off.
RTC	**VccRTC:** When other power is available (from the main supply), external diode coupling will provide power to reduce the drain on the RTC battery. Assumed to operate from 3.3 V down to 2.0 V.

USAGE MODEL ASSUMPTION: The power planes and control are shown in Figure 5.

Figure 5. Power Plane Usage Model



4.2 Integrated Pull-Ups and Pull-Downs

Table 26. Integrated Pull-Up and Pull-Down Resistors

Signal	Resistor Type	Nominal Value	Notes
AC_BITCLK	pull-down	20K	1
AC_RST#	pull-down	20K	2
AC_SDIN[2:0]	pull-down	20K	3
AC_SDOUT	pull-down	20K	2, 8
AC_SYNC	pull-down	20K	2, 8
DPRSLPVR	pull-down	20K	2
EE_DIN	pull-up	20K	3
EE_DOUT	pull-up	20K	3
GNT[B:A]# / GNT[5]# / GPIO[17:16]	pull-up	20K	3
LAD[3:0]# / FWH[3:0]#	pull-up	20K	3
LDRQ[1:0]	pull-up	20K	3
PME#	pull-up	20K	3
PWRBTN#	pull-up	20K	3
PDD[7] / SDD[7]	pull-down	11.5K	6

NOTES:
1. Simulation data shows that these resistor values may range from 10 KΩ to 40 KΩ.
2. Simulation data shows that these resistor values may range from 9 KΩ to 50 KΩ.
3. Simulation data shows that these resistor values may range from 15 KΩ to 35 KΩ
4. Simulation data shows that these resistor values may range from 7.5 KΩ to 16 KΩ.
5. Simulation data shows that these resistor values may range from 45 KΩ to 170 KΩ
6. Simulation data shows that these resistor values may range from 5.7 KΩ to 28.3 KΩ.
7. Simulation data shows that these resistor values may range from 14.25 KΩ to 24.8 KΩ
8. The pull-up or pull-down on this signal is only enabled at boot/reset for strapping function.

Table 26. Integrated Pull-Up and Pull-Down Resistors

Signal	Resistor Type	Nominal Value	Notes
PDDREQ / SDDREQ	pull-down	11.5K	6
SPKR	pull-down	20K	2, 8
USB[3:0] [P,N]	pull-down	15K	7

NOTES:
1. Simulation data shows that these resistor values may range from 10 KΩ to 40 KΩ.
2. Simulation data shows that these resistor values may range from 9 KΩ to 50 KΩ.
3. Simulation data shows that these resistor values may range from 15 KΩ to 35 KΩ
4. Simulation data shows that these resistor values may range from 7.5 KΩ to 16 KΩ.
5. Simulation data shows that these resistor values may range from 45 KΩ to 170 KΩ
6. Simulation data shows that these resistor values may range from 5.7 KΩ to 28.3 KΩ.
7. Simulation data shows that these resistor values may range from 14.25 KΩ to 24.8 KΩ
8. The pull-up or pull-down on this signal is only enabled at boot/reset for strapping function.

4.3 IDE Integrated Series Termination Resistors

Table 27 shows the Intel® 6300ESB ICH IDE signals that have integrated series termination resistors.

Table 27. IDE Series Termination Resistors

Signal	Integrated Series Termination Resistor Value
PDD[15:0], SDD[15:0], PDIOW#, SDIOW#, PDIOR#, PDIOW#, PDREQ, SDREQ, PDDACK#, SDDACK#, PIORDY, SIORDY, PDA[2:0], SDA[2:0], PDCS1#, SDCS1#, PDCS3#, SDCS3#, IRQ14, IRQ15]	approximately 33 Ω (See Note)

NOTE: Simulation data indicates that the integrated series termination resistors are a nominal 33 Ω but may range from 31 Ω to 43 Ω.

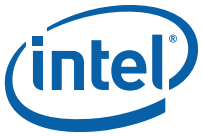
4.4 Output and I/O Signals Planes and States

Table 28 and Table 29 shows the power plane associated with the output and I/O signals, as well as the state at various times. Within the table, the following terms are used:

“High-Z”	Tri-state. The Intel® 6300ESB ICH is not driving the signal high or low.
“High”	The Intel® 6300ESB ICH is driving the signal to a logic ‘1’.
“Low”	The Intel® 6300ESB ICH is driving the signal to a logic ‘0’.
“Defined”	Driven to a level that is defined by the function (will be high or low).
“Undefined”	The Intel® 6300ESB ICH is driving the signal, but the value is indeterminate.
“Running”	Clock is toggling or signal is transitioning because the function is not stopping.

"Off"	The power plane is off, so the Intel® 6300ESB ICH is not driving.

Note: The signal levels are the same in S4 and S5.



4.5 Power Planes for Input Signals

Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 1 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
Hub Interface					
H1PD[7:0]	I/O	Core	Internal Termination	See table below	See table below
H1PSTRBS	I/O	Core	Internal Termination	See table below	See table below
H1PSTRBF	I/O	Core	Internal Termination	See table below	See table below
H1REQM	I	Core	Internal Termination	See table below	See table below
H1REQI	O	Core	Internal Termination	See table below	See table below
H1STOP	I/O	Core	Internal Termination	See table below	See table below
H1RCOMP	I/O	Core	Strap Pin External Resistor[1]	See table below	See table below
HLCLK	I	Core	None	See table below	See table below
H1PAR	I/O	Core	Internal Termination	See table below	See table below
VSWING	I	Core	External Resistor[1]	Input Only	Input Only
HLVREF	I	Core	External Resistor[1]	Input Only	Input Only
PCI-X Interface					
PXAD[63:32]	I/O	Core	External Pull-up	Z	Z
PXAD[31:0]	I/O	Core	None	Z	Driven (0 or 1)
PXC/BE[7:4]#	I/O	Core	External Pull-up	Z	Z
PXC/BE[3:0]#	I/O	Core	None	Z	Driven (0 or 1)
PXDEVSEL#	I/O	Core	External Pull-up	Z	Z
PXFRAME#	I/O	Core	External Pull-up	Z	Z
PXIRDY#	I/O	Core	External Pull-up	Z	Z
PXIRQ[3:0]#/ GPIO[36:33]	I / I/O **NOTE:** 9	Core	External Pull-up	Z	Z
PXTRDY#	I/O	Core	External Pull-up	Z	Z
PXSTOP#	I/O	Core	External Pull-up	Z	Z

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 2 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
PXPAR64	I/O	Core	External Pull-up	Z	Z
PXPAR	I/O	Core	None	Z	Driven (0 or 1)
PXPERR#	I/O	Core	External Pull-up	Z	Z
PXREQ[0]#	I	Core	External Pull-up	Input Only	Input Only
PXREQ[3:2]#/ GPIO[1:0]	I	Core	External Pull-up	Input Only	Input Only
PXGNT[0:1]#	O	Core	None	1	1
PXGNT[3]#/GPIO[17]	O	Core	None	1	1
PXGNT[2]#/GPIO[16]	O	Core	None	H	1
PCICLK	I	Core	None	Input Only	Input Only
RASERR#	OD	Core	External Pull-up	1	1
PXPCIRST#	O	Suspend	None	0	1
PXPLOCK#	I/O	Core	External Pull-up	Z	Z
PXSERR#	I/OD	Core	External Pull-up	Z	Z
PXREQ64#	I/O	Core	External Pull-up	H	H
PXACK64#	I/O	Core	External Pull-up	H	H
PXM66EN	I	Core	External 10K Pull Up	Input Only	Input Only
PXPCIXCAP	I	Core	External 10K Pull Up	Input Only	Input Only
PCI Interface					
AD[31:0]	I/O	Core	None	Z	Driven (0 or 1)
C/BE[3:0]#	I/O	Core	None	Z	Driven (0 or 1)
DEVSEL#	I/O	Core	External Pull-up	Z	Z
FRAME#	I/O	Core	External Pull-up	Z	Z
IRDY#	I/O	Core	External Pull-up	Z	Z
TRDY#	I/O	Core	External Pull-up	Z	Z
STOP#	I/O	Core	External Pull-up	Z	Z
PAR	I/O	Core	None	Z	Driven (0 or 1)
PERR#	I/O	Core	External Pull-up	Z	Z
REQ[3:0]#	I	Core	External Pull-up	Input Only	Input Only
GNT[3:0]#	O	Core	None	I	I
PCICLK	I/O	Core	None	Input Only	Input Only
PLOCK#	I/O	Core	External Pull-up	Z	Z
SERR#	I/OD	Core	External Pull-up	Z	Z

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

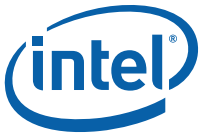


Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 3 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
PME#	I/OD	Suspend	Internal Pull-ups (15K - 35K)	H	H
LPC Interface					
LAD[3:0]	I/O	Core	Internal Pull-ups (15K - 35K)	H	H
LFRAME#	I	Core	Internal Pull-ups (15K - 35K)	1	1
LDRQ[0]#, LDRQ[1]#	I/O	Core	Internal Pull-ups (15K - 35K)	H	H
USB Interface					
USBP[3:0]P/N,	I/O	Suspend	Internal Pull-down and series resistors	L	L
OC[3:0]#	I	Suspend	None	Input Only	Input Only
USBRBIASP	O	Suspend	External Pull-down (22.6 ohm +/- 1%)	Z	1 during periodic Auto Current Cal
USBRBIASN	I	Suspend	Analog connection point for sensing the voltage across the external USBRBIASP resistor.	Input Only	Input Only
SATA Interface					
SATACLKP, SATACLKN	I	Core		Input Only	Input Only
SATA0TXP, SATA0TXN	O	Core		H	
SATA0RXP, SATA0RXN	I	Core		Input Only	Input Only
SATA1TXP, SATA1TXN	O	Core			
SATA1RXP, SATA1RXN	I	Core		Input Only	Input Only
SATALED#	OD	Core	External Pull-Up	Z	Z
SATARBIASP	I	Core			
SATARBIASN	I	Core			
IDE Interface					
PDCS1#, SDCS1#	O	Core	Internal Series resistors (21 ohms - 75 ohms)	1	1
PDCS3#, SDCS3#	O	Core	Internal Series resistors (21 ohms - 75 ohms)	1	1
PDA[2:0], SDA[2:0]	O	Core	Internal Series resistors (21 ohms - 75 ohms)	Driven (0 or 1)	Driven (0 or 1)

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 4 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
PDD[15:0], SDD[15:0]	I/O	Core	Internal Pull-downs on PDD[7] and SDD[7] (5.7K - 28.3K), Internal Series resistors (21 ohms - 75 ohms)	L on bit 7, Z on others	L on bit 7, Z on others
PDDREQ, SDDREQ	I	Core	Internal Pull-downs (5.7K - 28.3K), Internal Series resistors (21ohms - 75 ohms)	Input Only	Input Only
PDDACK#, SDDACK#	O	Core	Internal Series resistors (21 ohms - 75 ohms)	1	1
PDIOR# / (PDWSTB / PRDMARDY#) SDIOR# / (SDWSTB / SRDMARDY#)	O	Core	Internal Series resistors (21 ohms - 75 ohms)	1	1
PDIOW# / (PDSTOP) SDIOW# / (SDSTOP)	O	Core	Internal Series resistors (21 ohms - 75 ohms)	1	1
PIORDY / (PDRSTB / PWDMARDY#) SIORDY / (SDRSTB / SWDMARDY#)	I	Core	Internal Series resistors (21 ohms - 75 ohms) External Pull-ups	Input Only	Input Only
IRQ[14-15]	I	Core	Internal Series resistors (21 ohms - 75 ohms) External Pull-ups	Input Only	Input Only
Interrupt Pins					
SERIRQ	I/O	Core	External Pull-up	Z	Z
PIRQ[A:D]#	I	Core	External Pull-up	Z	Z
PIRQ[E:H]# / GPIO[2:5]	I/OD / I **NOTE:** 10	Core	External Pull-up when used as PIRQ	Z	Z
PXIRQ[3:0]#/ GPIO[36:33]	I / I/O **NOTE:** 9	Core	External Pull-up	Z	Z
AC'97 Interface					
AC_RST#	O	Suspend	Internal Pull-down (9K - 50K) when ACLink is shut	0	0
AC_SYNC	O	Core	Strap Pin Internal Pull-down[2] during reset (9K - 50K)	L	0
AC_BIT_CLK	I	Core	Internal Pull-down[2] (10K - 40K)	Input Only	Input Only

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 5 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
AC_SDATA_OUT	O	Core	Strap Pin Internal Pull-down[2] (9K - 50K)	L	0
AC_SDATA_IN[2:0]	I/O	Suspend	Internal Pull-down (9K - 50K)	L	
Power Management Pins					
THRM#	I	Core	None	Input Only	Input Only
THRMTRIP#	I	CPU I/O	None	Input Only	Input Only
SLP_S3#	O	Suspend	None	0	1
SLP_S4#	O	Suspend	None	0	1
SLP_S5#	O	Suspend	None	0	1
SYS_RESET#	I	Suspend	None	Input Only	Input Only
PWROK	I	RTC	None	Input Only	Input Only
PWRBTN#	I	Suspend	Internal Pull-Up (15K - 35K)	H	H
RI#	I	Suspend	None	Input Only	Input Only
RSMRST#	I	RTC	None	Input Only	Input Only
SUS_STAT#	O	Suspend	None	0	1 after PWROK rises
SUSCLK	O	Suspend	None	0	Toggling
VRMPWRGD	I	Core	None	Input Only	Input Only
CPU Interface Pins					
A20M#	O	CPU I/O	None	0 (before PWROK rising)	1
CPUSLP#	O	CPU I/O	None	1	1
FERR#	I	CPU I/O	External Pull-up	Input Only	Input Only
IGNNE#	O	CPU I/O	None	0 (before PWROK rising)	1
INIT#	O	CPU I/O	None	1	1
INTR	O	CPU I/O	None	0 (before PWROK rising)	0
NMI	O	CPU I/O	None	0 (before PWROK rising)	0
SMI#	O	CPU I/O	None	1	1
STPCLK#	O	CPU I/O	None	1	1

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 6 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
RCIN#	I	Core	Dependent upon the driving agent. External Pull-up when open-drain.	Input Only	Input Only
A20GATE	I	Core	Dependent upon the driving agent. External Pull-up when open-drain.	Input Only	Input Only
SMBus and System Management Pins					
SMBDATA	I/OD	Suspend	External Pull-up	Z	Z
SMBCLK	I/OD	Suspend	External Pull-up	Z	Z
SMBALERT#/ GPIO[11]	I	Suspend	External Pull-up (for SMBALERT#)	Input Only	Input Only
INTRUDER#	I	RTC	None	Input Only	Input Only
SMLINK[1:0]	I/OD	Suspend	External Pull-ups	Z	Z
Real Time Clock Pins					
RTCX1	Special	RTC	None	Analog Input	Analog Input
RTCX2	Special	RTC	None	Analog Input	Analog Input
Vbias	I	RTC	None	Analog Input	Analog Input
Miscellaneous Pins and GPIO					
CLK14	I	Core	None	Input Only	Input Only
CLK48	I	Core	None	Input Only	Input Only
SPKR	O	Core	Strap Pin Internal Pull-down (9K - 50K)	L	0
RTCRST#	I	RTC	External RC Circuit	Input Only	Input Only
WDT_TOUT#/ GPIO[32]	O / I/O	Core	None	1	1
GPIO[13:12],[8]	I	Suspend	None	Input Only	Input Only
GPIO[7:6]	I	Core	Internal Pull-Up	1	Input Only
GPIO[20:18]	O	Core	None	1	1 until driven
GPIO[21]	O	Core	None	0	1
GPIO[23]	O	Core	None	0	0 until driven
GPIO[25:24]	I/O	Suspend	None	1	1
GPIO[27:28]	I/O	Suspend	None	1	1

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

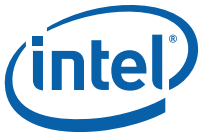


Table 28. Power Plane and States for Output and I/O Signal for Desktop Configurations (Sheet 7 of 7)

Signal Name	Type	Power Well	Resistors	During Reset (Note 2)	After Reset (Note 2)
GPIO[43:37]	I/O	Core	None	0	0 until driven
GPIO[57:56]	OD	RTC/ Suspend	External pull up	Driven (0 or 1)	Driven (0 or 1)
SIU Interface					
UART_CLK	I	Core	None	Input Only	Input Only
SIU0_RXD	I	Core	None	Input Only	Input Only
SIU1_RXD	I/O	Core	Internal Pull-Up (15K - 35K)	H	H
SIU0_TXD	I/O	Core	None	1	1
SIU1_TXD	I/O	Core	Internal Pull-Up (15K - 35K)	H	1
SIU0_CTS#	I	Core	None	Input Only	Input Only
SIU1_CTS#	I/O	Core	Internal Pull-Up (15K - 35K)	H	H
SIU0_DSR#	I	Core	None	Input Only	Input Only
SIU1_DSR#	I/O	Core	Internal Pull-Up (15K - 35K)	H	H
SIU0_DCD#	I	Core	None	Input Only	Input Only
SIU1_DCD#	I/O	Core	Internal Pull-Up (15K - 35K)	H	H
SIU0_RI#	I	Core	None	Input Only	Input Only
SIU1_RI#	I/O	Core	Internal Pull-Up (15K - 35K)	H	H
SIU0_DTR#	O	Core	Internal Pull-Up (15K - 35K)	1	1
SIU1_DTR#	I/O	Core	Internal Pull-Up (15K - 35K)	H	1
SIU0_RTS#	O	Core	None	1	1
SIU1_RTS#	I/O	Core	Internal Pull-Up (15K - 35K)	H	1

NOTES:
1. The Intel® 6300ESB ICH sets these signals at reset for CPU frequency strap.
2. The states of main I/O signals are taken at the times during PXPCIRST# and immediately after PXPCIRST#.
3. The states of resume I/O signals are taken at the times during RSMRST# and immediately after RSMRST#
4. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:55] are in the core well.
5. GPIO[8:15] and GPIO[24:31] are in the suspend well.
6. Core-well GPIO are 5 V tolerant, except for GPIO[7:6] and [32:43].
7. Resume-well GPIO are not 5 V tolerant.
8. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.
9. PXIRQ[3:0] are input only, GPIO[36:33] are I/O in GPIO mode
10. PIRQ[H:E] are I/OD, GPIO[5:2] are input only.

Table 29 shows the power plane associated with each input signal, as well as what device drives the signal at various times. Valid states include:

- High
- Low
- Static: Will be high or low, but will not change
- Driven: Will be high or low, and is allowed to change
- Running: For input clocks

Table 29. Power Plane for Input Signals for Desktop Configurations

Signal Name	Power Well	Driver During Reset	S1	S3	S5
A20GATE	Main I/O	External Microcontroller	Static	Low	Low
AC_BIT_CLK	Main I/O	AC'97 Codec	Low	Low	Low
AC_SDIN[2:0]	Resume I/O	AC'97 Codec	Low	Low	Low
APICCLK	Main I/O	Clock Generator	Running	Low	Low
CLK14	Main I/O	Clock Generator	Running	Low	Low
CLK48	Main I/O	Clock Generator	Running	Low	Low
CLK66	Main Logic	Clock Generator	Running	Low	Low
FERR#	CPU I/O	CPU	Static	Low	Low
INTRUDER#	RTC	External Switch	Driven	Driven	Driven
IRQ[15:14]	Main I/O	IDE	Static	Low	Low
LDRQ[0]#	Main I/O	LPC Devices	High	Low	Low
LDRQ[1]#	Main I/O	LPC Devices	High	Low	Low
OC[5:0]#	Resume I/O	External Pull-Ups	Driven	Driven	Driven
PCICLK	Main I/O	Clock Generator	Running	Low	Low
PDDREQ	Main I/O	IDE Device	Static	Low	Low
PIORDY	Main I/O	IDE Device	Static	Low	Low
PME#	Resume I/O	Internal Pull-Up	Driven	Driven	Driven
PWRBTN#	Resume I/O	Internal Pull-Up	Driven	Driven	Driven
PWROK	RTC	System Power Supply	Driven	Low	Low
RCIN#	Main I/O	External Microcontroller	High	Low	Low
REQ[0:5]#	Main I/O	PCI Master	Driven	Low	Low
REQ[B:A]#	Main I/O	PC/PCI Devices	Driven	Low	Low
RI#	Resume I/O	Serial Port Buffer	Driven	Driven	Driven
RSMRST#	RTC	External RC Circuit	High	High	High
RTCRST#	RTC	External RC Circuit	High	High	High
SDDREQ	Main I/O	IDE Drive	Static	Low	Low
SERR#	Main I/O	PCI Bus Peripherals	High	Low	Low
SIORDY	Main I/O	IDE Drive	Static	Low	Low
SMBALERT#	Resume I/O	External Pull-Up	Driven	Driven	Driven
SYS_RESET#	Resume I/O	External Circuit	Driven	Driven	Driven
THRM#	Main I/O	Thermal Sensor	Driven	Low	Low
THRMTRIP#	CPU I/O	Thermal Sensor	Driven	Low	Low
USBRBIAS#	Resume I/O	External Pull-Down	Driven	Driven	Driven

Functional Description 5

5.1 Hub Interface to PCI Bridge (D30:F0)

The Hub Interface to PCI Bridge resides in PCI Device 30, Function 0 on bus #0. This portion of the Intel® 6300ESB ICH implements the buffering and control logic between PCI and the Hub Interface. The arbitration for the PCI bus is handled by this PCI device. The PCI decoder in this device must decode the ranges for the Hub Interface. All register contents will be lost when core well power is removed.

5.1.1 PCI Bus Interface

The Intel® 6300ESB ICH PCI interface provides a 33 MHz, *PCI Local Bus Specification,* Rev. 2.2-compliant implementation. All PCI signals are 5 V tolerant. The Intel® 6300ESB ICH integrates a PCI arbiter that supports up to four external PCI bus masters in addition to the internal Intel® 6300ESB ICH requests.

Most transactions targeted to the Intel® 6300ESB ICH will first appear on the external PCI bus before being claimed back by the Intel® 6300ESB ICH. The exceptions are I/O cycles involving USB, IDE, and AC'97. These transactions will complete over the Hub Interface without appearing on the external PCI bus. Configuration cycles targeting USB, IDE or AC'97 will appear on the PCI bus. When the Intel® 6300ESB ICH is programmed for positive decode, the Intel® 6300ESB ICH will claim the cycles appearing on the external PCI bus in medium decode time. When the Intel® 6300ESB ICH is programmed for subtractive decode, the Intel® 6300ESB ICH will claim these cycles in subtractive time. When the Intel® 6300ESB ICH is programmed for subtractive decode, these cycles may be claimed by another positive decode agent out on PCI. This architecture enables the ability to boot off of a PCI card that positively decodes the boot cycles. In order to boot off a PCI card it is necessary to keep the Intel® 6300ESB ICH in subtractive decode mode. When booting off a PCI card, the BOOT_STS bit (bit 2, TCO2 Status Register) will be set.

When the processor issues a locked cycle to a resource that is too slow (e.g., PCI), the Intel® 6300ESB ICH will not allow upstream requests to be performed until the cycle completion. This may be critical for isochronous buses which assume certain timing for their data flow, such as AC'97 or USB. Devices on these buses may suffer from underrun when the asynchronous traffic is too heavy. Underrun means that the same data is sent over the bus while the Intel® 6300ESB ICH is not able to issue a request for the next data. Snoop cycles are not permitted while the processor side bus is locked.

Locked cycles are assumed to be rare. Locks by PCI targets are assumed to exist for a short duration (a few microseconds at most). When a system has a very large number of locked cycles and some that are very long, then the system will definitely experience underruns and overruns. The units most likely to have problems are the AC'97 controller and the USB controllers. Other units could get underruns/overruns, but are much less likely. The IDE controller (due to its stalling capability on the cable) should not get any underruns or overruns.

Note: The Intel® 6300ESB ICH's AC'97, IDE and USB Controllers cannot perform peer-to-peer traffic.

Note: Poor performing PCI devices that cause long latencies (numerous retries) to Processor-to-PCI Locked cycles may starve isochronous transfers between USB or AC'97 devices

and memory. This will result in overrun or underrun, causing reduced quality of the isochronous data, such as audio.

Note: PCI configuration write cycles, initiated by the processor, with the following characteristics will be converted to a Special Cycle with the Shutdown message type.

- Device Number (AD[15:11]) = '11111
- Function Number (AD[10:8]) = '111'
- Register Number (AD[7:2]) = '000000'
- Data = 00h
- Bus number matches secondary bus number

5.1.2 PCI-to-PCI Bridge Model

From a software perspective, the Intel® 6300ESB ICH contains a PCI-to-PCI bridge. This bridge connects the Hub Interface to the PCI bus. By using the PCI-to-PCI bridge software model, the Intel® 6300ESB ICH may have its decode ranges programmed by existing plug-and-play software such that PCI ranges do not conflict with AGP and graphics aperture ranges in the Host controller.

5.1.3 IDSEL to Device Number Mapping

When addressing devices on the external PCI bus (with the PCI slots) the Intel® 6300ESB ICH will assert one address signal as an IDSEL. When accessing device 0, the Intel® 6300ESB ICH will assert AD16. When accessing Device 1, the Intel® 6300ESB ICH will assert AD17. This mapping continues all the way up to device 15 where the Intel® 6300ESB ICH asserts AD31. Note that the Intel® 6300ESB ICH's internal functions (AC'97, IDE, USB, and PCI Bridge) are enumerated like they are on a separate PCI bus (the Hub Interface) from the external PCI bus.

5.1.4 SERR# Functionality

There are several internal and external sources that may cause SERR#. The Intel® 6300ESB ICH may be programmed to cause an NMI based on detecting that an SERR# condition has occurred. The NMI may also be routed to instead cause an SMI#.

Note: Note that the Intel® 6300ESB ICH does not drive the external PCI bus SERR# signal active onto the PCI bus. The external SERR# signal is an input into the Intel® 6300ESB ICH driven only by external PCI devices. The conceptual logic diagrams in Figure 6 and Figure 7 illustrate all sources of SERR#, along with their respective enable and status bits.

Figure 8 shows how the Intel® 6300ESB ICH error reporting logic is configured for NMI# generation.

Figure 6. Primary Device Status Register Error Reporting Logic

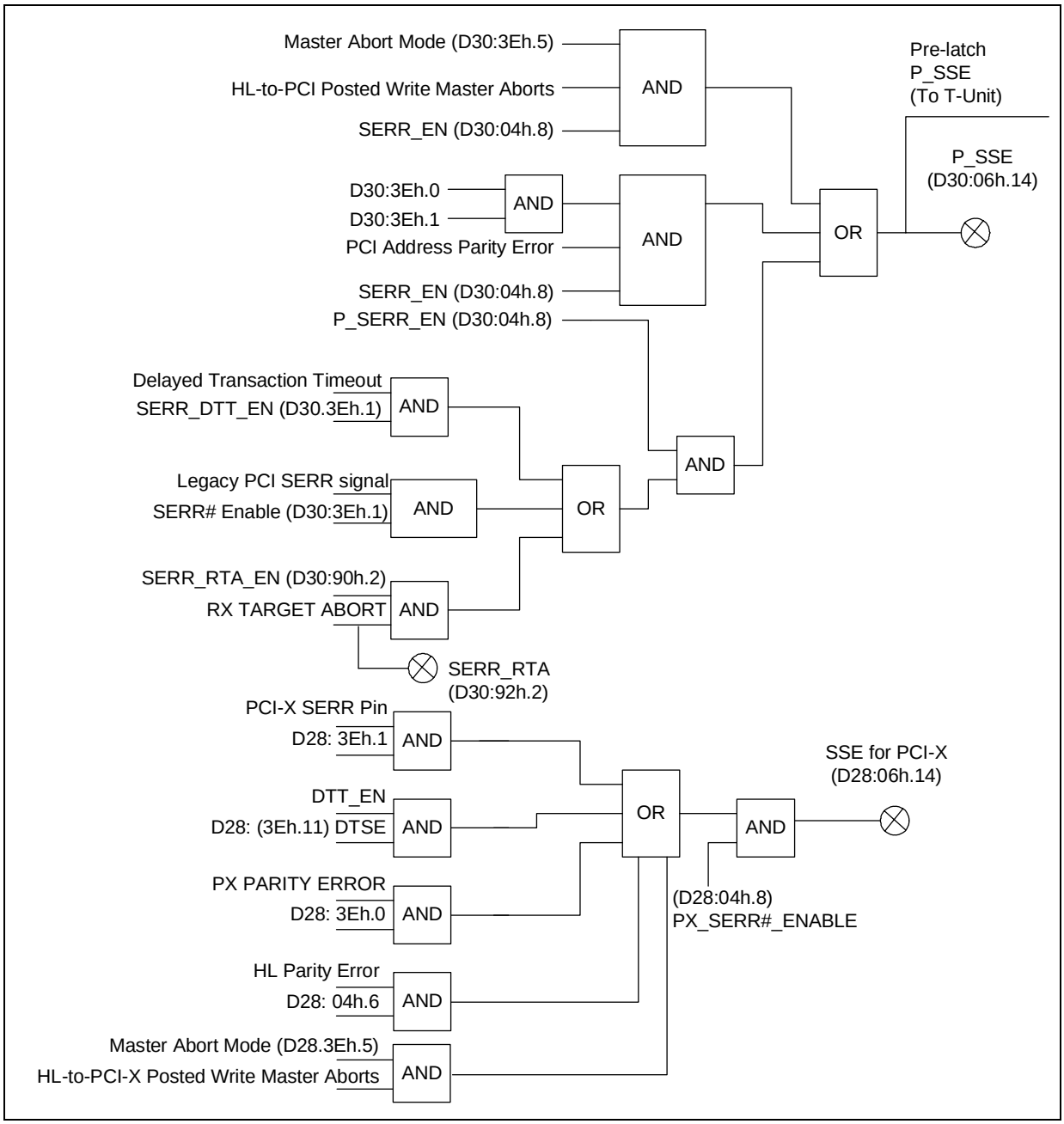


Figure 7. Secondary Status Register Error Reporting Logic

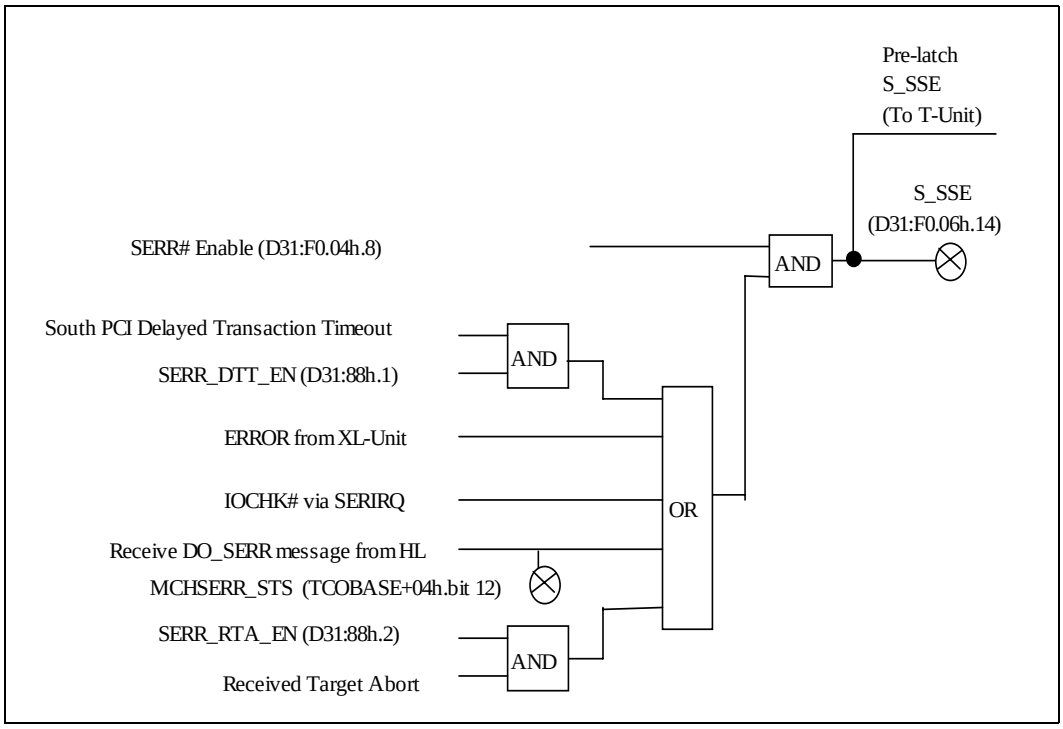
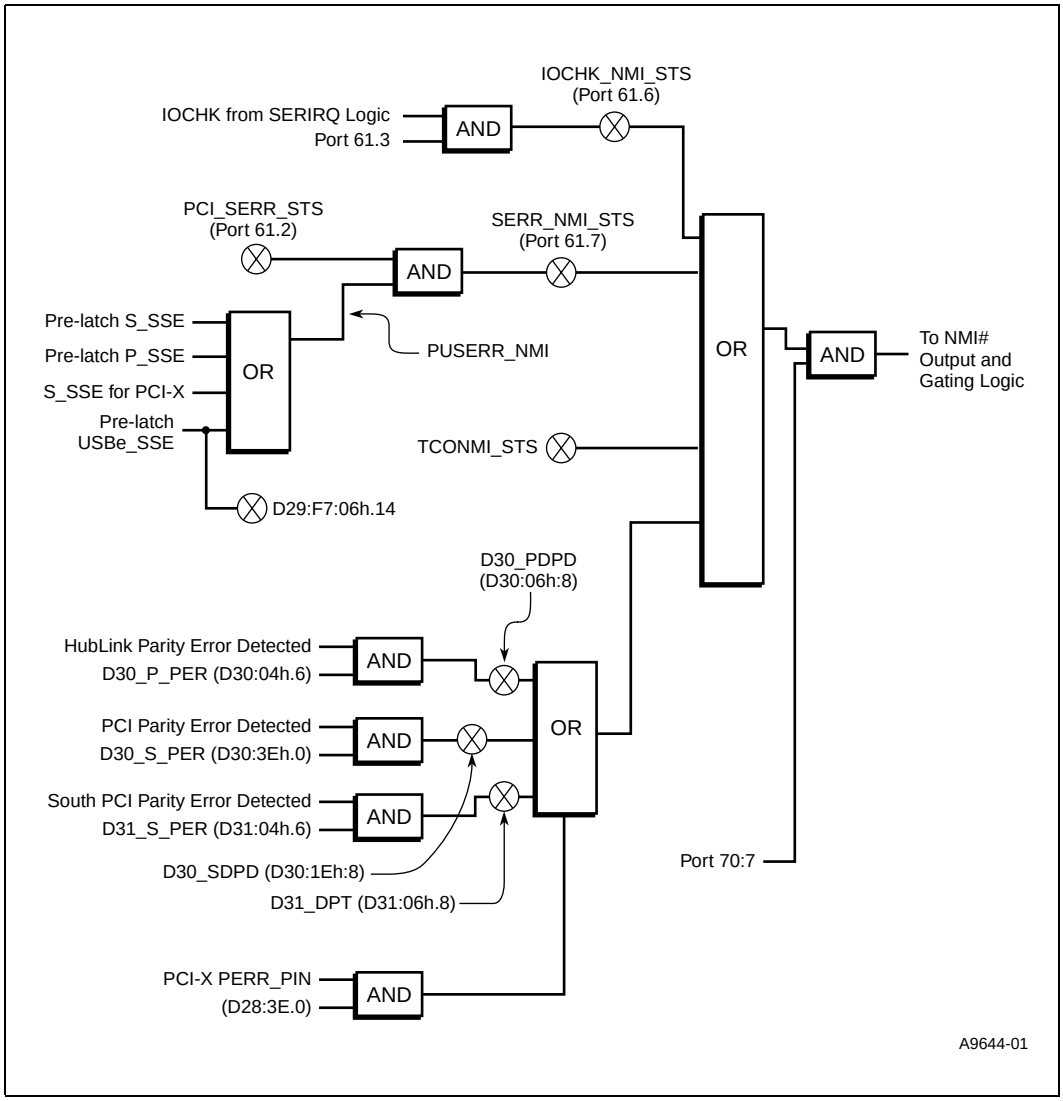


Figure 8. NMI# Generation Logic



5.1.5 Parity Error Detection

The Intel® 6300ESB ICH may detect and report different parity errors in the system. The Intel® 6300ESB ICH may be programmed to cause an NMI (or SMI# when NMI is routed to SMI#) based on detecting a parity error. The conceptual logic diagram in Figure 8 details all the parity errors that the Intel® 6300ESB ICH may detect, along with their respective enable bits, status bits, and the results.

Note: The Intel® 6300ESB ICH does not escalate a data parity mismatch reported by a PCI device (PERR#) across the P2P bridge.

Note: When NMIs are enabled, and parity error checking on PCI is also enabled, then parity errors will cause an NMI. Some operating systems will not attempt to recover from this NMI, since it considers the detection of a PCI error to be a catastrophic event.

5.1.6 Standard PCI Bus Configuration Mechanism

The PCI Bus defines a slot based “configuration space” that allows each device to contain up to eight functions with each function containing up to 256, 8-bit configuration registers. The PCI specification defines two bus cycles to access the PCI configuration space: Configuration Read and Configuration Write. Memory and I/O spaces are supported directly by the processor. Configuration space is supported by a mapping mechanism implemented within the Intel® 6300ESB ICH. The Intel® 6300ESB ICH only supports Mechanism #1 as defined in the PCI specification.

Configuration cycles for PCI Bus #0 devices #2 through #31, and for PCI Bus numbers greater than 0 will be sent towards the Intel® 6300ESB ICH from the host controller. The Intel® 6300ESB ICH compares the non-zero Bus Number with the Secondary Bus Number and Subordinate Bus number registers of its P2P bridge to determine when the configuration cycle is meant for Primary PCI or a downstream PCI bus.

5.1.6.1 Type 0 to Type 0 Forwarding

When a Type 0 configuration cycle is received on Hub Interface to any function other than USB EHCI or AC’97, the Intel® 6300ESB ICH forwards these cycles to PCI and then reclaims them. The Intel® 6300ESB ICH uses address bits AD[15:13] to communicate the Intel® 6300ESB ICH device numbers in Type 0 configuration cycles. When the Type 0 cycle on Hub Interface specifies any device number other than 29, 30 or 31, the Intel® 6300ESB ICH will not set any address bits in the range AD[31:11] during the corresponding transaction on PCI. Table 30 shows the device number translation.

Table 30. Type 0 Configuration Cycle Device Number Translation

Device # In Hub Interface Type 0 Cycle	AD[31:11] During Address Phase of Type 0 Cycle on PCI
0 through 28	0000000000000000_00000b
29	0000000000000000_00100b
30	0000000000000000_01000b
31	0000000000000000_10000b

The Intel® 6300ESB ICH logic will generate single D-word configuration read and write cycles on the PCI bus. The Intel® 6300ESB ICH will generate a Type 0 configuration cycle for configurations to the bus number matching the PCI bus. When the cycle is targeting a device behind an external bridge, the Intel® 6300ESB ICH will run a Type 1 cycle on the PCI bus.

5.1.6.2 Type 1 to Type 0 Conversion

When the bus number for the Type 1 configuration cycle matches the PCI (Secondary) bus number, the Intel® 6300ESB ICH will convert the address as follows:

1. For device numbers 0 through 15, only one bit of the PCI address [31:16] will be set. When the device number is 0, AD[16] is set; when the device number is 1, AD[17] is set; etc.
2. The Intel® 6300ESB ICH will always drive 0s on bits AD[15:11] when converting Type 1 configurations cycles to Type 0 configuration cycles on PCI.
3. Address bits [10:1] will also be passed unchanged to PCI.
4. Address bit [0] will be changed to ‘0’.

5.1.7 PCI Dual Address Cycle (DAC) Support

The Intel® 6300ESB ICH supports DAC format on PCI for cycles from PCI initiators to main memory. This allows PCI masters to generate an address up to 44 bits. The size of the actual supported memory space will be determined by the Memory Controller and the processor.

The DAC mode is only supported for PCI adapters and USB EHCI, and is not supported for any of the internal PCI masters (IDE, USB UHCI, AC'97, 8237 DMA, etc.).

When a PCI master wants to initiate a cycle with an address above 4G, it follows the following behavioral rules (See *PCI Local Bus Specification,* Revision 2.2, section 3.9 for more details):

1. On the first clock of the cycle (when FRAME# is first active), the peripheral uses the DAC encoding on the C/BE# signals. This unique encoding is: 1101.
2. Also during the first clock, the peripheral drives the AD[31:0] signals with the low address.
3. On the second clock, the peripheral drives AD[31:0] with the high address. The address is right justified: A[43:32] appear on AD[12:0]. The value of AD[31:13] is expected to be 0, however the Intel® 6300ESB ICH will ignore these bits. C/BE# indicate the bus command type (Memory Read, Memory Write, etc.)
4. The rest of the cycle proceeds normally.

5.2 LPC Bridge (with System and Management Functions) (D31:F0)

The LPC Bridge function of the Intel® 6300ESB ICH resides in PCI Device 31:Function 0. In addition to the LPC bridge function, D31:F0 contains other functional units including DMA, Interrupt Controllers, Timers, Power Management, System Management, GPIO, and RTC.

5.2.1 LPC Cycle Types

The Intel® 6300ESB ICH implements all of the cycle types described in the *Low Pin Count Interface Specification,* Revision 1.0. Table 31 shows the cycle types supported by the Intel® 6300ESB ICH.

Table 31. LPC Cycle Types Supported

Cycle Type	Comment
Memory Read	Single: 1 byte only[1]
Memory Write	Single: 1 byte only[1]
I/O Read	1 byte only. The Intel® 6300ESB ICH breaks up 16 and 32-bit processor cycles into multiple 8-bit transfers.
I/O Write	1 byte only. The Intel® 6300ESB ICH breaks up 16 and 32-bit processor cycles into multiple 8-bit transfers.
DMA Read	May be 1, or 2 bytes
DMA Write	May be 1, or 2 bytes
Bus Master Read	May be 1, 2, or 4 bytes.[2]
Bus Master Write	May be 1, 2, or 4 bytes.[2]

NOTES:
1. For memory cycles below 16M which do not target enabled FWH ranges, the Intel® 6300ESB ICH will perform standard LPC memory cycles. It will only attempt 8-bit transfers. When the cycle appears on PCI as a 16-bit transfer, it will appear as two consecutive 8-bit transfers on LPC. Likewise, when the cycle appears as a 32-bit transfer on PCI, it will appear as four consecutive 8-bit transfers on LPC. When the cycle is not claimed by any peripheral, it will be subsequently aborted, and the Intel® 6300ESB ICH will return a value of all 1s to the processor. This is done to maintain compatibility with ISA memory cycles where pull-up resistors would keep the bus high when no device responds.
2. Bus Master Read or Write cycles must be naturally aligned. For example, a 1-byte transfer may be to any address. However, the 2-byte transfer must be word aligned (i.e.,with an address where A0=0). A DWORD transfer must be DWORD aligned (i.e., with an address where A1and A0 are both 0).

5.2.1.1 Start Field Definition

Table 32. Start Field Bit Definitions

Bits[3:0] Encoding	Definition
0000	Start of cycle for a generic target.
0010	Grant for bus master 0.
0011	Grant for bus master 1.
1101	Start of cycle for firmware memory read cycle
1110	Start of cycle for firmware memory write cycle
1111	Stop/Abort: End of a cycle for a target.

NOTE: All other encodings are Reserved.

5.2.1.2 Cycle Type/Direction (CYCTYPE + DIR)

The Intel® 6300ESB ICH will always drive bit 0 of this field to zero. Peripherals running bus master cycles must also drive bit 0 to 0. The following table shows the valid bit encodings:

Table 33. Cycle Type Bit Definitions

Bits[3:2]	Bit[1]	Definition
00	0	I/O Read
00	1	I/O Write
01	0	Memory Read
01	1	Memory Write
10	0	DMA Read
10	1	DMA Write
11	x	Reserved. When a peripheral performing a bus master cycle generates this value, the Intel® 6300ESB ICH will abort the cycle.

5.2.1.3 SIZE

Bits[3:2] are reserved. The Intel® 6300ESB ICH will always drive them to 00. Peripherals running bus master cycles are also supposed to drive 00 for bits 3:2, however, the Intel® 6300ESB ICH will ignore those bits. Bits[1:0] are encoded as follows:

Table 34. Transfer Size Bit Definition

Bits[1:0]	Size
00	8-bit transfer (1 byte)
01	16-bit transfer (2 bytes)
10	Reserved. The Intel® 6300ESB ICH will never drive this combination. When a peripheral running a bus master cycle drives this combination, the Intel® 6300ESB ICH may abort the transfer.
11	32-bit transfer (4 bytes)

5.2.1.4 SYNC

Valid values for the SYNC field are:

Table 35. SYNC Bit Definition (Sheet 1 of 2)

Bits[3:0]	Indication
0000	**Ready:** SYNC achieved with no error. For DMA transfers, this also indicates DMA request deassertion and no more transfers desired for that channel.
0101	**Short Wait:** Part indicating wait-states. For bus master cycles, the Intel® 6300ESB ICH will not use this encoding. It will instead use the Long Wait encoding (see next encoding below).

NOTE: All other combinations are Reserved.

Table 35. SYNC Bit Definition (Sheet 2 of 2)

Bits[3:0]	Indication
0110	**Long Wait:** Part indicating wait-states, and many wait-states will be added. This encoding driven by the Intel® 6300ESB ICH for bus master cycles, rather than the Short Wait (0101).
1001	**Ready More (Used only by peripheral for DMA cycle):** SYNC achieved with no error and more DMA transfers desired to continue after this transfer. This value is valid only on DMA transfers and is not allowed for any other type of cycle.
1010	**Error:** Sync achieved with error. This is generally used to replace the SERR# or IOCHK# signal on the PCI/ISA bus. It indicates that the data is to be transferred, but there is a serious error in this transfer. For DMA transfers, this not only indicates an error, but also indicates DMA request deassertion and no more transfers desired for that channel.

NOTE: All other combinations are Reserved.

5.2.1.5 SYNC Time-Out

There are several error cases that may occur on the LPC I/F. The following table indicates the failing case and the Intel® 6300ESB ICH response:

Table 36. Response to Sync Failures

Possible Sync Failure	Intel® 6300ESB ICH Response
Intel® 6300ESB ICH starts a Memory, I/O, or DMA cycle, but no device drives a valid SYNC after 4 consecutive clocks. This could occur when the processor tries to access an I/O location to which no device is mapped.	Intel® 6300ESB ICH aborts the cycle after the fourth clock.
Intel® 6300ESB ICH drives a Memory, I/O, or DMA cycle, and a peripheral drives more than 8 consecutive valid SYNC to insert wait-states using the Short (‘0101b’) encoding for SYNC. This could occur when the peripheral is not operating properly.	Continues waiting
Intel® 6300ESB ICH starts a Memory, I/O, or DMA cycle, and a peripheral drives an invalid SYNC pattern. This could occur when the peripheral is not operating properly or when there is excessive noise on the LPC I/F.	Intel® 6300ESB ICH aborts the cycle when the invalid Sync is recognized.

There may be other peripheral failure conditions, however these are not handled by the Intel® 6300ESB ICH.

5.2.1.6 SYNC Error Indication

The SYNC protocol allows the peripheral to report an error through the LAD[3:0] = ‘1010b’ encoding. The intent of this encoding is to give peripherals a method of communicating errors to aid higher layers with more robust error recovery.

When the Intel® 6300ESB ICH is reading data from a peripheral, data will still be transferred in the next two nibbles. This data may be invalid, but it must be transferred by the peripheral. When the Intel® 6300ESB ICH is writing data to the peripheral, the data had already been transferred.

In the case of multiple byte cycles, such as for memory and DMA cycles, an error SYNC terminates the cycle. Therefore, when the Intel® 6300ESB ICH is transferring four bytes from a device, and if the device returns the error SYNC in the first byte, the other three bytes will not be transferred.

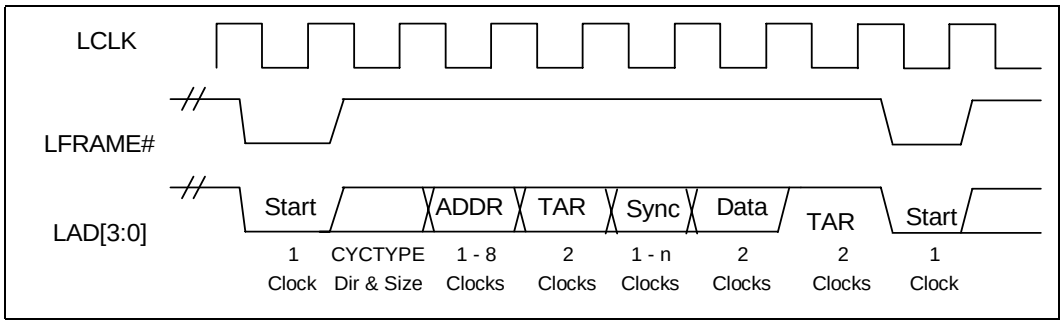
Upon recognizing the SYNC field indicating an error, the Intel® 6300ESB ICH will treat this the same as IOCHK# going active on the ISA bus.

5.2.1.7 LFRAME# Usage

Start of Cycle

For Memory, I/O, and DMA cycles, the Intel® 6300ESB ICH will assert LFRAME# for one clock at the beginning of the cycle (Figure 9) During that clock, the Intel® 6300ESB ICH will drive LAD[3:0] with the proper START field.

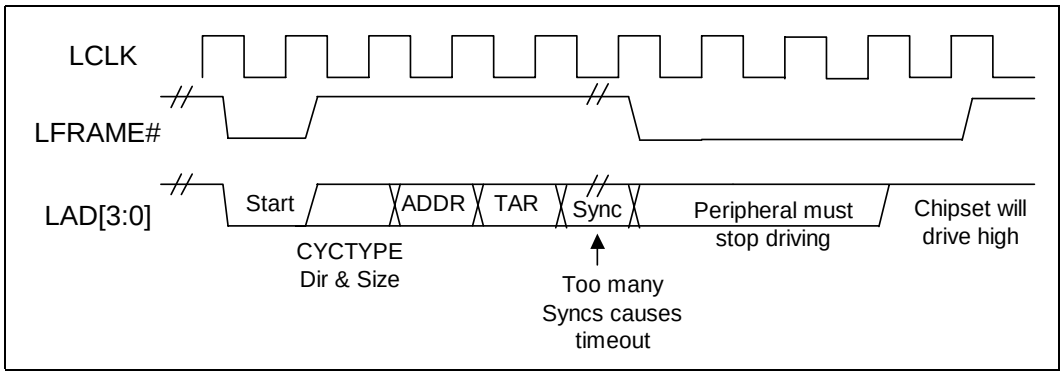
Figure 9. Typical Timing for LFRAME#



Abort Mechanism

When performing an Abort, the Intel® 6300ESB ICH will drive LFRAME# active for four consecutive clocks. On the fourth clock, it will drive LAD[3:0] to '1111b'.

Figure 10. Abort Mechanism



The Intel® 6300ESB ICH will perform an abort for the following cases (possible failure cases):

- Intel® 6300ESB ICH starts a Memory, I/O, or DMA cycle, but no device drives a valid SYNC after four consecutive clocks.
- Intel® 6300ESB ICH starts a Memory, I/O, or DMA cycle, and the peripheral drives an invalid SYNC pattern.
- A peripheral drives an illegal address when performing bus master cycles.
- A peripheral drives an invalid value.

5.2.1.8 I/O Cycles

For I/O cycles targeting registers specified in the Intel® 6300ESB ICH's decode ranges, the Intel® 6300ESB ICH performs I/O cycles as defined in the LPC spec. These will be 8-bit transfers. When the processor attempts a 16-bit or 32-bit transfer, the Intel® 6300ESB ICH will break the cycle up into multiple 8-bit transfers to consecutive I/O addresses.

Note: When the cycle is not claimed by any peripheral (and subsequently aborted), the Intel® 6300ESB ICH will return a value of all ones (FFh) to the processor. This is to maintain compatibility with ISA I/O cycles where pull-up resistors would keep the bus high when no device responds.

5.2.1.9 Bus Master Cycles

The Intel® 6300ESB ICH supports Bus Master cycles and requests (using LDRQ#) as defined in the LPC specification. The Intel® 6300ESB ICH has two LDRQ# inputs, and thus supports two separate bus master devices. It uses the associated START fields for Bus Master 0 ('0010b') or Bus Master 1 ('0011b').

Note: The Intel® 6300ESB ICH does not support LPC Bus Masters performing I/O cycles. LPC Bus Masters should only perform memory read or memory write cycles.

5.2.1.10 LPC Power Management

LPCPD# Protocol

Same timings as for SUS_STAT#. Upon driving SUS_STAT# low, LPC peripherals will drive LDRQ# low or tri-state it. The Intel® 6300ESB ICH will shut off the LDRQ# input buffers. After driving SUS_STAT# active, the Intel® 6300ESB ICH drives LFRAME# low, and tri-states (or drive low) LAD[3:0].

The Intel® 6300ESB ICH does not follow one part of the LPC spec that says "LRESET# is always asserted after LPCPD#". The exception is the S1-M state. In that case, LPCPD# (SUSSTAT#) will go active, but LRESET# (PXPCIRST#) will not go active.

5.2.1.11 Configuration and Intel® 6300ESB ICH Implications

LPC I/F Decoders

In order to allow the I/O cycles and memory mapped cycles to go to the LPC I/F, the Intel® 6300ESB ICH includes several decoders. During configuration, the Intel® 6300ESB ICH must be programmed with the same decode ranges as the peripheral. The decoders are programmed through the Device 31:Function 0 configuration space.

Note: The Intel® 6300ESB ICH cannot accept PCI write cycles from PCI-to-PCI bridges or devices with similar characteristics (specifically those with a "Retry Read" feature which is enabled) to an LPC device when there is an outstanding LPC read cycle towards the same PCI device or bridge. These cycles are not part of normal system operation, but may be encountered as part of platform validation testing using custom test fixtures.

Bus Master Device Mapping and START Fields

Bus Masters must have a unique START field. In the case of the Intel® 6300ESB ICH, which supports two LPC bus masters, it will drive 0010 for the START field for grants to bus master #0 (requested through LDRQ[0]#) and 0011 for grants to bus master #1 (requested through LDRQ[1]#.). Thus no registers are needed to config the START fields for a particular bus master.

BIOS Mapping and START Fields

To reduce decoding logic in the FWH, the Intel® 6300ESB ICH will use a unique IDSEL field for each EPROM. To do this, the Intel® 6300ESB ICH has configuration registers to assign a particular BIOS range to a particular IDSEL field.

5.3 DMA Operation (D31:F0)

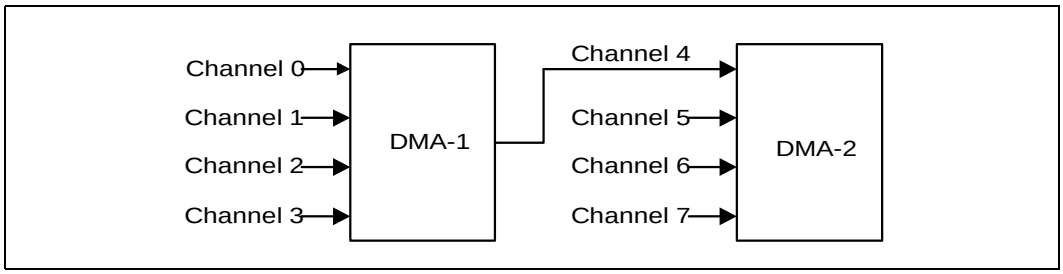
5.3.1 DMA Overview

The Intel® 6300ESB ICH supports LPC DMA through LPC, similar to ISA DMA.

The DMA controller has registers that are fixed in the lower 64 Kbyte of I/O space. The DMA controller is configured using registers in the PCI config space. These registers allow configuration of individual channels for use by LPC DMA.

The DMA circuitry incorporates the functionality of two 8237 DMA controllers with seven independently programmable channels (Figure 11). DMA Controller 1 (DMA-1) corresponds to DMA channels 0-3 and DMA Controller 2 (DMA-2) corresponds to channels 5-7. DMA channel 4 is used to cascade the two controllers and will default to cascade mode in the DMA Channel Mode (DCM) Register. Channel 4 is not available for any other purpose. In addition to accepting requests from DMA slaves, the DMA controller also responds to requests that software initiates. Software may initiate a DMA service request by setting any bit in the DMA Channel Request Register to 1.

Figure 11. Intel® 6300ESB ICH DMA Controller



Each DMA channel is hardwired to the compatible settings for DMA device size: channels [3:0] are hardwired to 8-bit, count-by-bytes transfers, and channels [7:5] are hardwired to 16-bit, count-by-words (address shifted) transfers.

The Intel® 6300ESB ICH provides 24-bit addressing in compliance with the ISA-compatible specification. Each channel includes a 16-bit ISA-Compatible Current Register which holds the 16 least-significant bits of the 24-bit address, an ISA-Compatible Page Register which contains the eight next most significant bits of address.

The DMA controller also features refresh address generation, and autoinitialization following a DMA termination.

5.3.2 Channel Priority

For priority resolution, the DMA consists of two logical channel groups: channels 0–3 and channels 4–7. Each group may be in either fixed or rotate mode, as determined by the DMA Command Register. See Section 8.2, “DMA I/O Registers” for more information.

DMA I/O slaves normally assert their DREQ line to arbitrate for DMA service. However, a software request for DMA service may be presented through each channel's DMA Request Register. A software request is subject to the same prioritization as any hardware request. Please see Section 8.4.9, “OCW3—Operational Control Word 3 Register” for detailed register description for Request Register programming information in the DMA I/O Register (Section 8.2, “DMA I/O Registers”).

5.3.2.1 Fixed Priority

The initial fixed priority structure is as described in Table 37.

Table 37. Fixed Priority

High priority.....Low priority	
(0, 1, 2, 3)	(5, 6, 7)

The fixed priority ordering is zero through seven. In this scheme, channel 0 has the highest priority, and channel 7 has the lowest priority. Channels [3:0] of DMA-1 assume the priority position of channel 4 in DMA-2, thus taking priority over channels 5, 6, and 7.

5.3.2.2 Rotating Priority

Rotation allows for “fairness” in priority resolution. The priority chain rotates so that the last channel serviced is assigned the lowest priority in the channel group (0–3, 5–7).

Channels 0–3 rotate as a group of four. They are always placed between channel 5 and channel 7 in the priority list.

Channel 5–7 rotate as part of a group of four. That is, channels (5–7) form the first three positions in the rotation, while channel group (0–3) comprises the fourth position in the arbitration.

5.3.3 Address Compatibility Mode

Whenever the DMA is operating, the addresses do not increment or decrement through the High and Low Page Registers. Therefore, when a 24-bit address is 01FFFFh and increments, the next address will be 010000h, not 020000h. Similarly, when a 24-bit address is 020000h and decrements, the next address will be 02FFFFh, not 01FFFFh. However, when the DMA is operating in 16 bit mode, the addresses still do not increment or decrement through the High and Low Page Registers but the page boundary is now 128K. Therefore, if a 24 bit address is 01FFFEh and increments, the next address will be 000000h, not 010000h. Similarly, if a 24 bit address is 020000h and decrements, the next address will be 03FFFEh, not 02FFFEh. This is compatible with the 8237 and Page Register implementation used in the PC-AT. This mode is set after CPURST is valid.

5.3.4 Summary of DMA Transfer Sizes

Table 39 lists each of the DMA device transfer sizes. The column labeled “Current Byte/Word Count Register” indicates that the register contents represents either the number of bytes to transfer or the number of 16-bit words to transfer. The column labeled “Current Address Increment/Decrement” indicates the number added to or taken from the Current Address register after each DMA transfer cycle. The DMA Channel Mode Register determines when the Current Address Register will be incremented or decremented.

5.3.4.1 Address Shifting When Programmed for 16-Bit I/O Count by Words

Table 38. DMA Transfer Size

DMA Device Date Size And Word Count	Current Byte/Word Count Register	Current Address Increment/ Decrement
8-Bit I/O, Count By Bytes	Bytes	1
16-Bit I/O, Count By Words (Address Shifted)	Words	1

The Intel® 6300ESB ICH maintains compatibility with the implementation of the DMA in the PC-AT which used the 8237. The DMA shifts the addresses for transfers to/from a 16-bit device count-by-words. Note that the least significant bit of the Low Page Register is dropped in 16-bit shifted mode. When programming the Current Address Register (when the DMA channel is in this mode), the Current Address must be programmed to an even address with the address value shifted right by one bit. The address shifting is described in Table 39.

Table 39. Address Shifting in 16-bit I/O DMA Transfers

Output Address	8-Bit I/O Programmed Address (Ch 0–3)	16-Bit I/O Programmed Address (Ch 5–7) (Shifted)
A0 A[16:1] A[23:17]	A0 A[16:1] A[23:17]	0 A[15:0] A[23:17]

NOTE: The least significant bit of the Page Register is dropped in 16-bit shifted mode.

5.3.5 Autoinitialize

By programming a bit in the DMA Channel Mode Register, a channel may be set up as an autoinitialize channel. When a channel undergoes autoinitialization, the original values of the Current Page, Current Address and Current Byte/Word Count Registers are automatically restored from the Base Page, Address, and Byte/Word Count Registers of that channel following TC. The Base Registers are loaded simultaneously with the Current Registers by the microprocessor when the DMA channel is programmed and remain unchanged throughout the DMA service. The mask bit is not set when the channel is in autoinitialize. Following autoinitialize, the channel is ready to perform another DMA service, without processor intervention, as soon as a valid DREQ is detected.

5.3.6 Software Commands

There are three additional special software commands that the DMA controller may execute. The three software commands are:

1. Clear Byte Pointer Flip-Flop
2. Master Clear
3. Clear Mask Register

They do not depend on any specific bit pattern on the data bus.

5.3.6.1 Clear Byte Pointer Flip-Flop

This command is executed prior to writing or reading new address or word count information to/from the DMA controller. This initializes the flip-flop to a known state so that subsequent accesses to register contents by the microprocessor will address upper and lower bytes in the correct sequence.

When the Host processor is reading or writing DMA registers, two Byte Pointer flip-flops are used; one for channels 0–3 and one for channels 4–7. Both of these act independently. There are separate software commands for clearing each of them (0Ch for channels 0–3, 0D8h for channels 4–7).

5.3.6.2 DMA Master Clear

This software instruction has the same effect as the hardware reset. The Command, Status, Request, and Internal First/Last Flip-Flop Registers are cleared and the Mask Register is set. The DMA controller will enter the idle cycle.

There are two independent master clear commands; 0Dh which acts on channels 0–3, and 0DAh which acts on channels 4–7.

5.3.6.3 Clear Mask Register

This command clears the mask bits of all four channels, enabling them to accept DMA requests.
I/O port 00Eh is used for channels 0–3 and I/O port 0DCh is used for channels 4–7.

5.4 LPC DMA

DMA on LPC is handled through the use of the LDRQ# lines from peripherals and special encodings on LAD[3:0] from the host. Single, Demand, Verify, and Increment modes are supported on the LPC interface. Channels 0 – 3 are 8 bit channels. Channels 5 – 7 are 16-bit channels. Channel 4 is reserved as a generic bus master request.

5.4.1 Asserting DMA Requests

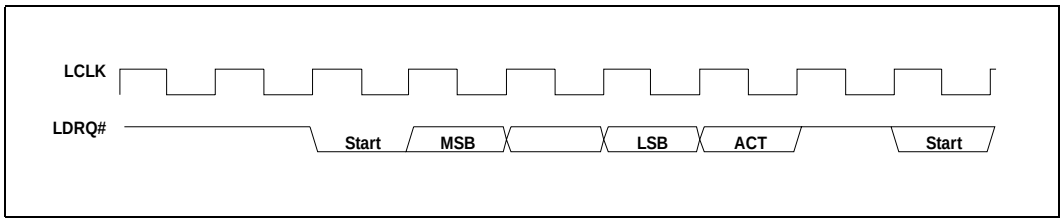
Peripherals that need DMA service encode their requested channel number on the LDRQ# signal. To simplify the protocol, each peripheral on the LPC I/F has its own dedicated LDRQ# signal (they may not be shared between two separate peripherals). The Intel® 6300ESB ICH has two LDRQ# inputs, allowing at least two devices to support DMA or bus mastering.

LDRQ# is synchronous with LCLK (PCI clock). As shown in Figure 12 the peripheral uses the following serial encoding sequence:

- Peripheral starts the sequence by asserting LDRQ# low (start bit). LDRQ# is high during idle conditions.
- The next 3 bits contain the encoded DMA channel number (MSB first).
- The next bit (ACT) indicates whether the request for the indicated DMA channel is active or inactive. The ACT bit will be a 1 (high) to indicate when it is active and 0 (low) when it is inactive. The case where ACT is low will be rare, and is only used to indicate that a previous request for that channel is being abandoned.
- After the active/inactive indication, the LDRQ# signal must go high for at least 1 clock. After that one clock, LDRQ# signal may be brought low to the next encoding sequence.

When another DMA channel also needs to request a transfer, another sequence may be sent on LDRQ#. For example, if an encoded request is sent for channel 2, and then channel 3 needs a transfer before the cycle for channel 2 is run on the interface, the peripheral may send the encoded request for channel 3. This allows multiple DMA agents behind an I/O device to request use of the LPC interface, and the I/O device does not need to self-arbitrate before sending the message.

Figure 12. DMA Request Assertion Through LDRQ#



5.4.2 Abandoning DMA Requests

DMA Requests may be deasserted in two fashions: on error conditions by sending an LDRQ# message with the ‘ACT’ bit set to ‘0’, or normally through a SYNC field during the DMA transfer. This section describes boundary conditions where the DMA request needs to be removed prior to a data transfer.

There may be some special cases where the peripheral desires to abandon a DMA transfer. The most likely case of this occurring is due to a floppy disk controller which has overrun or underrun its FIFO, or software stopping a device prematurely.

In these cases, the peripheral wishes to stop further DMA activity. It may do so by sending an LDRQ# message with the ACT bit as ‘0’. However, since the DMA request was seen by the Intel® 6300ESB ICH, there is no ensuring that the cycle has not been granted and will shortly run on LPC. Therefore, peripherals must take into account that a DMA cycle may still occur. The peripheral may choose not to respond to this cycle, in which case the host will abort it, or it may choose to complete the cycle normally with any random data.

This method of DMA deassertion should be prevented whenever possible, to limit boundary conditions both on the Intel® 6300ESB ICH and the peripheral.

5.4.3 General Flow of DMA Transfers

Arbitration for DMA channels is performed through the 8237 within the host. Once the host has won arbitration on behalf of a DMA channel assigned to LPC, it asserts LFRAME# on the LPC I/F and begins the DMA transfer. The general flow for a basic DMA transfer is as follows:

1. The Intel® 6300ESB ICH starts transfer by asserting '0000b' on LAD[3:0] with LFRAME# asserted.
2. The Intel® 6300ESB ICH asserts 'cycle type' of DMA, direction based on DMA transfer direction.
3. The Intel® 6300ESB ICH asserts channel number and, when applicable, terminal count.
4. The Intel® 6300ESB ICH indicates the size of the transfer: 8 or 16 bits.
5. When a DMA read...
   - The Intel® 6300ESB ICH drives the first 8 bits of data and turns the bus around.
   - The peripheral acknowledges the data with a valid SYNC.
   - When a 16 bit transfer, the process is repeated for the next 8 bits.
6. When a DMA write...
   - The Intel® 6300ESB ICH turns the bus around and waits for data.
   - The peripheral indicates data ready through SYNC and transfers the first byte.
   - When a 16 bit transfer, the peripheral indicates data ready and transfers the next byte.
7. The peripheral turns around the bus.

5.4.4 Terminal Count (TC)

Terminal count is communicated through LAD[3] on the same clock that DMA channel is communicated on LAD[2:0]. This field is the CHANNEL field. Terminal count indicates the last byte of transfer, based upon the size of the transfer.

For example, on an eight bit transfer size (SIZE field is '00b'), when the TC bit is set, this is the last byte. On a 16 bit transfer (SIZE field is '01b'), when the TC bit is set, the second byte is the last byte. The peripheral, therefore, must internalize the TC bit when the CHANNEL field is communicated, and only signal TC when the last byte of that transfer size has been transferred.

5.4.5 Verify Mode

Verify mode is supported on the LPC interface. A verify transfer to the peripheral is similar to a DMA write, where the peripheral is transferring data to main memory. The indication from the host is the same as a DMA write, so the peripheral will be driving data onto the LPC interface. However, the host will not transfer this data into main memory.

5.4.6 DMA Request Deassertion

An end of transfer is communicated to the Intel® 6300ESB ICH through a special SYNC field transmitted by the peripheral. An LPC device must not attempt to signal the end of a transfer by deasserting LDREQ#. When a DMA transfer is several bytes, such as a transfer from a demand mode device, the Intel® 6300ESB ICH needs to know when to deassert the DMA request based on the data currently being transferred.

The DMA agent uses a SYNC encoding on each byte of data being transferred, which indicates to the Intel® 6300ESB ICH whether this is the last byte of transfer or when more bytes are requested. To indicate the last byte of transfer, the peripheral uses a SYNC value of '0000b' (ready with no error), or '1010b' (ready with error). These encodings tell the Intel® 6300ESB ICH that this is the last piece of data transferred on a DMA read (Intel® 6300ESB ICH to peripheral), or the byte which follows is the last piece of data transferred on a DMA write (peripheral to the Intel® 6300ESB ICH).

When the Intel® 6300ESB ICH sees one of these two encodings, it ends the DMA transfer after this byte and deasserts the DMA request to the 8237. Therefore, when the Intel® 6300ESB ICH indicated a 16 bit transfer, the peripheral may end the transfer after one byte by indicating a SYNC value of '0000b' or '1010b'. The Intel® 6300ESB ICH will not attempt to transfer the second byte, and will deassert the DMA request internally. This also holds true for any byte in a 32 bit transfer. This allows the peripheral, therefore, to terminate a DMA burst.

When the peripheral indicates a '0000b' or '1010b' SYNC pattern on the last byte of the indicated size, then the Intel® 6300ESB ICH will only deassert the DMA request to the 8237 since it does not need to end the transfer.

When the peripheral wishes to keep the DMA request active, it uses a SYNC value of '1001b' (ready plus more data). This tells the 8237 that more data bytes are requested after the current byte has been transferred, so the Intel® 6300ESB ICH will keep the DMA request active to the 8237. Therefore, on an 8-bit transfer size, when the peripheral indicates a SYNC value of '1001b' to the Intel® 6300ESB ICH, the data will be transferred and the DMA request will remain active to the 8237. At a later time, the Intel® 6300ESB ICH will then come back with another START–CYCTYPE–CHANNEL–SIZE etc. combination to initiate another transfer to the peripheral.

The peripheral must not assume that the next START indication from the Intel® 6300ESB ICH is another grant to the peripheral when it had indicated a SYNC value of '1001b'. On a single mode DMA device, the 8237 will rearbitrate after every transfer. Only demand mode DMA devices may be ensured that they will receive the next START indication from the Intel® 6300ESB ICH.

Note: Indicating a '0000b' or '1010b' encoding on the SYNC field of an odd byte of a 16 bit channel (first byte of a 16 bit transfer) is an error condition.

Note: The host will stop the transfer on the LPC bus as indicated, fill the upper byte with random data on DMA writes (peripheral to memory), and indicate to the 8237 that the DMA transfer occurred, incrementing the 8237's address and decrementing its byte count.

5.4.7 SYNC Field / LDRQ# Rules

Since DMA transfers on LPC are requested through an LDRQ# assertion message, and are ended through a SYNC field during the DMA transfer, the peripheral must obey the following rule when initiating back-to-back transfers from a DMA channel.

The peripheral must not assert another message for eight LCLKs after a deassertion is indicated through the SYNC field. This is needed to allow the 8237, which typically runs off a much slower internal clock, to see a message deasserted before it is re-asserted so that it may arbitrate to the next agent.

Under default operation, the host will only perform 8-bit transfers on 8-bit channels and 16-bit transfers on 16 bit channels.

The method by which this communication between host and peripheral through system BIOS is performed is beyond the scope of this specification. Since the LPC host and LPC peripheral are motherboard devices, no "plug-n-play" registry is required.

The peripheral must not assume that the host will be able to perform transfer sizes that are larger than the size allowed for the DMA channel, and be willing to accept a SIZE field that is smaller than what it may currently have buffered.

To that end, it is recommended that future devices which may appear on the LPC bus, which require higher bandwidth than 8-bit or 16-bit DMA allow, do so with a bus mastering interface and not rely on the 8237.

5.5 8254 Timers (D31:F0)

The Intel® 6300ESB ICH contains three counters which have fixed uses. All registers and functions associated with the 8254 timers are in the core well. The 8254 unit is clocked by a 14.31818 MHz clock. The 14.31818 MHz clock will stop during the S3-S5 and G3 states.

5.5.1 Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is typically programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value one counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two each counter period. The counter negates IRQ0 when the count value reaches zero. It then reloads the initial count value and again decrements the initial count value by two each counter period. The counter then asserts IRQ0 when the count value reaches zero, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

5.5.2 Counter 1, Refresh Request Signal

Prior to ICH1, typically in ISA platforms, this counter provided the refresh request signal. Today, it is still typically programmed for Mode 2 operation and only impacts the period of the REF_TOGGLE bit in Port 61. The initial count value is loaded one counter period after being written to the counter I/O address. The REF_TOGGLE bit will have a square wave behavior (alternate between 0 and 1) and will toggle at a rate based on the value in the counter.

Programming the counter to anything other than Mode 2 will result in undefined behavior for the REF_TOGGLE bit. See Section 8.7.1, “NMI_SC—NMI Status and Control Register” (D31:F0:61h:bit 4) for REF_TOGGLE bit details.

5.5.3 Counter 2, Speaker Tone

This counter provides the speaker tone and is typically programmed for Mode 3 operation. The counter provides a speaker frequency equal to the counter clock frequency (1.193 MHz) divided by the initial count value. The speaker must be enabled by a write to port 061h (see Section 8.7.1, “NMI_SC—NMI Status and Control Register” for more information).

5.5.4 Timer Programming

The counter/timers are programmed in the following fashion:

1. Write a control word to select a counter.
2. Write an initial count for that counter.
3. Load the least and/or most significant bytes (as required by Control Word bits 5, 4) of the 16-bit counter. See Section 8.3, “Timer I/O Registers” for more information.
4. Repeat with other counters.

Only two conventions need to be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting will be affected as described in the mode definitions. The new count must follow the programmed count format.

When a counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of all three counters. Several commands are available:

- **Control Word Command.** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command.** Latches the current count so that it may be read by the system. The countdown process continues.
- **Read Back Command.** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

Table 40 lists the six operating modes for the interval counters.

Table 40. Counter Operating Modes

Mode	Function	Description
0	Out signal on end of count (=0)	Output is ‘0’. When count goes to 0, output goes to ‘1’ and stays at ‘1’ until counter is reprogrammed.
1	Hardware retriggerable one-shot	Output is ‘0’. When count goes to 0, output goes to ‘1’ for one clock period.
2	Rate generator (divide by n counter)	Output is ‘1’. Output goes to ‘0’ for one clock time, then back to ‘1’ and counter is reloaded.
3	Square wave output	Output is ‘1’. Output goes to ‘0’ when counter rolls over, and counter is reloaded. Output goes to ‘1’ when counter rolls over, and counter is reloaded, etc.
4	Software triggered strobe	Output is ‘1’. Output goes to ‘0’ when count expires for one clock period.
5	Hardware triggered strobe	Output is ‘1’. Output goes to ‘0’ when count expires for one clock period.

5.5.5 Reading from the Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters: a simple read operation, counter Latch command, and the Read-Back command. Each is explained below.

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, when the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

5.5.5.1 Simple Read

The first method is to perform a simple read operation. The counter is selected through port 40h (counter 0), 41h (counter 1), or 42h (counter 2).

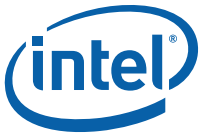
Note: Performing a direct read from the counter will not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of counter 2, the count may be stopped by writing to the GATE bit in port 61h.

5.5.5.2 Counter Latch Command

The Counter Latch Command, written to port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a two-byte count. The count value is then read from each counter's Count Register as was programmed by the Control Register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

When a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch command is ignored. The count read will be the count at the time the first Counter Latch command was issued.



5.5.5.3 Read Back Command

The Read Back command, written to port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. When multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. When multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. When multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

When both count and status of a counter are latched, the first read operation from that counter will return the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, return the latched count. Subsequent reads return unlatched count.

5.6 8259 Interrupt Controllers (PIC) (D31:F0)

The Intel® 6300ESB ICH incorporates the functionality of two 8259 interrupt controllers that provide system interrupts for the ISA compatible interrupts. These interrupts are: system timer, keyboard controller, serial ports, parallel ports, floppy disk, IDE, mouse, and DMA channels. In addition, this interrupt controller may support the PCI based interrupts, by mapping the PCI interrupt onto the compatible ISA interrupt line. Each 8259 core supports eight interrupts, numbered zero through seven. Table 41 shows how the cores are connected.

Table 41. Interrupt Controller Core Connections (Sheet 1 of 2)

8259	8259 Input	Typical Interrupt Source	Connected Pin / Function
Master	0	Internal	Internal Timer / Counter 0 output / MMT #0
	1	Keyboard	IRQ1 via SERIRQ
	2	Internal	Slave Controller INTR output
	3	Serial Port A	IRQ3 via SERIRQ, PIRQx
	4	Serial Port B	IRQ4 via SERIRQ, PIRQx
	5	Parallel Port / Generic	IRQ5 via SERIRQ, PIRQx
	6	Floppy Disk	IRQ6 via SERIRQ PIRQx,
	7	Parallel Port / Generic	IRQ7 via SERIRQ PIRQx,

Table 41. Interrupt Controller Core Connections (Sheet 2 of 2)

8259	8259 Input	Typical Interrupt Source	Connected Pin / Function
Slave	0	Internal Real Time Clock	Internal RTC / MMT #1
	1	Generic	IRQ9 via SERIRQ, SCI or TCO, PIRQx, Boot Interrupt
	2	Generic	IRQ10 via SERIRQ, SCI, or TCO, PIRQx
	3	Generic	IRQ11 via SERIRQ, SCI, or TCO, PIRQx, Multimedia Timer #2
	4	PS/2 Mouse	IRQ12 via SERIR, SCI, or TCO, PIRQx
	5	Internal	State Machine output based on processor FERR# assertion. See Section 5.8.4, "Specific Interrupts Not Supported via SERIRQ" for more information.
	6	Primary IDE cable	IRQ14 from input signal (primary IDE in legacy mode only) or via SERIRQ PIRQx
	7	Secondary IDE Cable	IRQ15 from input signal (secondary IDE in legacy mode only) or via SERIRQ, PIRQx

The Intel® 6300ESB ICH cascades the slave controller onto the master controller through master controller interrupt input two. This means there are only 15 possible interrupts for the Intel® 6300ESB ICH PIC.

Interrupts may individually be programmed to be edge or level, except for IRQ0, IRQ2 and IRQ8#.

Note that previous PIIXn devices internally latched IRQ12 and IRQ1 and required a port 60h read to clear the latch. The Intel® 6300ESB ICH may be programmed to latch IRQ12 or IRQ1 (see bit 11 and bit 12 in General Control Register, D31:F0, offset D0h).

5.6.1 Interrupt Handling

5.6.1.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 42 defines the IRR, ISR and IMR.

Table 42. Interrupt Status Registers

Bit	Description
IRR	**Interrupt Request Register.** This bit is set on a low to high transition of the interrupt line in edge mode, and by an active high level in level mode. This bit is set whether or not the interrupt is masked. However, a masked interrupt will not generate INTR.
ISR	**Interrupt Service Register.** This bit is set, and the corresponding IRR bit cleared, when an interrupt acknowledge cycle is seen, and the vector returned is for that interrupt.
IMR	**Interrupt Mask Register.** This bit determines whether an interrupt is masked. Masked interrupts will not generate INTR.

5.6.1.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle which is translated by the host bridge into a PCI Interrupt Acknowledge Cycle to the Intel® 6300ESB ICH. The PIC translates this command into two internal INTA# pulses expected by the 8259 cores. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave will send the interrupt vector to the processor with the acknowledged interrupt code. This code is based upon bits [7:3] of the corresponding ICW2 register, combined with three bits representing the interrupt within that controller.

Table 43. Content of Interrupt Vector Byte

Master, Slave Interrupt	Bits [7:3]	Bits [2:0]
IRQ7,15	ICW2[7:3]	111
IRQ6,14		110
IRQ5,13		101
IRQ4,12		100
IRQ3,11		011
IRQ2,10		010
IRQ1,9		001
IRQ0,8		000

5.6.1.3 Hardware/Software Interrupt Sequence

1. One or more of the Interrupt Request lines (IRQ) are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor when an asserted interrupt is not masked.
3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle. The cycle is translated into a PCI interrupt acknowledge cycle by the host bridge. This command is broadcast over PCI by the Intel® 6300ESB ICH.
4. Upon observing its own interrupt acknowledge cycle on PCI, the Intel® 6300ESB ICH converts it into the two cycles that the internal 8259 pair may respond to. Each cycle appears as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.
5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine when it must respond with an interrupt vector during the second INTA# pulse.
6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. When no interrupt request is present because the request was too short in duration, the PIC will return vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

5.6.2 Initialization Command Words (ICWx)

Before operation may begin, each 8259 must be initialized. In the Intel® 6300ESB ICH, this is a four byte sequence. The four initialization command words are referred to by their acronyms: ICW1, ICW2, ICW3, and ICW4.

The base address for each 8259 initialization command word is a fixed location in the I/O memory space: 20h for the master controller, and A0h for the slave controller.

5.6.2.1 ICW1

An I/O write to the master or slave controller base address with data bit 4 equal to 1 is interpreted as a write to ICW1. Upon sensing this write, the Intel® 6300ESB ICH PIC expects three more byte writes to 21h for the master controller, or A1h for the slave controller, to complete the ICW sequence.

A write to ICW1 starts the initialization sequence during which the following automatically occur:

1. Following initialization, an interrupt request (IRQ) input must make a low-to-high transition to generate an interrupt.
2. The Interrupt Mask Register is cleared.
3. IRQ7 input is assigned priority 7.
4. The slave mode address is set to 7.
5. Special mask mode is cleared and Status Read is set to IRR.

5.6.2.2 ICW2

The second write in the sequence, ICW2, is programmed to provide bits [7:3] of the interrupt vector that will be released during an interrupt acknowledge. A different base is selected for each interrupt controller.

5.6.2.3 ICW3

The third write in the sequence, ICW3, has a different meaning for each controller.

- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. Within the Intel® 6300ESB ICH, IRQ2 is used. Therefore, bit 2 of ICW3 on the master controller is set to a 1, and the other bits are set to 0s.
- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller when the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and when it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

5.6.2.4 ICW4

The final write in the sequence, ICW4, must be programmed both controllers. At the very least, bit 0 must be set to one to indicate that the controllers are operating in an Intel® Architecture-based system.

5.6.3 Operation Command Words (OCW)

These command words reprogram the Interrupt Controller to operate in various interrupt modes.

- OCW1 masks and unmasks interrupt lines.
- OCW2 controls the rotation of interrupt priorities when in rotating priority mode, and controls the EOI function.

- OCW3 is sets up ISR/IRR reads, enables/disables the special mask mode (SMM), and enables/disables polled interrupt mode.

5.6.4 Modes of Operation

5.6.4.1 Fully Nested Mode

In this mode, interrupt requests are ordered in priority from zero through seven, with zero being the highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the ISR for the interrupt is set. This ISR bit remains set until: the processor issues an EOI command immediately before returning from the service routine; or when in AEOI mode, on the trailing edge of the second INTA#. While the ISR bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels will generate another interrupt.

Interrupt priorities may be changed in the rotating priority mode.

5.6.4.2 Special Fully-Nested Mode

This mode will be used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully-nested mode will be programmed to the master controller. This mode is similar to the fully-nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave will be recognized by the master and will initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.
- When exiting the Interrupt Service routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a Non-Specific EOI command to the slave and then reading its ISR. When it is zero, a non-specific EOI may also be sent to the master.

5.6.4.3 Automatic Rotation Mode (Equal Priority Devices)

In some applications, there are a number of interrupting devices of equal priority. Automatic rotation mode provides for a sequential 8-way rotation. In this mode, a device receives the lowest priority after being serviced. In the worst case, a device requesting an interrupt will have to wait until each of seven other devices are serviced at most once.

There are two ways to accomplish automatic rotation using OCW2; the Rotation on Non-Specific EOI Command (R=1, SL=0, EOI=1) and the rotate in automatic EOI mode which is set by (R=1, SL=0, EOI=0).

5.6.4.4 Specific Rotation Mode (Specific Priority)

Software may change interrupt priorities by programming the bottom priority. For example, when IRQ5 is programmed as the bottom priority device, IRQ6 will be the highest priority device. The Set Priority Command is issued in OCW2 to accomplish this, where: R=1, SL=1, and LO-L2 is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes may be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (R=1, SL=1, EOI=1 and LO-L2=IRQ level to receive bottom priority.

5.6.4.5 Poll Mode

Poll mode may be used to conserve space in the interrupt vector table. Multiple interrupts that may be serviced by one interrupt service routine do not need separate vectors when the service routine uses the poll command. Poll mode may also be used to expand the number of interrupts. The polling interrupt service routine may call the appropriate service routine, instead of providing the interrupt vectors in the vector table. In this mode, the INTR output is not used and the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting P=1 in OCW3. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit when there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read will contain a '1' in bit 7 when there is an interrupt, and the binary code of the highest priority level in bits 2:0.

5.6.4.6 Cascade Mode

The PIC in the Intel® 6300ESB ICH has one master 8259 and one slave 8259 cascaded onto the master through IRQ2. This configuration may handle up to 15 separate priority levels. The master controls the slaves through a three bit internal bus. In the Intel® 6300ESB ICH, when the master drives 010b on this bus, the slave controller takes responsibility for returning the interrupt vector. An EOI command must be issued twice: once for the master and once for the slave.

5.6.4.7 Edge and Level Triggered Mode

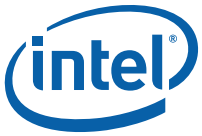
In ISA systems this mode is programmed using bit 3 in ICW1, which sets level or edge for the entire controller. In the Intel® 6300ESB ICH, this bit is disabled and a new register for edge and level triggered mode selection, per interrupt input, is included. This is the Edge/Level control Registers ELCR1 and ELCR2.

When an ELCR bit is '0', an interrupt request will be recognized by a low to high transition on the corresponding IRQ input. The IRQ input may remain high without generating another interrupt. When an ELCR bit is '1', an interrupt request will be recognized by a high level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

In both the edge and level triggered modes, the IRQ inputs must remain active until after the falling edge of the first internal INTA#. When the IRQ input goes inactive before this time, a default IRQ7 vector will be returned.

5.6.4.8 End of Interrupt Operations

An EOI may occur in one of two fashions: by a command word write issued to the PIC before returning from a service routine, the EOI command; or automatically when AEOI bit in ICW4 is set to one.



5.6.4.9 Normal End of Interrupt

In Normal EOI, software writes an EOI command before leaving the interrupt service routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific. When a Non-Specific EOI command is issued, the PIC will clear the highest ISR bit of those that are set to one. Non-Specific EOI is the normal mode of operation of the PIC within the Intel® 6300ESB ICH, as the interrupt being serviced currently is the interrupt entered with the interrupt acknowledge. When the PIC is operated in modes which preserve the fully nested structure, software may determine which ISR bit to clear by issuing a Specific EOI. An ISR bit that is masked will not be cleared by a Non-Specific EOI when the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

5.6.4.10 Automatic End of Interrupt Mode

In this mode, the PIC will automatically perform a Non-Specific EOI operation at the trailing edge of the last interrupt acknowledge pulse. From a system standpoint, this mode should be used only when a nested multi-level interrupt structure is not required within a single PIC. The AEOI mode may only be used in the master controller and not the slave controller.

5.6.5 Masking Interrupts

5.6.5.1 Masking on an Individual Interrupt Request

Each interrupt request may be masked individually by the Interrupt Mask Register (IMR). This register is programmed through OCW1. Each bit in the IMR masks one interrupt channel. Masking IRQ2 on the master controller will mask all requests for service from the slave controller.

5.6.5.2 Special Mask Mode

Some applications may require an interrupt service routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The special mask mode enables all interrupts not masked by a bit set in the Mask Register. Normally, when an interrupt service routine acknowledges an interrupt without issuing an EOI to clear the ISR bit, the interrupt controller inhibits all lower priority requests. In the special mask mode, any interrupts may be selectively enabled by loading the Mask Register with the appropriate pattern. The special mask mode is set by OCW3 where: SSMM=1, SMM=1, and cleared where SSMM=1, SMM=0.

5.6.6 Steering PCI Interrupts

The Intel® 6300ESB ICH may be programmed to allow PIRQA#-PIRQH# to be internally routed to interrupts 3-7, 9-12, 14 or 15. The assignment is programmable through the PIRQx Route Control registers, located at 60-63h and 68-6Bh in function 0. One or more PIRQx# lines may be routed to the same IRQx input. When interrupt steering is not required, the Route Registers may be programmed to disable steering.

The PIRQx# lines are defined as active low, level sensitive to allow multiple interrupts on a PCI Board to share a single line across the connector. When a PIRQx# is routed to specified IRQ line, software must change the IRQ's corresponding ELCR bit to level

sensitive mode. The Intel® 6300ESB ICH will internally invert the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ may no longer be used by an ISA device (through SERIRQ). However, active low non-ISA interrupts may share their interrupt with PCI interrupts.

Internal sources of the PIRQs, including SCI and TCO interrupts, cause the external PIRQ to be asserted. The Intel® 6300ESB ICH receives the PIRQ input, like all of the other external sources, and routes it accordingly.

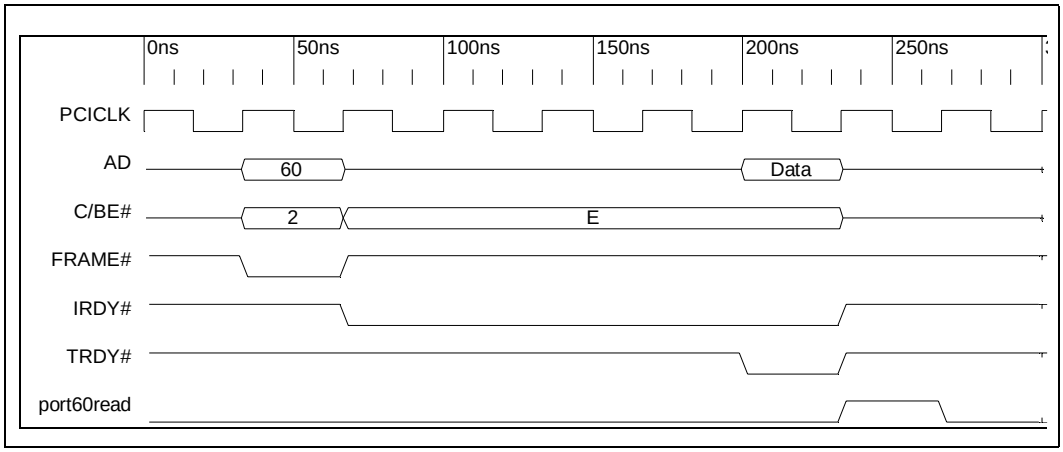
5.6.7 Special Handling of IRQ1 and IRQ12

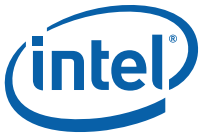
IRQ1 and IRQ12 interrupts are treated in a slightly different fashion from other interrupts in the system. In a legacy PC environment, these interrupts were not held active until serviced, but rather pulsed whenever a key or button was pressed. In newer systems, this pulsing is no longer done. However, the Intel® 6300ESB ICH must still handle old keyboard controllers which perform the pulse operation. Therefore, the Intel® 6300ESB ICH contains logic which may sample and hold these interrupts when so required.

Two register bits in configuration register D0h in function 0 enable the latching of IRQ1 and 12. IRQ1 may optionally be latched through bit 12, and IRQ12 may optionally be latched through bit 11. When these bits are set, the corresponding interrupt is held to the 8259 until an I/O read from port 60 is seen. The port 60 read is an indication to the keyboard controller that the interrupt has been serviced.

Another item to note is that on previous components (ICHx), it was always ensured that the keyboard controller would exist behind the Intel® 6300ESB ICH on the ISA bus. On Intel® 6300ESB ICH, this is not the case. Therefore, the clearing of the latch must be done through a snoop of port 60h. The waveform which performs this snoop is shown in Figure 13. Note that the signal which indicates that a port 60 read occurred is only one PCI clock wide. This cannot be a handshake signal because the Intel® 6300ESB ICH is not necessarily responding to the cycle.

Figure 13. Port 60 Read Clearing IRQ1 AND IRQ12 Latch





5.7 Advanced Interrupt Controller (APIC) (D29:F5)

There are two APICs in the Intel® 6300ESB ICH: APIC0 and APIC1 (device 29, function 5). APIC0's direct registers are assigned with base address FEC0xxxxH; however, only primary (legacy) PCI devices can write to these registers. APIC1's direct register are assigned with base address FEC1xxxxH. To support legacy devices/drivers on the PCI-X segment used with the Intel ICHx, APIC1 has an alternate base address FEC0xxxxH. This means devices on the PCI-X segment can only write to the IRQ Pin Assertion Register (either FEC0_0020H or FEC1_0020H) to generate an interrupt from APIC1. APIC1 writes to addresses FEC1_0020 to FEC1_0027 are claimed by APIC1 from PCI-X. Devices on the primary PCI Bus can write to IRQ Pin Assertion Register FEC0_0020H to generate an APIC0 interrupt. Devices/drivers on the PCI-X segment have write access only to the APIC1 IRQ Pin Assertion Register. Devices/drivers on the PCI segment can access only APIC0 registers. Since the Intel® 6300ESB ICH does not implement Hub Interface EOI special cycles, the MCH will translate EOI special cycle to a memory write cycle to EOI register at address FEC0_0040H and passes it to the Intel® 6300ESB ICH. This memory write cycle will be passed to both APIC0 and APIC1 internally.

From the CPU/MCH point of view, it should always use address FEC0xxxxH to access APIC0 registers and address FEC1xxxxH to access APIC1 registers. APIC1 will not respond to CPU/MCU's access to address FEC0xxxxH, other than the EOI cycle stated above.

5.7.1 Interrupt Handling

The I/O APIC handles interrupts very differently than the 8259. Briefly, these differences are:

- **Method of Interrupt Transmission.** Interrupts are handled without the need for the processor to run an interrupt acknowledge cycle. The Intel® 6300ESB ICH only supports FSB delivery of interrupts.
- **Interrupt Priority.** The priority of interrupts in the I/O APIC is independent of the interrupt number. For example, interrupt 10 may be given a higher priority than interrupt 3.
- **More Interrupts.** The I/O APIC in the Intel® 6300ESB ICH supports a total of 24 interrupts.
- **Multiple Interrupt Controllers.** The I/O APIC interrupt transmission protocol has an arbitration phase, which allows for multiple I/O APICs in the system with their own interrupt vectors. The Intel® 6300ESB ICH I/O APIC must arbitrate for the APIC bus before transmitting its interrupt message.

5.7.2 SMI/NMI/INIT/ExtINT Delivery Modes

These delivery modes are **not** supported by the Intel® 6300ESB ICH for the following reasons:

NMI/INIT: This signal has issues with delivery under power management. It cannot be delivered while the processor is in the Stop Grant state. In addition, this is a break event for power management. Breaking on the APIC bus message is more difficult than breaking on the pin.

SMI: On the 82093, the I/O APIC could deliver the SMI through the pin SMIOUT# or as an APIC bus message. When the message was masked by the OS, then the SMIOUT# will be used. In other words, there is no way to block the delivery of the SMI#, except through BIOS. Adding this interrupt to the I/O APIC only increases validation time.

5.7.3 Boot Interrupt

The Intel® 6300ESB ICH's APIC1 contains a capability to logically OR several of its interrupt inputs together to generate a single interrupt through PIC. This is necessary for systems that do not support the APIC, and for boot. The generated interrupt is routed to IRQ 9.

This interrupt is generated when the following conditions met:

- Boot interrupt is enabled in configuration register.
- Any of PXIRQ[3:0] or internal interrupt source is asserted.
- Boot interrupts are not MASKed in redirection table. (Refer to Bit 16 in the Redirection Table)
- IRQ9 of PIC is enabled with bit 6 set to 0 of the ETR1- Extended Features Register, D:31:F0:offset F4h,bit 6 or PIRQG# is assigned to an enabled IRQx of the PIC with ETR1 bit 6 set to 1. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information.

To support this function, all internal interrupt sources to APIC1 are level trigger, active low signals immediately after reset.

5.7.4 Interrupt Mapping

Only level-triggered interrupts can be shared. PCI interrupts (PIRQs and PXIRQs) are inherently shared on the board; these should, therefore, be programmed as level-triggered.

The following tables show the mapping of the various interrupts in Non-APIC and APIC modes.

Table 44. Interrupt Mapping in Non-APIC

Non-APIC Mode			
IRQ #	**Via SERIRQ**	**Direct from Pin**	**Internal Modules**
0	No	No	8254 Counter 0, MMT#0
1	Yes	No	
2	No	No	8259 #2 cascade only
3	Yes	No	Option for PIRQx
4	Yes	No	Option for PIRQx
5	Yes	No	Option for PIRQx
6	Yes	No	Option for PIRQx
7	Yes	No	Option for PIRQx
8	No	No	RTC, MMT#1
9	Yes	No	Option for PIRQx, SCI, TCO, boot interrupt
10	Yes	No	Option for PIRQx, SCI, TCO
11	Yes	No	Option for PIRQx, SCI, TCO, MMT #2
12	Yes	No	Option for PIRQx
13	No	No	FERR# Logic
14	Yes	Yes[3]	PIRQx, Storage (IDE/SATA) Primary (legacy mode)
15	Yes	Yes[3]	PIRQx, Storage (IDE/SATA) Secondary (legacy mode)

NOTES:
1. If an interrupt is used for Boot interrupt, PCI IRQ[A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ or IRQ14/15 pins). IRQ9 will be default to boot interrupt.
2. In non-APIC mode, the PCI interrupts are mapped to IRQ3, 4, 5, 6, 7, 9, 10, 11, 12, 14, or 15.
3. IRQ 14 and 15 can only be driven directly from the pins when in Legacy IDE mode.
4. If IRQ11 is used for MMT #2, software should ensure IRQ 11 is not shared with any other devices to ensure the proper operation of MMT #2. The Intel® 6300ESB ICH does not prevent sharing of IRQ 11.
5. SW: Boot interrupt may optionally be routed to PIRQG# output for programmable PIRQx# mapping.

There are two APICS within the Intel® 6300ESB ICH supporting 24 APIC interrupts each. APIC0 and APIC1 (Device 29 Function 5). APIC0 supports PCI messages interrupt from external device. Each interrupt has its own unique vector assigned by software. The interrupt vectors are mapped as follows.

5.7.5 APIC Bus Functional Description

Table 45. APIC Interrupt Mapping, APIC0 Agent

IRQ #	Via SERIRQ	Direct from pin	Via PCI message	Internal Modules
0	No	No	No	Cascade from 8259 #1
1	Yes	No	No[2]	
2	No	No	No	8254 Counter 0, MMT #0 (legacy mode)
3	Yes	No	No[2]	
4	Yes	No	No[2]	
5	Yes	No	No[2]	
6	Yes	No	No[2]	
7	Yes	No	No[2]	
8	No	No	No	RTC, MMT #1 (legacy mode)
9	Yes	No	No[2]	Option for SCI, TCO
10	Yes	No	No[2]	Option for SCI, TCO
11	Yes	No	No[2]	Option for SCI, TCO, MMT #2
12	Yes	No	No[2]	
13	No	No	No	FERR# logic
14	Yes	Yes[1]	No[2]	Storage (IDE/SATA) Primary (legacy mode)
15	Yes	Yes[1]	No[2]	Storage (IDE/SATA) Secondary (legacy mode)
16	PIRQ[A]#	PIRQ[A]#	No	USB1 UHCI Controller #1
17	PIRQ[B]#	PIRQ[B]#	No	AC'97 Audio, Modem, option for SMbus
18	PIRQ[C]#	PIRQ[C]#	No	Storage (IDE/SATA) native mode
19	PIRQ[D]#	PIRQ[D]#	No	USB 1.0 UHCI Controller #2
20	N/A	PIRQ[E]#	No[2]	Option for SCI, TCO, MMT #0,1,2
21	N/A	PIRQ[F]#	No[2]	Option for SCI, TCO, MMT #0,1,2
22	N/A	PIRQ[G]#	No[2]	Option for SCI, TCO, MMT #0,1,2
23	N/A	PIRQ[H]#	No[2]	USB 2.0 EHCI Controller, option for SCI, TCO, MMT #0,1,2

NOTES:
1. IRQ 14 and 15 may only be driven directly from the pins when in Legacy IDE mode.
2. NO from external devices, YES of access from processor
3. In APIC mode, the PCI interrupts A:H are mapped to IRQ[16:23].
4. When an interrupt is used for PCI IRQ[A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ)
5. When programming the polarity of internal interrupt sources on the APIC, interrupts 0 through 15 receive active-high internal interrupt sources; interrupts 16 through 23 receive active-low internal interrupt sources
6. When IRQ11 is used for MMT #2, software should ensure IRQ 11 is not shared with any other devices to ensure the proper operation of MMT #2. The Intel® 6300ESB ICH does not prevent sharing of IRQ 11.
7. PCI Message interrupts are not prevented by hardware in these cases. However, the system must not program these interrupts as edge-triggered (as required for PCI message interrupts) because the internal and external PIRQs on these inputs must be programmed in level-triggered modes.

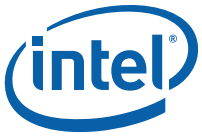


Table 46. APIC Interrupt Mapping, APIC1 Agent

APIC Mode Supported by APIC1 agent				
IRQ #	**Via SERIRQ**	**Direct from Pin**	**From PCI Message**	**Internal Modules**
0	No	PXIRQ0	Yes	
1	No	PXIRQ1	Yes	
2	No	PXIRQ2	Yes	
3	No	PXIRQ3	Yes	
4	No	No	Yes	
5	No	No	Yes	
6	No	No	Yes	
7	No	No	Yes	
8	No	No	Yes	
9	No	No	Yes	
10	No	No	Yes	WDT
11	No	No	Yes	
12	No	No	Yes	
13	No	No	Yes	
14	No	No	Yes	
15	No	No	Yes	
16	No	No	Yes	
17	No	No	Yes	
18	No	No	Yes	
19	No	No	Yes	
20	No	No	Yes	
21	No	No	Yes	
22	No	No	Yes	
23	No	No	Yes	

5.7.5.1 APIC Bus Arbitration

The I/O APIC uses one wire arbitration to win bus ownership. A rotating priority scheme is used for APIC bus arbitration. The winner of the arbitration becomes the lowest priority agent and assumes an arbitration ID of zero. All other agents, except the agent whose arbitration ID is 15, increment their Arbitration IDs by one. The agent whose ID was 15 will take the winner's arbitration ID and will increment it by one. Arbitration IDs are changed only for messages that are transmitted successfully (except for the Low Priority messages). A message is transmitted successfully when no CS error or acceptance error was reported for that message.

An APIC agent may use two different priority schemes: Normal or EOI. EOI has the highest priority. EOI priority is used to send EOI messages for level interrupts from a local APIC to an I/O APIC. When an agent requests the bus with EOI priority, all other agents requesting the bus with normal priorities will back off.

When the Intel® 6300ESB ICH detects a bus idle condition on the APIC Bus, and it has an interrupt to send over the APIC bus, it drives a start cycle to begin arbitration, by driving bit 0 to a '0' on an APICCLK rising edge. It then samples bit 1. When Bit 1 was a zero, then a local APIC started arbitration for an EOI message on the same clock edge that the Intel® 6300ESB ICH started arbitration. The Intel® 6300ESB ICH has thus lost arbitration and will stop driving the APIC bus.

When the Intel® 6300ESB ICH did not see an EOI message start, it will start transferring its arbitration ID, located in bits [27:24] of its Arbitration ID register (ARBID). Starting in Cycle 2, through Cycle 5, it will tri-state bit 0, and drive bit 1 to a '0' when ARBID[27] is a '1'. When ARBID[27] is a '0', it will also tri-state bit 1. At the end of each cycle, the Intel® 6300ESB ICH will sample the state of Bit 1 on the APIC bus. When the Intel® 6300ESB ICH did not drive Bit 1 (ARBID[27] = '0'), and it samples a '0', then another APIC agent started arbitration for the APIC bus at the same time as the Intel® 6300ESB ICH, and it has higher priority. The Intel® 6300ESB ICH will stop driving the APIC bus. Table 47 describes the arbitration cycles.

Table 47. Arbitration Cycles

Cycle	Bit 1	Bit 0	Comment
1	EOI	0	Bit 1 = 1: Normal, Bit 1 = 0: EOI
2	NOT (**ARBID[27]**)	1	Arbitration ID. When the Intel® 6300ESB ICH samples a different value than it sent, it lost arbitration.
3	NOT (**ARBID[26]**)	1	
4	NOT (**ARBID[25]**)	1	
5	NOT (**ARBID[24]**)	1	

5.7.5.2 Bus Message Formats

After bus arbitration, the winner is granted exclusive use of the bus and will drive its message. APIC messages come in four formats, determined by the Delivery Mode bits. These four messages are of different length, and are known by all APICs on the bus through the transmission of the Delivery Mode bits:

Table 48. APIC Message Formats

Message	# of Cycles	Delivery Mode Bits	Comments
EOI	14	xxx	End of Interrupt transmission from Local APIC to I/O APIC on Level interrupts. EOI is known by the EOI bit at the start of arbitration
Short	21	001, 010, 100, 101, 111	I/O APIC delivery on Fixed, NMI, SMI, Reset, ExtINT, and Lowest Priority with focus processor messages
Lowest Priority	33	001	Transmission of Lowest Priority interrupts when the status field indicates that the processor does not have focus.
Remote Read	39	011	Message from one Local APIC to another to read registers.

EOI Message for Level Triggered Interrupts

EOI messages are used by local APICs to send an EOI cycle occurring for a level triggered interrupt to an I/O APIC. This message is needed so that the I/O APIC may differentiate between a new interrupt on the interrupt line versus the same interrupt on the interrupt line. The target of the EOI is given by the local APIC through the transmission of the priority vector (V7 through V0) of the interrupt. Upon receiving this

message, the I/O APIC resets the Remote IRR bit for that interrupt. When the interrupt signal is still active after the IRR bit is reset, the I/O APIC will treat it as a new interrupt.

Table 49. EOI Message

Cycle	Bit 1	Bit 0	Comments
1	0	0	EOI message
2 - 5	ARBID	1	Arbitration ID
6	NOT(V7)	NOT(V6)	Interrupt vector bits V7 - V0 from redirection table register
7	NOT(V5)	NOT(V4)	
8	NOT(V3)	NOT(V2)	
9	NOT(V1)	NOT(V0)	
10	NOT(C1)	NOT(C0)	Check Sum from Cycles 6 - 9
11	1	1	Postamble
12	NOT(A)	NOT(A)	Status Cycle 0
13	NOT(A1)	NOT(A1)	Status Cycle 1
14	1	1	Idle

Short Message

Short messages are used for the delivery of Fixed, NMI, SMI, Reset, ExtINT and Lowest Priority with Focus processor interrupts. The Delivery Mode bits (M2–M0) specify the message. All short messages take 21 cycles including the idle cycle.

Table 50. Short Message

Cycle	Bit 1	Bit 0	Comments
1	1	0	Normal Arbitration
2 - 5	ARBID	1	Arbitration ID
6	NOT(DM)	NOT(M2)	DM[1] = Destination Mode from bit 11 of the redirection table register
7	NOT(M1)	NOT(M0)	M2-M0 = Delivery Mode from bits 10:8 of the redirection table register
8	NOT(L)	NOT(TM)	L = Level, TM = Trigger Mode
9	NOT(V7)	NOT(V6)	Interrupt vector bits V7 - V0 from redirection table register
10	NOT(V5)	NOT(V4)	
11	NOT(V3)	NOT(V2)	
12	NOT(V1)	NOT(V0)	
13	NOT(D7)	NOT(D6)	Destination field from bits 63:56 of redirection table register[1]
14	NOT(D5)	NOT(D4)	
15	NOT(D3)	NOT(D2)	
16	NOT(D1)	NOT(D0)	
17	NOT(C1)	NOT(C0)	Checksum for Cycles 6 - 16[2]
18	1	1	Postamble[3]
19	NOT(A)	NOT(A)	Status Cycle 0. See Table 51.
20	NOT(A1)	NOT(A1)	Status Cycle 1. See Table 51.
21	1	1	Idle

NOTES:

1. When DM is 0 (physical mode), cycles 15 and 16 are the APIC ID and cycles 13 and 14 are sent as ‘1’. When DM is 1 (logical mode), cycles 13 through 16 are the 8-bit Destination field. The interpretation of the logical mode 8-bit Destination field is performed by the local units using the Destination Format Register. Shorthands of “all-incl-self” and “all-excl-self” both use physical destination mode and a destination field containing APIC ID value of all ones. The sending APIC knows whether it should (incl) or should not (excl) respond to its own message.
2. The checksum field is the cumulative add (mod 4) of all data bits (DM, M0-3, L, TM, V0-7,D0-7). The APIC driving the message provides this checksum. This, in essence, is the lower two bits of an adder at the end of the message.
3. This cycle allows all APICs to perform various internal computations based on the information contained in the received message. One of the computations takes the checksum of the data received in cycles 6 through 16 and compares it with the value in cycle 18. When any APIC computes a different checksum than the one passed in cycle 17, the APIC will signal an error on the APIC bus (“00”) in cycle 19. When this occurs, all APICs will assume the message was never sent and the sender must try sending the message again, which includes re-arbitrating for the APIC bus. In lowest priority delivery when the interrupt has a focus processor, the focus processor will signal this by driving a “01” during cycle 19. This tells all the other APICs that the interrupt has been accepted, the arbitration is preempted, and short message format is used. Cycle 19 and 20 indicates the status of the message, i.e., accepted, check sum error, retry or error. The following table shows the status signal combinations and their meanings for all delivery modes.

Table 51. APIC Bus Status Cycle Definition

Delivery Mode	A	Comments	A1	Comments
Fixed, EOI	11	Checksum OK	1x	Error
			01	Accepted
			00	Retry
	10	Error	xx	
	01	Error	xx	
	00	Checksum Error	xx	
NMI, SMM, Reset, ExtINT	11	Checksum OK	1x	Error
			01	Accepted
			00	Error
	10	Error	xx	
	01	Error	xx	
	00	Checksum Error	xx	
Lowest Priority	11	Checksum OK: No Focus Processor	1x	Error
			01	End and Retry
			00	Go for Low Priority Arbitration
	10	Error	xx	
	01	Checksum OK: Focus Processor	xx	
	00	Checksum Error	xx	
Remote Read	11	Checksum OK	xx	
	10	Error	xx	
	01	Error	xx	
	00	Checksum Error	xx	

Lowest Priority without Focus Processor (FP) Message

This message format is used to deliver an interrupt in the lowest priority mode in which it does not have a Focus Process. Cycles 1 through 21 for this message is same as for the short message discussed above. Status cycle 19 identifies when there is a Focus processor (10) and a status value of 11 in cycle 20 indicates the need for lowest priority arbitration.

Table 52. Lowest Priority Message (Without Focus Processor)

Cycle	Bit 1	Bit 0	Comments
1	1	0	Normal Arbitration
2 - 5	ARBID	1	Arbitration ID
6	NOT(DM)	NOT(M2)	DM = Destination Mode from bit 11 of the redirection table register.
7	NOT(M1)	NOT(M0)	M2-M0 = Delivery Mode from bits 10:8 of the redirection table register.
8	NOT(L)	NOT(TM)	L = Level, TM = Trigger Mode
9	NOT(V7)	NOT(V6)	Interrupt vector bits V7 - V0 from redirection table register.
10	NOT(V5)	NOT(V4)	
11	NOT(V3)	NOT(V2)	
12	NOT(V1)	NOT(V0)	
13	NOT(D7)	NOT(D6)	Destination field from bits 63:56 of redirection table register.
14	NOT(D5)	NOT(D4)	
15	NOT(D3)	NOT(D2)	
16	NOT(D1)	NOT(D0)	
17	NOT(C1)	NOT(C0)	Checksum for Cycles 6 - 16
18	1	1	Postamble
19	NOT(A)	NOT(A)	Status Cycle 0.
20	NOT(A1)	NOT(A1)	Status Cycle 1.
21	P7	1	Inverted Processor Priority P7 - P0
22	P6	1	
23	P5	1	
24	P4	1	
25	P3	1	
26	P2	1	
27	P1	1	
28	P0	1	
29	ArbID3	1	
30	ArbID2	1	
31	ArbID1	1	
32	ArbID0	1	
33	S	S	Status
34	1	1	Idle

NOTES:
1. Cycle 21 through 28 are used to arbitrate for the lowest priority processor. The processor that takes part in the arbitration drives the processor priority on the bus. Only the local APICs that have “free interrupt slots” will participate in the lowest priority arbitration.
2. Cycles 29 through 32 are used to break tie in case two more processors have lowest priority. The bus arbitration ID's are used to break the tie.

Remote Read Message

Remote read message is used when a local APIC wishes to read the register in another local APIC. The I/O APIC in the Intel® 6300ESB ICH neither generates or responds to this cycle. The message format is same as short message for the first 21 cycles.

Table 53. Remote Read Message (Sheet 1 of 2)

Cycle	Bit 1	Bit 0	Comments
1	1	0	Normal Arbitration
2 - 5	ARBID	1	Arbitration ID
6	NOT(DM)	NOT(M2)	DM = Destination Mode from bit 11 of the redirection table register.
7	NOT(M1)	NOT(M0)	M2-M0 = Delivery Mode from bits 10:8 of the redirection table register.
8	NOT(L)	NOT(TM)	L = Level, TM = Trigger Mode
9	NOT(V7)	NOT(V6)	Interrupt vector bits V7 - V0 from redirection table register.
10	NOT(V5)	NOT(V4)	
11	NOT(V3)	NOT(V2)	
12	NOT(V1)	NOT(V0)	
13	NOT(D7)	NOT(D6)	Destination field from bits 63:56 of redirection table register.
14	NOT(D5)	NOT(D4)	
15	NOT(D3)	NOT(D2)	
16	NOT(D1)	NOT(D0)	
17	NOT(C1)	NOT(C0)	Checksum for Cycles 6 - 16
18	1	1	Postamble
19	NOT(A)	NOT(A)	Status Cycle 0.
20	NOT(A1)	NOT(A1)	Status Cycle 1.
21	d31	d30	Remote register data 31-0
22	d29	d28	
23	d27	d26	
24	d25	d24	
25	d23	d22	
26	d21	d20	
27	d19	d18	
28	d17	d16	
29	d15	d14	
30	d13	d12	
31	d11	d10	
32	d09	d08	
33	d07	d06	
34	d05	d04	
35	d03	d02	
36	d01	d00	

Table 53. Remote Read Message (Sheet 2 of 2)

Cycle	Bit 1	Bit 0	Comments
37	S	S	Data Status: 00 = valid, 11 = invalid
38	C	C	Check Sum for data d31-d00
39	1	1	Idle

NOTE: Cycle 21 through 36 contain the remote register data. The status information in cycle 37 specifies when the data is valid or invalid. Remote read cycle is always successful (although the data may be valid or invalid) in that it is never retried. The reason for this is that Remote Read is a debug feature, and a “hung” remote APIC that is unable to respond should not cause the debugger to hang.

5.7.6 PCI Message-Based Interrupts

5.7.6.1 Theory of Operation

The following scheme is only supported when the internal I/O(x) APIC is used, rather than just the 8259.

The Intel® 6300ESB ICH supports the new method for PCI devices to deliver interrupts as write cycles rather than using the traditional PIRQ[A:D] signals. Essentially, the PCI devices are given a write path directly to a register that will cause the desired interrupt. This mode is only supported when the Intel® 6300ESB ICH’s internal I/O APIC is enabled. Upon recognizing the write from the peripheral, the Intel® 6300ESB ICH will send the interrupt message to the processor using the I/O APIC’s serial bus.

The interrupts associated with the PCI Message-based interrupt method must be set up for edge triggered mode, rather than level triggered, since the peripheral only does the write to indicate the edge.

The following sequence is used:

1. During PCI PnP, the PCI peripheral is first programmed with an address (MESSAGE_ADDRESS) and data value (MESSAGE_DATA) that will be used for the interrupt message delivery. For the Intel® 6300ESB ICH, the MESSAGE_ADDRESS is the IRQ Pin Assertion Register, which is mapped to memory location: FEC0_0020h.
2. To cause the interrupt, the PCI peripheral requests the PCI bus and when granted, writes the MESSAGE_DATA value to the location indicated by the MESSAGE_ADDRESS. The MESSAGE_DATA value indicates which interrupt occurred. This MESSAGE_DATA value is a binary encoded. For example, to indicate that interrupt 7 should go active, the peripheral will write a binary value of 0000111. The MESSAGE_DATA will be a 32-bit value, although only the lower 5 bits are used.
3. When the PRQ bit in the APIC Version Register is set, the Intel® 6300ESB ICH positively decodes the cycles (as a slave) in Medium time.
4. The Intel® 6300ESB ICH decodes the binary value written to MESSAGE_ADDRESS and sets the appropriate IRR bit in the internal I/O APIC. The corresponding interrupt must be set up for edge-triggered interrupts. The Intel® 6300ESB ICH supports interrupts 00h through 23h. Binary values outside this range will not cause any action.
5. After sending the interrupt message to the processor, the Intel® 6300ESB ICH will automatically clear the interrupt.

Since they are edge triggered, the interrupts that are allocated to the PCI bus for this scheme may not be shared with any other interrupt (such as the standard PCI PIRQ[A:D], those received via SERIRQ#, or the internal level-triggered interrupts such as SCI or TCO).

The Intel® 6300ESB ICH will ignore interrupt messages sent by PCI masters that attempt to use IRQ0, 2, 8, or 13.

5.7.6.2 Registers and Bits Associated with PCI Interrupt Delivery

Capabilities Indication

The capability to support PCI interrupt delivery will be indicated through ACPI configuration techniques. This involves the BIOS creating a data structure that gets reported to the ACPI configuration software. The OS reads the PRQ bit in the APIC Version Register to see when the Intel® 6300ESB ICH is capable of supporting PCI-based interrupt messages.

Interrupt Message Register

The PCI devices will all write their message into the IRQ Pin Assertion Register, which is a memory-Mapped register located at the APIC base memory location + 20h.

5.7.7 Processor System Bus Interrupt Delivery

5.7.7.1 Theory of Operation

For processors that support Processor System Bus interrupt delivery, the Intel® 6300ESB ICH has an option to let the integrated I/O APIC behave as an I/O (x) APIC. In this case, it will deliver interrupt messages to the processor in a parallel manner, rather than using the I/O APIC serial scheme. The Intel® 6300ESB ICH is intended to be compatible with the I/O (x) APIC specification, Rev 1.1

This is done by the Intel® 6300ESB ICH writing (through the Hub Interface) to a memory location that is snooped by the processor(s). The processor(s) snoop the cycle to know which interrupt goes active.

The processor enables the mode by setting the I/O APIC Enable (APIC_EN) bit and by setting the DT bit in the I/O APIC ID register.

The following sequence is used:

1. When the Intel® 6300ESB ICH detects an interrupt event (active edge for edge-triggered mode or a change for level-triggered mode), it sets or resets the internal IRR bit associated with that interrupt.
2. Internally, the Intel® 6300ESB ICH requests to use the bus in a way that automatically flushes upstream buffers. This may be internally implemented similar to a DMA device request.
3. The Intel® 6300ESB ICH then delivers the message by performing a write cycle to the appropriate address with the appropriate data. The address and data formats are described below in Section 5.7.7.5, “Interrupt Message Format”.

Note: PSB Interrupt Delivery compatibility with processor clock control depends on the processor, not the Intel® 6300ESB ICH.

5.7.7.2 Edge-Triggered Operation

In this case, the “Assert Message” is sent when there is an inactive-to-active edge on the interrupt.

5.7.7.3 Level-Triggered Operation

In this case, the “Assert Message” is sent when there is an inactive-to-active edge on the interrupt. When the interrupt is still active after an EOI, another “Assert Message” is sent to indicate that the interrupt is still active.

5.7.7.4 Registers Associated with Processor System Bus Interrupt Delivery

Capabilities Indication

The capability to support Processor System Bus interrupt delivery will be indicated through ACPI configuration techniques. This involves the BIOS creating a data structure that gets reported to the ACPI configuration software.

DT Bit in the Boot Configuration Register

This enables the Intel® 6300ESB ICH to deliver interrupts as memory writes. This bit is ignored when the APIC mode is not enabled.

5.7.7.5 Interrupt Message Format

The Intel® 6300ESB ICH writes the message to PCI (and to the Host Controller) as a 32-bit memory write cycle. It uses the formats shown in Table 54 and Table 55 for the Address and Data.

The local APIC (in the processor) has a delivery mode option to interpret Processor System Bus messages as a SMI in which case the processor treats the incoming interrupt as a SMI instead of as an interrupt. This does not mean that the Intel® 6300ESB ICH has any way to have a SMI source from the Intel® 6300ESB ICH power management logic cause the I/OAPIC to send an SMI message (there is no way to do this). The Intel® 6300ESB ICH’s I/OAPIC may only send interrupts due to interrupts which do not include SMI, NMI or INIT. This means that in IA32/IA64 based platforms, Processor System Bus interrupt message format delivery modes 010 (SMI/PMI), 100 (NMI), and 101 (INIT) as indicated in this section, must not be used and is not supported. Only the hardware pin connection is supported by the Intel® 6300ESB ICH.

Table 54. Interrupt Message Address Format (Sheet 1 of 2)

Bit	Description
31:20	Will always be FEEh
19:12	**Destination ID:** This will be the same as bits 63:56 of the I/O Redirection Table entry for the interrupt associated with this message.
11:4	**Extended Destination ID:** This will be the same as bits 55:48 of the I/O Redirection Table entry for the interrupt associated with this message.

Table 54. Interrupt Message Address Format (Sheet 2 of 2)

Bit	Description
3	**Redirection Hint:** This bit is used by the processor host bridge to allow the interrupt message to be redirected. 0 = The message will be delivered to the agent (processor) listed in bits 19:12. 1 = The message will be delivered to an agent with a lower interrupt priority. This may be derived from bits 10:8 in the Data Field (see below). The Redirection Hint bit will be a 1 when bits 10:8 in the delivery mode field associated with corresponding interrupt are encoded as 001 (Lowest Priority). Otherwise, the Redirection Hint bit will be 0.
2	**Destination Mode:** This bit is used only the Redirection Hint bit is set to 1. When the Redirection Hint bit and the Destination Mode bit are both set to 1, then the logical destination mode is used, and the redirection is limited only to those processors that are part of the logical group as based on the logical ID.
1:0	Will always be 00.

Table 55. Interrupt Message Data Format

Bit	Description
31:16	Will always be 0000h.
15	**Trigger Mode:** 1 = Level, 0 = Edge. Same as the corresponding bit in the I/O Redirection Table for that interrupt.
14	**Delivery Status:** 1 = Assert, 0 = Deassert. When using edge-triggered interrupts, this bit will always be 1, since only the assertion is sent. When using level-triggered interrupts, this bit indicates the state of the interrupt input.
13:12	Will always be 00
11	**Destination Mode:** 1 = Logical. 0 = Physical. Same as the corresponding bit in the I/O Redirection Table for that interrupt.
10:8	**Delivery Mode:** This is the same as the corresponding bits in the I/O Redirection Table for that interrupt. 000 = Fixed 100 = NMI 001 = Lowest Priority 101 = INIT 010 = SMI/PMI 110 = Reserved 011 = Reserved 111 = ExtINT
7:0	**Vector:** This is the same as the corresponding bits in the I/O Redirection Table for that interrupt.

5.8 Serial Interrupt (D31:F0)

The Intel® 6300ESB ICH supports a serial IRQ scheme. This allows a single signal to be used to report interrupt requests. The signal used to transmit this information is shared between the host, the Intel® 6300ESB ICH, and all peripherals that support serial interrupts. The signal line, SERIRQ, is synchronous to PCI clock, and follows the sustained tri-state protocol that is used by all PCI signals. This means that when a device has driven SERIRQ low, it will first drive it high synchronous to PCI clock and release it the following PCI clock. The serial IRQ protocol defines this sustained tri-state signaling in the following fashion:

- **S - Sample Phase.** Signal driven low
- **R - Recovery Phase.** Signal driven high

- **T - Turn-around Phase.** Signal released

The Intel® 6300ESB ICH supports a message for 21 serial interrupts. These represent the 15 ISA interrupts (IRQ0-1, 2-15), the four PCI interrupts, and the control signals SMI# and IOCHK#. The serial IRQ protocol does not support the additional APIC interrupts (20-23).

Serial interrupt information is transferred using three types of frames:

- **Start Frame:** SERIRQ line driven low by the Intel® 6300ESB ICH to indicate the start of IRQ transmission
- **Data Frames:** IRQ information transmitted by peripherals. The Intel® 6300ESB ICH will support 21 data frames.
- **Stop Frame:** SERIRQ line driven low by the Intel® 6300ESB ICH to indicate end of transmission and next mode of operation.

Note: When the IDE primary and secondary controllers are configured for native IDE mode, the only way to use the internal IRQ14 and IRQ15 connections to the Interrupt Controllers is through the Serial Interrupt pin.

5.8.1 Start Frame

The serial IRQ protocol has two modes of operation which affect the start frame. These two modes are: Continuous, where the Intel® 6300ESB ICH is solely responsible for generating the start frame; and Quiet, where a serial IRQ peripheral is responsible for beginning the start frame.

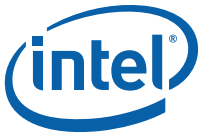
The mode that must first be entered when enabling the serial IRQ protocol is continuous mode. In this mode, the Intel® 6300ESB ICH will assert the start frame. This start frame is 4, 6, or 8 PCI clocks wide based upon the Serial IRQ Control Register, bits 1:0 at 64h in Device 31:Function 0 configuration space. This is a polling mode.

When the serial IRQ stream enters quiet mode (signaled in the Stop Frame), the SERIRQ line remains inactive and pulled up between the Stop and Start Frame until a peripheral drives the SERIRQ signal low. The Intel® 6300ESB ICH senses the line low and continues to drive it low for the remainder of the Start Frame. Since the first PCI clock of the start frame was driven by the peripheral in this mode, the Intel® 6300ESB ICH will drive the SERIRQ line low for 1 PCI clock less than in continuous mode. This mode of operation allows for a quiet, and therefore lower power, operation.

5.8.2 Data Frames

Once the Start frame has been initiated, all of the SERIRQ peripherals must start counting frames based on the rising edge of SERIRQ. Each of the IRQ/DATA frames has exactly 3 phases of 1 clock each:

- **Sample Phase.** During this phase, the SERIRQ device drives SERIRQ low when the corresponding interrupt signal is low. When the corresponding interrupt is high, the SERIRQ devices will tri-state the SERIRQ signal. The SERIRQ line will remain high due to pull-up resistors (there is no internal pull-up resistor on this signal, an external pull-up resistor is required). A low level during the IRQ0-1 and IRQ2-15 frames indicates that an active-high ISA interrupt is not being requested, but a low level during the PCI INT[A:D], SMI#, and IOCHK# frame indicates that an active-low interrupt is being requested.
- **Recovery Phase.** During this phase, the device will drive the SERIRQ line high when in the Sample Phase it was driven low. When it was not driven in the sample phase, it will be tri-stated in this phase.
- **Turn-around Phase.** The device will tri-state the SERIRQ line.



5.8.3 Stop Frame

After all data frames, a Stop Frame is driven by the Intel® 6300ESB ICH. The SERIRQ signal is driven low by the Intel® 6300ESB ICH for 2 or 3 PCI clocks. The number of clocks is determined by the SERIRQ configuration register. The number of clocks determines the next mode:

Table 56. Stop Frame Explanation

Stop Frame Width	Next Mode
2 PCI clocks	**Quiet Mode.** Any SERIRQ device may initiate a Start Frame
3 PCI clocks	**Continuous Mode.** Only the host (Intel® 6300ESB ICH) may initiate a Start Frame

5.8.4 Specific Interrupts Not Supported via SERIRQ

There are three interrupts seen through the serial stream which are not supported by the Intel® 6300ESB ICH. These interrupts are generated internally, and are not sharable with other devices within the system. These interrupts are:

- IRQ0. Heartbeat interrupt generated off of the internal 8254 counter 0.
- IRQ8#. RTC interrupt may only be generated internally.
- IRQ13. Floating point error interrupt generated off of the processor assertion of FERR#.

The Intel® 6300ESB ICH will ignore the state of these interrupts in the serial stream, and will not adjust their level based on the level seen in the serial stream. In addition, the interrupts IRQ14 and IRQ15 from the serial stream are treated differently than their ISA counterparts. These two frames are not passed to the Bus Master IDE logic. The Bus Master IDE logic expects IDE to be behind the Intel® 6300ESB ICH.

5.8.5 Data Frame Format

Table 57 shows the format of the data frames. For the PCI interrupts (A-D), the output from the Intel® 6300ESB ICH is ANDed with the PCI input signal. This way, the interrupt may be signaled through both the PCI interrupt input signal and through the SERIRQ signal (they are shared).

Table 57. Data Frame Format (Sheet 1 of 2)

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
1	IRQ0	2	Ignored. IRQ0 may only be generated through the internal 8524.
2	IRQ1	5	Before Port 60h latch
3	SMI#	8	Causes SMI# when low. Will set the SERIRQ_SMI_STS bit, (bit 15).
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	
8	IRQ7	23	

Table 57. Data Frame Format (Sheet 2 of 2)

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
9	IRQ8	26	Ignored. IRQ8# may only be generated internally or on ISA.
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	Before Port 60h latch
14	IRQ13	41	Ignored. IRQ13 may only be generated from FERR#.
15	IRQ14	44	Do not include in BM IDE interrupt logic.
16	IRQ15	47	Do not include in BM IDE interrupt logic.
17	IOCHCK#	50	Same as ISA IOCHCK# going active.
18	PCI INTA#	53	Drive PIRQA#
19	PCI INTB#	56	Drive PIRQB#
20	PCI INTC#	59	Drive PIRQC#
21	PCI INTD#	62	Drive PIRQD#

NOTE: SIU_SERIRQ Period 13 is used to transfer IRQ12.

5.9 Real Time Clock (D31:F0)

5.9.1 RTC Overview

The Real Time Clock (RTC) module provides a battery backed-up date and time keeping device with two banks of static RAM with 128 bytes each, although the first bank has 114 bytes for general purpose usage. Three interrupt features are available: time of day alarm with once a second to once a month range, periodic rates of 122 µs to 500 ms, and end of update cycle notification. Seconds, minutes, hours, days, day of week, month, and year are counted. Daylight savings compensation is optional. The hour is represented in twelve or twenty-four hour format, and data may be represented in BCD or binary format. The design is meant to be functionally compatible with the Motorola* MS146818B. The time keeping comes from a 32.768 KHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block has very specific functions. The first ten are for time and date information. The next four (0Ah to 0Dh) are registers, which configure and report RTC functions. See Table 290 for more information.

The time and calendar data should match the data mode (BCD or binary) and hour mode (12 or 24 hour) as selected in register B. It is up to the programmer to make sure that data stored in these locations is within the reasonable values ranges and represents a possible date and time. The exceptions to these ranges is to store a value of C0-FFh in the alarm bytes to indicate a “do not care” situation. All alarm conditions must match to trigger an Alarm Flag, which could trigger an alarm interrupt when enabled. The SET bit must be one while programming these locations to avoid clashes with an update cycle. Access to time and date information is done through the RAM locations. When a RAM read from the ten time and date bytes is attempted during an update cycle, the value read will not necessarily represent the true contents of those locations. Any RAM writes under the same conditions will be ignored.

Note: The Intel® 6300ESB ICH supports the ability to generate an SMI# based on a century rollover. See Section 5.9.1.4, "Century Rollover" for more information on the century rollover.

Note: The Intel® 6300ESB ICH does not implement month/year alarms.

5.9.1.1 Update Cycles

An update cycle occurs once a second, when the SET bit of register B is not asserted and the divide chain is properly configured. During this procedure, the stored time and date will be incremented, overflow will be checked, a matching alarm condition will be checked, and the time and date will be rewritten to the RAM locations. The update cycle will start at least 488 µs after the UIP bit of register A is asserted, and the entire cycle will not take more than 1984 µs to complete. The time and date RAM locations (0-9) will be disconnected from the external bus during this time.

To avoid update and data corruption conditions, external RAM access to these locations may safely occur at two times. When an updated-ended interrupt is detected, almost 999 ms is available to read and write the valid time and date. When the UIP bit of Register A is detected to be low, there is at least 488 µs before the update cycle begins.

Warning: The overflow conditions for leap years and daylight savings adjustments are based on more than one date or time item. To ensure proper operation when adjusting the time, the new time and data values should be set at least two seconds before one of these conditions (leap year, daylight savings time adjustments) occurs.

5.9.1.2 Interrupts

The real-time clock interrupt is internally routed within the Intel® 6300ESB ICH both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt does not leave the Intel® 6300ESB ICH, nor is it shared with any other interrupt. IRQ8# from the SERIRQ stream is ignored.

5.9.1.3 Lockable RAM Ranges

The RTC's battery-backed RAM supports two 8-byte ranges that may be locked through the configuration space. When the locking bits are set, the corresponding range in the RAM will not be readable or writable. A write cycle to those locations will have no effect. A read cycle to those locations will return an undefined value.

Once a range is locked, the range may be unlocked only by a hard reset, which will invoke the BIOS and allow it to relock the RAM range.

5.9.1.4 Century Rollover

The Intel® 6300ESB ICH will detect a rollover when the Year byte (RTC I/O space, index offset 09h) transitions form 99 to 00. Upon detecting the rollover, the Intel® 6300ESB ICH will set the NEWCENTURY_STS bit (TCOBASE + 04h, bit 7). When the system is in an S0 state, this will cause an SMI#. The SMI# handler may update registers in the RTC RAM that are associated with century value. When the system is in a sleep state (S1-S5) when the century rollover occurs, the Intel® 6300ESB ICH will also set the NEWCENTURY_STS bit, but no SMI# is generated. When the system resumes from the sleep state, BIOS should check the NEWCENTURY_STS bit and update the century value in the RTC RAM.

5.9.1.5 Clearing Battery-Backed RTC RAM

Clearing CMOS RAM in an Intel® 6300ESB ICH-based platform may be done by using a jumper on RTCRST# or GPI, or using SAFEMODE strap. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low.

Using RTCRST# to clear CMOS:

A jumper on RTCRST# may be used to clear CMOS values, as well as reset to default, the state of those configuration bits that reside in the RTC power well. When the RTCRST# is strapped to ground, the RTC_PWR_STS bit (D31:F0:A4h bit 2) will be set and those configuration bits in the RTC power well will be set to their default state. BIOS may monitor the state of this bit, and manually clear the RTC CMOS array once the system is booted. The normal position would cause RTCRST# to be pulled up through a weak pull-up resistor. Table 58 presents which bits are set to their default state when RTCRST# is asserted.

Table 58. Configuration Bits Reset By RTCRST# Assertion

Bit Name	Default State	Register	Location	Bit(s)
AIE	RTC Reg B	I/O space	5	0
AF	RTC Reg C	I/O space	5	0
PWR_FLR	GEN_PMCON_3	D31:F0:A4h	1	0
AFTERG3_EN	GEN_PMCON_3	D31:F0:A4h	0	0
RTC_PWR_STS	GEN_PMCON_3	D31:F0:A4h	2	1
PRBTNOR_STS	PM1_STS	PMBase + 00h	11	0
PME_EN	GPE0_EN	PMBase + 2Ah	11	0
RI_EN	GPE0_EN	PMBase + 2Ah	8	0
NEW_CENTURY_STS	TCO1_STS	TCOBase + 04h	7	0
INTRD_DET	TCO2_STS	TCOBase + 06h	0	0
TOP_SWAP	GEN_STS	D31:F0:D4h	13	0
RTC_EN	PM1_EN	PMBase + 02h	10	0
BATLOW_EN	GPE0_EN	PMBase + 2Ah	10	0

Using a GPI to Clear CMOS:

A jumper on a GPI may also be used to clear CMOS values. BIOS would detect the setting of this GPI on system boot-up, and manually clear the CMOS array.

Using the SAFEMODE Strap to Clear CMOS:

A jumper on AC_SDOUT (SAFEMODE strap) may also be used to clear CMOS values. BIOS would detect the setting of the SAFE_MODE status bit (D31:F0: Offset D4h bit 2) on system boot-up, and manually clear the CMOS array.

Note: Both the GPI and SAFEMODE strap techniques to clear CMOS require multiple steps to implement. The system is booted with the jumper in new position, then powered back down. The jumper is replaced back to the normal position, then the system is rebooted again. The RTCRST# jumper technique allows the jumper to be moved and then replaced, all while the system is powered off. Then, once booted, the RTC_PWR_STS may be detected in the set state.

5.10 Processor Interface (D31:F0)

The Intel® 6300ESB ICH interfaces to the processor with a variety of signals

- Standard Outputs to the processor: A20M#, SMI#, NMI, INIT#, INTR, STPCLK#, IGNNE#, CPUSLP#
- The FERR# input to the Intel® 6300ESB ICH has special buffer requirements. The Vil threshold is compatible with processors that drive FERR# no higher than 1.3V +/- 5%.

Most Intel® 6300ESB ICH outputs to the processor use standard buffers. The Intel® 6300ESB ICH has a separate V_{CC} signal which is pulled up at the system level to the processor voltage, and thus determines V_{OH} for the outputs to the processor. Note that this is different than previous generations of chips, that have used open-drain outputs. This new method saves up to 12 external pull-up resistors.

The Intel® 6300ESB ICH does not support the processor's FRC mode.

5.10.1 Processor Interface Signals

This section describes each of the signals that interface between the Intel® 6300ESB ICH and the processor(s).

5.10.1.1 A20M#

The A20M# signal will be active (low) when both of the following conditions are true:

- The ALT_A20_GATE bit (Bit 1 of PORT92 register) is a '0'
- The A20GATE input signal is a '0'

The A20GATE input signal is expected to be generated by the external microcontroller (KBC).

5.10.1.2 INIT#

The INIT# signal will be active (low) based on any one of several events described in Table 59. When any of these events occur, INIT# will be driven low for 16 PCI clocks, then driven high.

The 16-clock counter for INIT# assertion will halt while STPCLK# is active. INIT# will not go active until after STPCLK# goes inactive.

Table 59. INIT# Going Active (Sheet 1 of 2)

Cause of INIT# Going Active	Comment
Shutdown special cycle from processor.	
PORT92 write, where INIT_NOW (bit 0) transitions from a 0 to a 1.	

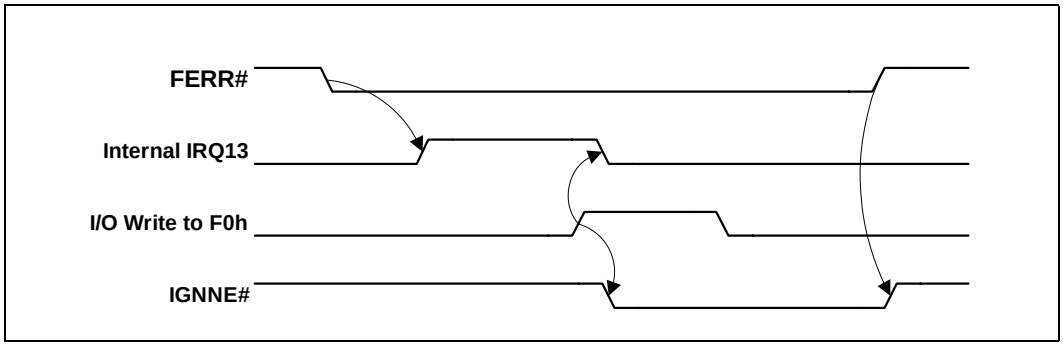
Table 59. INIT# Going Active (Sheet 2 of 2)

Cause of INIT# Going Active	Comment
PORTCF9 write, where RST_CPU (bit 2) was a 0 and SYS_RST(bit 1) transitions from 0 to 1.	
RCIN# input signal goes low. RCIN# is expected to be driven by the external microcontroller (KBC).	0 to 1 transition on RCIN# must occur before the Intel® 6300ESB ICH will arm INIT# to be generated again. **NOTE:** RCIN# signal is expected to be high during S1-M and low during S3, S4, and S5 states. Transition on the RCIN# signal in those states (or the transition to those states) may not necessarily cause the INIT# signal to be generated to the processor
CPU BIST	In order to enter BIST, the software sets CPU_BIST_EN bit and then does a full processor reset using the CF9 register.

5.10.1.3 FERR# / IGNNE# (Coprocessor Error)

The Intel® 6300ESB ICH supports the coprocessor error function with the FERR#/ IGNNE# pins. The function is enabled through the COPROC_ERR_EN bit (Device 31:Function 0, Offset D0, bit 13). FERR# is tied directly to the Coprocessor Error signal of the processor. When FERR# is driven active by the processor, IRQ13 goes active (internally). When it detects a write to the COPROC_ERR register, the Intel® 6300ESB ICH negates the internal IRQ13 and drives IGNNE# active. IGNNE# remains active until FERR# is driven inactive. IGNNE# is never driven active unless FERR# is active.

Figure 14. Coprocessor Error Timing Diagram



When COPROC_ERR_EN is not set, the assertion of FERR# will have not generate an internal IRQ13, nor will the write to F0h generate IGNNE#.

5.10.1.4 NMI

Non-Maskable Interrupts (NMIs) may be generated by several sources, as described in Table 60.

Table 60. NMI Sources

Cause of NMI	Comment
SERR# goes active (either internally, externally through SERR# signal, or through a message from MCH)	May instead be routed to generate an SCI, through the NMI2SCI_EN bit (Device 31:Function 0, offset 4E, bit 11).
IOCHK# goes active via SERIRQ# stream (ISA System Error)	May instead be routed to generate an SCI, through the NMI2SCI_EN bit (Device 31:Function 0, offset 4E, bit 11).
D30_PD_STS register (D30:F0:06h), bit 8 (Detected parity error on Hub Interface)	Enabled by D30:F0:04h, bit 6
D30_SECSTS register (D30:F0:1Eh), bit 8 (Detected parity error on PCI by North PCI unit)	Enabled by D30:F0:04h, bit 6
D31F0_DEV_STS register (D31:F0:06h), bit 8 (Detected parity error on PCI by South PCI unit)	

5.10.1.5 STPCLK# and CPUSLP# Signals

The Intel® 6300ESB ICH power management logic controls these active-low signals. Refer to Section 5.11, “Power Management (D31:F0)” for more information on the functionality of these signals.

5.10.2 Dual Processor Issues

5.10.2.1 Signal Differences

In dual processor designs, some of the processor signals are used differently than in uniprocessor designs.

Table 61. DP Signal Differences

Signal	Difference
A20M# / A20GATE	Generally not used, but still supported by the Intel® 6300ESB ICH.
STPCLK#	Used for S1 State as well as preparation for entry to S3-S5. Also allows for THERM# based throttling (not through ACPI control methods). Should be connected to both processors.
FERR# / IGNNE#	Generally not used, but still supported by the Intel® 6300ESB ICH.

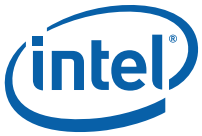
5.10.2.2 Dual Processor Power Management

For multiple-CPU (or Multiple-core) configurations in which more than one Stop Grant cycle may be generated, the MCH is expected to count Stop Grant cycles and only pass the last one through to the 6300ESB. This prevents the 6300ESB from getting out of sync with the processor on multiple STPCLK# assertions.

Because the S1 state will have the STPCLK# signal active, the STPCLK# signal can be connected to both processors. However, for ACPI implementations, the BIOS must indicate that the 6300ESB only supports the C1 state for dual-processor designs.

In going to the S1 state, multiple Stop-Grant cycles will be generated by the CPUs. The Intel 6300ESB also has the option to assert the CPU's SLP# signal (CPUSLP#). It is assumed that prior to setting the SLP_EN bit (which causes the transition to the S1 state), the CPUs will not be executing code that is likely to delay the Stop-Grant cycles.

In going to the S3, S4, or S5 states, the system will appear to pass through the S1 state; thus, STPCLK# and SLP# are also used. During the S3, S4, and S5 states, both processors will lose power. Upon exit from those states, the processors will have their power restored.



5.11 Power Management (D31:F0)

5.11.1 Features

- ACPI Power and Thermal Management Support
  - Processor THRMTRIP# emergency shutdown
  - ACPI 24-Bit Timer
  - Software initiated throttling of processor performance for Thermal and Power Reduction
  - Hardware Override to throttle processor performance when system too hot
  - SCI and SMI# Generation
- PME# Signal for Wake Up from Low-Power states (PME signal shared between both PCI and PCI-X interfaces)
- SYS_Reset# input to eliminate external glue logic
- System Clock Control
  - ACPI C2 state: Stop-Grant (in desktop) or Quickstart (in mobile) state (using STPCLK# signal) halts processor's instruction stream
- System Sleeping State Control
  - ACPI S1 state: Like C2 state (only STPCLK# active, and SLP# optional)
  - ACPI S3 state - Suspend to RAM (STR)
  - ACPI S4 state - Suspend-to-Disk(STD)
  - ACPI G2/S5 state - Soft Off(SOFF)
  - Power Failure Detection and Recovery
  - Supports new output signal - SLP_S4#
- Streamlined Legacy Power Management Support for APM-Based Systems
- Support for Prescott Processor

5.11.2 Intel® 6300ESB ICH Power States and Transition Rules

Table 62 shows the power states defined for Intel® 6300ESB ICH-based platforms. The state names generally match the corresponding ACPI states.

Table 62. General Power States for Systems Using Intel® 6300ESB ICH

State/ Substates	Legacy Name / Description
G0/S0/C0	**Full On:** Processor operating. Individual devices may be shut down to save power. The different processor operating levels are defined by Cx states, as shown in Table 63. Within the C0 state, the Intel® 6300ESB ICH may throttle the STPCLK# signal to reduce power consumption. The throttling may be initiated by software or by the THRM# input signal.
G0/S0/C1	**Auto-Halt:** Processor has executed a AutoHalt instruction and is not executing code. The processor snoops the bus and maintains cache coherency.
G0/S0/C2	**Stop-Grant:** The STPCLK# signal goes active to the processor. The processor performs a Stop-Grant cycle, halts its instruction stream, and remains in that state until the STPCLK# signal goes inactive. In the Stop-Grant (desktop) state, the processor snoops the bus and maintains cache coherency. **NOTE:** This state is not supported for IA64 processors. They should instead use C1.
G1/S1	**Stop-Grant:** Similar to G0/S0/C2 state. The Intel® 6300ESB ICH also has the option to assert the CPUSLP# signal to further reduce processor power consumption. **NOTE:** The behavior for this state is slightly different when supporting IA64 processors.
G1/S3	**Suspend-To-RAM (STR):** The system context is maintained in system DRAM, but power is shut off to non-critical circuits. Memory is retained, and refreshes continue. All clocks stop except RTC clock.
G1/S4	**Suspend-To-Disk (STD):** The context of the system is maintained on the disk. All power is then shut off to the system except for the logic required to resume.
G2/S5	**Soft Off (SOFF):** System context is not maintained. All power is shut off except for the logic required to restart. A full boot is required when waking.
G3	**Mechanical OFF (MOFF):** System context not maintained. All power is shut off except for the RTC. No “Wake” events are possible, because the system does not have any power. This state occurs when the user removes the batteries, turns off a mechanical switch, or when the system power supply is at a level that is insufficient to power the “waking” logic. When system power returns, transition will depends on the state just prior to the entry to G3 and the AFTERG3 bit in the GEN_PMCON3 register (D31:F0, offset A4). Refer to Table 71 for more details.

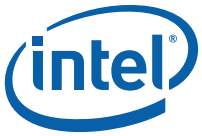


Table 63 shows the transitions rules among the various states. Note that transitions among the various states may appear to temporarily transition through intermediate states. For example, in going from S0 to S1, it may appear to pass through the G0/S0/C2 states. These intermediate transitions and states are not listed in the table.

Table 63. State Transition Rules for Intel® 6300ESB I/O Controller Hub

Present State	Transition Trigger	Next State
G0/S0/C0	• Processor halt instruction • Level 2 Read • SLP_EN bit set • Power Button Override • Mechanical Off/Power Failure	• G0/S0/C1 • G0/S0/C2 • G1/Sx or G2/S5state • G2/S5 • G3
G0/S0/C1	• Any Enabled break event • STPCLK# goes active • Power Button Override • Power Failure	• G0/S0/C0 • G0/S0/C2 • G2/S5 • G3
G0/S0/C2	• Any Enabled break event • STPCLK# goes inactive and previously in C1 • Power Button Override • Power Failure	• G0/S0/C0 • G0/S0/C1 • G2/S5 • G3
G1/S1, G1/S3, or G1/S4	• Any Enabled Wake Event • Power Button Override • Power Failure	• G0/S0/C0 • G2/S5 • G3
G2/S5	• Any Enabled Wake Event • Power Failure	• G0/S0/C0 • G3
G3	• Power Returns	• Optional to go to S0/C0 (reboot) or G2/S5 (stay off until power button pressed or other wake event). **NOTE:** Some wake events may be preserved through power failure.

5.11.3 System Power Planes

The system has several independent power planes. These power planes can be shut off and the voltage set to a zero volt level. The power planes and their control signals are listed in Table 64.

Table 64. System Power Plane

Plane	Controlled By	Description
Processor	SLP_S3# signal	The SLP_S3# signal may be used to cut the processor's power completely.
MAIN	SLP_S3# signal	When SLP_S3# goes active, power may be shut off to any circuit not required to wake the system from the S3 state. Since the S3 state requires that the memory context be preserved, power must be retained to the main memory. The processor, devices on the PCI bus, LPC I/F downstream Hub Interface and AGP will typically be shut off when the Main power plane is shut off, although there may be small subsections powered.
MEMORY	SLP_S4# signal	When the SLP_S4# goes active, power may be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory may also be shut down.
MEMORY	SLP_S5# signal	When the SLP_S5# goes active, power may be shut off to any circuit not required to wake the system from the S5. Since the memory context does not need to be preserved in the S5 state, the power to the memory may also be shut down.
DEVICE[n]	GPIO	Individual subsystems may have their own power plane. For example, GPIO signals may be used to control the power to disk drives, audio amplifiers, or the display screen.

5.11.4 Intel® 6300ESB ICH Power Planes

The Intel® 6300ESB ICH power planes were previously defined in Section 4.1, "Power Planes".

Although not specific power planes within the Intel® 6300ESB ICH, there are many interface signals that go to devices that may be powered down. These include:

- IDE: Output signals may be tri-stated or driven low and all input buffers may be shut off
- USB: Output signals may be tri-stated and all input buffers may by shut off when USB wakeup is not desired.
- AC'97: Output signals may be driven low and input buffers may be shut off.

5.11.5 SMI#/SCI Generation

Upon any SMI# event, the Intel® 6300ESB ICH will assert SMI# to the processor, which will cause it to enter SMM space. SMI# remains active until the EOS bit is set. When the EOS bit (bit 1) is set, SMI# will go inactive for a minimum of four PCI clocks. See Section 8.8.3.9, "SMI_EN—SMI Control and Enable Register" for details on the SMI control register.

The SCI is a level-mode interrupt that is typically handled by an ACPI-aware operating system. In non-APIC systems (which is the default), the SCI IRQ is routed to one of the 8259 interrupts (IRQ9, 10, or 11). The 8259 interrupt controller must be programmed to level mode for that interrupt.

In systems using the APIC, the SCI may be routed to interrupts 9, 10, 11, 20, 21, 22, or 23 The interrupt polarity changes depending on whether it is on an interrupt shareable with a PIRQ or not. The interrupt will remain asserted until all SCI sources are removed.

Table 32 shows which events may cause an SCI, and Table 65 shows the causes of an SMI#. Note that some events may be programmed to cause either an SMI# or SCI. The usage of the event for SCI (instead of SMI#) is typically associated with an ACPI-based system.

Causes of TCO SCI are discussed in Section 5.12.3, "TCO Theory of Operation".

Table 65. Causes of SCI

Cause	Additional Enables[1]	Where Reported
PME#	PME_EN = 1	PME_STS
Internal EHCI wake (PME_B0)	PME_B0_EN = 1	PME_B0_STS
Power Button Press	PWRBTN_EN = 1	PWRBTN_STS
RTC Alarm	RTC_EN = 1	RTC_STS
Ring Indicate	RI_EN = 1	RI_STS
AC'97 wakes	AC97_EN = 1	AC97_STS
USB #1 wakes	USB1_EN = 1	USB1_STS
USB #2 wakes	USB2_EN = 1	USB2_STS
USB #3 wakes	USB3_EN = 1	USB3_STS
THRM# pin active (based on THRM#_POL)	THRM_EN = 1	THRM_STS
ACPI Timer overflow (2.34 seconds)	TMROF_EN = 1	TMROF_STS
Any GPI	GPI[x]_Route = 10, GPE[x]_EN = 1	GPI[x]_STS
TCO SCI Logic[1, 2]	TCOSCI_EN = 1	TCOSCI_STS
BIOS_RLS written to 1	GBL_EN = 1	GBL_STS
WDT 1st timeout	WDT_ENABLE = 1, WDT_INT_TYPE = "01"	WDTSCI_STS

NOTES:
1. SCI_EN must be 1 to enable SCI.
2. SCI may be routed to cause Interrupt 9:11 or 20:23 (20:23 only available in APIC mode).

Table 66. Causes of SMI# (Sheet 1 of 2)

Cause	Additional Enables	Where Reported	Synch
PME#	SCI_EN = 0, PME_EN = 1	PME_STS	
Internal EHCI wake (PME_B0)	SCI_EN = 0, PME_B0_EN = 1	PME_B0_STS	
Power Button Press	SCI_EN = 0, PWRBTN_EN = 1	PWRBTN_STS	
RTC Alarm	SCI_EN = 0, RTC_EN = 1	RTC_STS	
Ring Indicate	SCI_EN = 0, RI_EN = 1	RI_STS	
AC'97 wakes	SCI_EN = 0, AC97_EN = 1	AC97_STS	
USB #1 wakes	SCI_EN = 0, USB1_EN = 1	USB1_STS	
USB #2 wakes	SCI_EN = 0, USB2_EN = 1	USB2_STS	
USB #3 wakes	SCI_EN = 0, USB3_EN = 1	USB3_STS	
THRM# pin active (based on THRM#_POL)	SCI_EN = 0, THRM_EN = 1	THRM_STS	
ACPI Timer overflow (2.34 seconds)	SCI_EN = 0, TMROF_EN = 1	TMROF_STS	
Any GPI	GPI[x]_Route = 01, GPE[x]_EN = 1	GPE1_STS, GPI[x]_STS	
TCO SMI Logic (see table below)	TCO_EN = 1	TCO_STS	
NMI occurred (and NMIs mapped to SMI) See NMI section for causes of NMI	NMI2SMI_EN = 1	TCO_STS, NMI2SMI_STS	
GBL_RLS written to 1	BIOS_EN = 1	BIOS_STS	
Write to B2h register	None	APM_STS	X
Periodic timer expires	PERIODIC_EN = 1	PERIODIC_STS	
64 ms timer expires	SWSMI_TMR_EN = 1	SWSMI_TMR_STS	
Enhanced USB Legacy Support Event	LEGACY_USB2_EN = 1	LEGACY_USB2_STS	
Enhanced USB Intel-Specific Event	INTEL_USB2_EN = 1	INTEL_USB2_STS	
Classic USB Legacy logic (Port 64/60 R/W, End of pass through)	LEGACY_USB_EN = 1	LEGACY_USB_STS	X
Classic USB Legacy logic (IRQ)	LEGACY_USB_EN = 1	LEGACY_USB_STS	
Serial IRQ SMI Reported	None	SERIRQ_SMI_STS	

NOTES:
1. GBL_SMI_EN must be 1 to enable SMI.
2. EOS must be written to 1 to re-enable SMI for the next one.
3. Some SMI#s are considered "synchronous", in that the processor should recognize the SMI# prior to completing the instruction (I/O read, I/O write, Memory read, or Memory write) that should cause the SMI#. This is accomplished by having the SMI# signal go active to the processor prior to the processor observing the RDY# signal that terminates the cycle. SMI#s marked with X in the Synch column are treated as Synchronous.Synchronous SMI#s are not possible in IA64 platforms, since they do not support the SMI# signal.
4. NMI2SMI_STS is not gated by TCO_EN.

Table 66. Causes of SMI# (Sheet 2 of 2)

Cause	Additional Enables	Where Reported	Synch
Device Monitors matches an address in its range	DEV[n]_TRAP_EN = 1	DEVMON_STS, DEV[n]_TRAP_STS	X
SMBus Host Controller	SMB_SMI_EN Host Controller enabled	Various bits in the SMBus Host Status Register	
SMBus Slave SMI message	None	SMBUS_SMI_STS	
SMBus SMBALERT# signal active	None	SMBUS_SMI_STS	
SMBus Host Notify message received	HOST_NOTIFY_INTREN	SMBUS_SMI_STS, HOST_NOTIFY_STS	
Access to Microcontroller Range (62h/66h)	MCSMI_EN	MCSMI_STS	X
SLP_EN bit written to 1	SMI_ON_SLP_EN = 1	SMI_ON_SLP_EN_STS	X
WDT 1st timeout	WDT_ENABLE = 1, WDT_INT_TYPE = '10'	WDT_SMI_STS	

NOTES:
1. GBL_SMI_EN must be 1 to enable SMI.
2. EOS must be written to 1 to re-enable SMI for the next one.
3. Some SMI#s are considered "synchronous", in that the processor should recognize the SMI# prior to completing the instruction (I/O read, I/O write, Memory read, or Memory write) that should cause the SMI#. This is accomplished by having the SMI# signal go active to the processor prior to the processor observing the RDY# signal that terminates the cycle. SMI#s marked with X in the Synch column are treated as Synchronous.Synchronous SMI#s are not possible in IA64 platforms, since they do not support the SMI# signal.
4. NMI2SMI_STS is not gated by TCO_EN.

Table 67. Causes of TCO SMI#

Cause	Additional Enables	Where Reported
Century Rollover	None	NEWCENTURY_STS
TCO TIMEROUT	None	TIMEOUT
OS writes to TCO_DAT_IN register	None	OS_TCO_SMI
Message from MCH	None	MCHSMI_STS
NMI occurred (and NMIs mapped to SMI) See NMI section for causes of NMI	NMI2SMI_EN = 1	NMI2SMI_STS
NOTE: NMI2SMI_STS is not gated by TCO_EN. See table above.		
INTRUDER# signal goes active	INTRD_SEL = 10	INTRD_DET
Changes of the BIOSWP bit from 0 to 1	BLD = 1	BIOSWR_STS
Write attempted to BIOS	BIOSWP = 1	BIOSWR_STS

See Section 5.12.3, "TCO Theory of Operation" for details on the TCO SMI# causes.

5.11.6 Dynamic Processor Clock Control

The Intel® 6300ESB ICH has extensive control for dynamically starting and stopping system clocks. The clock control is used for transitions among the various S0/Cx states, and processor throttling. Each dynamic clock control method is described in this section. The various Sleep states may also perform types of non-dynamic clock control.

The Intel® 6300ESB ICH supports the ACPI C0, C1 and C2 states. C3 and C4 are not supported.

The Dynamic Processor Clock control is handled using the following signal:

- STPCLK#: Used to halt processor instruction stream.

Note: The Intel® 6300ESB ICH does support THRM# based throttling. The C1 state (processor auto halt) may be used with either one or two processors, however. Processors are free to perform their own dynamic clock control; however, this is done without any coordination by the Intel® 6300ESB ICH.

The C1 state is entered based on the processor performing an auto halt instruction.

The C2 state is entered based on the processor reading the Level 2 register in the Intel® 6300ESB ICH.

A C1 or C2 state ends due to a break event. Based on the break event, the Intel® 6300ESB ICH returns the system to C0 state. Table 68 lists the possible break events from C2 states. The break events from C1 are indicated in the processor's datasheet.

Table 68. Break Events

Event	Breaks from	Comment
Any unmasked interrupt goes active	C2	IRQ[0:15] when using the 8259s, IRQ[0:23] for I/O xAPIC0 and I/O xA[IC1. Since SCI is an interrupt, any SCI will also be a break event.
Any internal event that will cause an NMI or SMI#	C2	Many possible sources.
Any internal event that will cause INIT# to go active	C2	Could be indicated by the keyboard controller through the RCIN input signal.
Processor Pending break event Indication	C2	Only available when FERR# enabled for break event indication (See FERR# Mux-En in Section 8.1.22, "Offset D0h - D3h: GEN_CNTL—General Control Register (LPC I/F—D31:F0)").

The Intel® 6300ESB ICH supports the Pending Break Event (PBE) indication from the processor using the FERR# signal. The following rules apply:

1. When STPCLK# is detected active by the processor, the FERR# signal from the processor will be redefined to indicate whether an interrupt is pending. The signal is active low (i.e., FERR# will be low to indicate a pending interrupt).
2. When the Intel® 6300ESB ICH asserts STPCLK#, it will latch the current state of the FERR# signal and continue to present this state to the FERR# state machine (independent of what the FERR# pin does after the latching).
3. When the Intel® 6300ESB ICH detects the Stop-Grant cycle, it will start looking at the FERR# signal as a break event indication. When FERR# is sampled low, a break event is indicated. This will force a transition to the C0 state.
4. When the processor detects the deassertion of STPCLK#, the processor will start driving the FERR# signal with the natural value (i.e.the value it would do when the pin was not muxed). The time from STPCLK# inactive to the FERR# signal transition back to the native function must be less than 120 ns.

5. The Intel® 6300ESB ICH waits at least 180 ns to 8 PCI clocks (240 ns) after deasserting STPCLK# and then starts using the FERR# signal for an indication of a floating point error. The maximum time that the Intel® 6300ESB ICH may wait is bounded such that it must have a chance to look at the FERR# signal before reasserting STPCLK#. Based on current implementation, that maximum time would be 240 ns (8 PCI clocks). Since the processor has 120-210 ns to revert to the proper FERR# function, there are 60-30 ns of margin inherent in the timings.

The break event associated with this new mechanism does not need to set any particular status bit, since the pending interrupt will be serviced by the processor after returning to the C0 state.

5.11.6.1 Throttling Using STPCLK#

Throttling is used to lower power consumption or reduce heat. The Intel® 6300ESB ICH asserts STPCLK# to throttle the processor clock and the processor appears to temporarily enter a C2 state. After a programmable time, the Intel® 6300ESB ICH deasserts STPCLK# and the processor appears to return to the C0 state. This allows the processor to operate at reduced average power, with a corresponding decrease in performance. Two methods are included to start throttling:

1. Software enables a timer with a programmable duty cycle. The duty cycle is set by the THTL_DTY field and the throttling is enabled using the THTL_EN field. This is known as Manual Throttling. The period is fixed to be in the non-audible range, due to the nature of switching power supplies.
2. A Thermal Override condition (THRM# signal active for >2 seconds) occurs that unconditionally forces throttling, independent of the THTL_EN bit. The throttling due to Thermal Override has a separate duty cycle (THRM_DTY) which may vary by field and system. The Thermal Override condition will end when THRM# goes inactive.

Throttling due to the THRM# signal has higher priority than the software initiated throttling.

Throttling does not occur when the system is in a C2 state, even when Thermal override occurs.

5.11.6.2 Transition Rules among S0/Cx and Throttling States

The following priority rules and assumptions apply among the various S0/Cx and throttling states:

- Entry to any S0/Cx state is mutually exclusive with entry to any S1–S5 state. This is because the processor may only perform one register access at a time and Sleep states have higher priority than thermal throttling.
- When the SLP_EN bit is set (system going to a sleep state (S1–S5), the THTL_EN bit may be internally treated as being disabled (no throttling while going to sleep state). Note that thermal throttling (based on THRM# signal) cannot be disabled in an S0 state. However, once the SLP_EN bit is set, the thermal throttling is shut off (since STPCLK# will be active in S1–S5 states).
- When the THTL_EN bit is set, and a Level 2 read then occurs, the system should immediately go and stay in a C2 state until a break event occurs. A Level 2 read has higher priority than the software initiated throttling or thermal throttling.
- When Thermal Override is causing throttling, and a Level 2 read then occurs, the system will stay in a C2 state until a break event occurs. A Level 2 read has higher priority than the Thermal Override.
- After an exit from a C2 state (due to a break event), and when the THTL_EN bit is still set, or when a Thermal Override is still occurring, the system will continue to throttle STPCLK#. Depending on the time of break event, the first transition on STPCLK# active may be delayed by up to one THRM period (1024 PCI clocks=30.72 microseconds).

- The Host controller must post Stop-Grant cycles in such a way that the processor gets an indication of the end of the special cycle prior to the Intel® 6300ESB ICH observing the Stop-Grant cycle. This ensures that the STPCLK# signals stays active for a sufficient period after the processor observes the response phase.
- When in the C1 state and the STPCLK# signal goes active, the processor will generate a Stop-Grant cycle, and the system should go to the C2 state. When STPCLK# goes inactive, it should return to the C1 state.

5.11.6.3 STPCLK# Implementation Notes

The processor treats STPCLK# like an interrupt and recognizes it on instruction boundaries (no INTA cycles are run). When it recognizes STPCLK# active, the processor stops execution on the next instruction boundary, stops the pre-fetch unit, empties internal pipelines and write buffers, and generates a Stop-Grant bus cycle before entering the Stop Grant state. The processor may stop the clock to most of its internal modules, and no instructions are executed. The processor exits the Stop Grant state when it is reset, or upon sampling STPCLK# inactive.

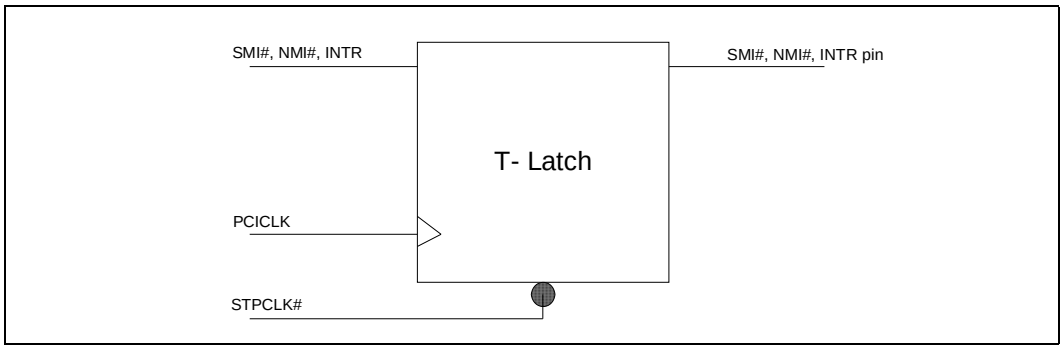
The processor will latch transitions on the external interrupt signals (SMI#, NMI, INTR, and INIT#) while in Stop Grant state. These interrupts are taken after the deassertion of STPCLK#.

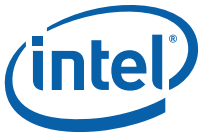
For the Intel® Pentium® 4 processor, the following edge signals must not transition while STPCLK# is active:

- INTR
- INIT#
- SMI#
- NMI

These signals should be run through a transparent latch internal to the Intel® 6300ESB ICH. While STPCLK# is inactive (HIGH) these signals propagate to the Intel® 6300ESB ICH's pins and onto the processor as normal. However when STPCLK# is asserted then these signals are latched so that they may not change until STPCLK# is deasserted. This ensures that an edge on these signals is seen while the processor has a valid clock. These signals need to be latched at least 1 HCLK clock before STPCLK# assertion, and held 16 HCLK clocks after STPCLK# deassertion.

Figure 15. Latching Processor I/F Signals with STOPCLK#





Other Implementation Notes:

- When STPCLK# goes active due to a Level read, it must go active prior to the completion of the associated I/O read. This is to ensure that the STPCLK# is recognized by the processor prior to it recognizing the end of the I/O cycle. That will prevent the next instruction from being executed.
- The state machine must insure that the STPCLK# signal stays high for a minimum period of time. When STPCLK# is to go low due to throttling (regular or due to the THRM# signal), this could be very soon after it was driven high. The MCH should ensure that the Stop-Grant cycle coming down the Hub Interface occurs after the BRDY# is seen by the processor.

Exception: For SMI#s that are caused by a processor I/O cycle, when STPCLK# is active, the Intel® 6300ESB ICH will still drive SMI# active. This is because the STPCLK# was obviously too late to be recognized at the instruction boundary. The I/O cycles that may cause SMI# include: writes to the APM register (B2h), accesses to 60/64h when "Legacy USB KBC scheme" is used, traps for Monitors 4, 5, 6, and 7, the SMI# on SLP_EN bit, accesses to 62/66h when the MCSMI_EN bit is set, access to registers with their associated enable set in the DEVTRAP_EN register, and the BIOS_STS bit (which is set by the processor writing a 1 to the GLB_RLS bit when the BIOS_EN bit is also set).

5.11.7 Sleep States

5.11.7.1 Sleep State Overview

The Intel® 6300ESB ICH directly supports different sleep states (S1–S5), which are entered by setting the SLP_EN bit, or due to a Power Button press. The entry to the Sleep states are based on several assumptions:

- Entry to a Cx state is mutually exclusive with entry to a Sleep state. This is because the processor may only perform one register access at a time. A request to Sleep always has higher priority than throttling.
- Prior to setting the SLP_EN bit, the software will turn off processor-controlled throttling. Note that thermal throttling cannot be disabled, but setting the SLP_EN bit will disable thermal throttling (since S1–S5 sleep state has higher priority).
- The G3 state cannot be entered through any software mechanism. The G3 state indicates a complete loss of power.

5.11.7.2 Initiating Sleep State

Sleep states (S1–S5) are initiated by:

- Masking interrupts, turning off all bus master enable bits, setting the desired type in the SLP_TYP field and then setting the SLP_EN bit. The hardware will then attempt to gracefully put the system into the corresponding Sleep state by first going to a C2 state. See Section 5.11.6, "Dynamic Processor Clock Control" for details on going to the C2 state.
- Pressing the PWRBTN# Signal for more than four seconds to cause a Power Button Override event. In this case the transition to the S5 state will be less graceful, since there will be no dependencies on observing Stop-Grant cycles from the processor or on clocks other than the RTC clock.

Table 69. Sleep Types

Sleep Type	Comment
S1	The Intel® 6300ESB ICH asserts the STPCLK# signal. It also has the option to assert CPUSLP# signal. This will lower the processor's power consumption. No snooping is possible in this state.
S3	The Intel® 6300ESB ICH asserts SLP_S3#. The SLP_S3# signal will control the power to non-critical circuits. Power will only be retained to devices needed to wake from this sleeping state, as well as to the memory.
S4	The Intel® 6300ESB ICH asserts SLP_S3# and SLP_S4#. The SLP_S4# signal will shut off the power to the memory subsystem. Only devices needed to wake from this state should be powered.
S5	Same power state as S4. The Intel® 6300ESB ICH asserts SLP_S3#, SLP_S4# and SLP_S5#.

Other Assumptions:

- Entry to a Cx state is mutually exclusive with entry to a Sleep state. This is because the processor may only perform one register access at a time. A request to Sleep always has higher priority than throttling.
- Prior to setting the SLP_EN bit, the software will turn off processor-controlled throttling. Note that thermal throttling cannot be disabled, but setting the SLP_EN bit will disable throttling (since S1-S5 sleep state has higher priority).
- The G3 state cannot be entered through any software mechanism. The G3 state indicates a complete loss of power.

Before entering sleep state, an ACPI OS will mask all interrupts and will turn off all bus master enable bits. For non-ACPI systems, the BIOS will mask interrupts and turn off all bus master enable bits.

Note: Interrupts might not be masked at the I/O subsystem. Some Operating Systems have been observed to only mask interrupts inside the processor.

5.11.7.3 Exiting Sleep States

Sleep states (S1–S5) are exited based on Wake events. The Wake events will force the system to a full on state (S0), although some non-critical subsystems might still be shut off and have to be brought back manually. For example, the hard disk may be shut off during a sleep state, and have to be enabled through a GPIO pin before it may be used.

Upon exit from the Intel® 6300ESB ICH-controlled Sleep states, the WAK_STS bit will be set.

The possible causes of Wake Events (and their restrictions) are shown in Table 70.

Table 70. Causes of Wake Events (Sheet 1 of 2)

Cause	States Can Wake From	How Enabled
RTC Alarm	S1–S5	Set RTC_EN bit in PM1_EN Register

NOTE: When in the S5 state due to a powerbutton override, the only wake events are Power Button, Wake SMBUs Slave Message (01h), and Hard Reset SMBus Slave Messages (03h, 04h).

NOTE: PME#, RTC, GPI[0:n], and RI# will be wake events from S5 only when it was entered through software setting the SLP_EN and SLP_TYP bits, or if there is a power failure.

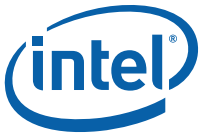


Table 70. Causes of Wake Events (Sheet 2 of 2)

Cause	States Can Wake From	How Enabled
Power Button	S1–S5	Always enabled as Wake event
GPI[0:n]	S1–S5	GPE0_EN register (after having gone to S5 through SLP_EN, but not after a power failure.) **NOTE:** GPIs that are in the core well are not capable of waking the system from sleep states where the core well is not powered.
USB	S1–S5	Set USB1_EN, USB 2_EN and USB3_EN bits in GPE0_EN Register
RI#	S1–S5	Set RI_EN bit in GPE0_EN Register
AC'97	S1–S5	Set AC'97_EN bit in GPE0_EN Register
Secondary PME#	S1–S5	Set PME_EN bit in GPE0_EN Register.
SMBALERT#	S1–S4	SMB_WAK_EN in the GPE0 Register. Always enabled as Wake event
SMBus Slave Message	S1–S5	Wake/SMI# command always enabled as a Wake Event. **NOTE:** SMBus Slave Message may wake the system from S1-S5, as well as from S5 due to Power Button Override.
PME_B0 (internal USB EHCI controller)	S1–S5	Set PME_B0_EN bit in GPE0_EN Register.

NOTE: When in the S5 state due to a powerbutton override, the only wake events are Power Button, Wake SMBUs Slave Message (01h), and Hard Reset SMBus Slave Messages (03h, 04h).

NOTE: PME#, RTC, GPI[0:n], and RI# will be wake events from S5 only when it was entered through software setting the SLP_EN and SLP_TYP bits, or if there is a power failure.

5.11.7.4 Sx-G3-Sx, Handling Power Failures

In desktop systems, power failures may occur when the AC power is cut (a real power failure) or when the system is unplugged. In either case, PWROK and RSMRST# are assumed to go low. Depending on when the power failure occurs and how the system is designed, different transitions could occur due to a power failure.

The AFTER_G3 bit provides the ability to program whether or not the system should boot once power returns after a power loss event. When the policy is to not boot, the system will remain in an S5 state (unless previously in S4). There are only three possible events that will wake the system after a power failure.

1. **PWRBTN#:** PWRBTN# is always enabled as a wake event. When RSMRST# is low (G3 state), the PWRBTN_STS bit is reset. When the Intel® 6300ESB ICH exits G3 after power returns (RSMRST# goes high), the PWRBTN# signal is already high (because V_{CC}-standby goes high before RSMRST# goes high) and the PWRBTN_STS bit is 0.
2. **RI#:** RI# does not have an internal pull-up. Therefore, when this signal is enabled as a wake event, it is important to keep this signal powered during the power loss event. When this signal goes low (active), when power returns the RI_STS bit will be set and the system will interpret that as a wake event.
3. **RTC Alarm:** The RTC_EN bit is in the RTC well and is preserved after a power loss. Like PWRBTN_STS the RTC_STS bit is cleared when RSMRST# goes low.

The Intel® 6300ESB ICH monitors both PWROK and RSMRST# to detect for power failures. When PWROK goes low, the PWROK_FLR bit is set. When RSMRST# goes low, PWR_FLR is set.

Table 71. Transitions Due to Power Failure

State at Power Failure	AFTERG3_EN bit	Transition When Power Returns
S0, S1, S3	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0

5.11.8 Thermal Management

The Intel® 6300ESB ICH has mechanisms to assist with managing thermal problems in the system.

5.11.8.1 THRM# Signal

The THRM# signal is used as a status input for a thermal sensor. Based on the THRM# signal going active, the Intel® 6300ESB ICH generates an SMI# or SCI (depending on SCI_EN).

When the THRM_POL bit is set low, when the THRM# signal goes low, the THRM_STS bit will be set. This is an indicator that the thermal threshold has been exceeded. When the THRM_EN bit is set, then when THRM_STS goes active, either an SMI# or SCI will be generated (depending on the SCI_EN bit being set).

The power management software (BIOS or ACPI) may then take measures to start reducing the temperature. Examples include shutting off unwanted subsystems, or halting the processor.

By setting the THRM_POL bit to high, another SMI# or SCI may optionally be generated when the THRM# signal goes back high. This allows the software (BIOS or ACPI) to turn off the cooling methods.

Note: THRM# assertion will not cause a TCO event message in S1-M, S3, or S4. The level of the signal will not be reported in the heartbeat message.

5.11.8.2 THRM# Initiated Passive Cooling

When the THRM# signal remains active for some time greater than two seconds and the Intel® 6300ESB ICH is in the S0/G0/C0 state, then the Intel® 6300ESB ICH enters an auto-throttling mode, in which it provides a duty cycle on the STPCLK# signal. This will reduce the overall power consumption by the system, and should cool the system. The intended result of the cooling is that the THRM# signal should go back inactive.

For all programmed values (001–111), THRM# going active will result in STPCLK# active for a minimum time of 12.5% and a maximum of 87.5%. The period is 1024 PCI clocks. Thus, the STPCLK# signal may be active for as little as 128 PCI clocks or as much as 896 PCI clocks. The actual slowdown (and cooling) of the processor will depend on the instruction stream, because the processor is allowed to finish the current instruction. Furthermore, the Intel® 6300ESB ICH waits for the STOP-GRANT cycle before starting the count of the time the STPCLK# signal is active.

When THRM# goes inactive, the throttling will stop.

Note: There is a small window where the Intel® 6300ESB ICH may assert STPCLK# for one more throttling period after THRM# goes inactive. This is due to a sampling delay on THRM# (the signal is still active internally, but has just gone inactive externally).

In case that the Intel® 6300ESB ICH is already attempting throttling because the THTL_EN bit is set, the duty cycle associated with the THRM# signal will have higher priority.

When the Intel® 6300ESB ICH is in the C2, or S1–S5 states, then no throttling will be caused by the THRM# signal being active.

5.11.8.3 THRM# Override Software Bit

The FORCE_THTL bit allows the BIOS to force passive cooling, just as though the THRM# signal had been active for two seconds. When this bit is set, the Intel® 6300ESB ICH will start throttling using the ratio in the THRM_DTY field.

When this bit is cleared the Intel® 6300ESB ICH will stop throttling, unless the THRM# signal has been active for two seconds or when the THTL_EN bit is set (indicating that ACPI software is attempting throttling).

5.11.8.4 Processor Initiated Passive Cooling (Via Programmed Duty Cycle on STPCLK#)

Using the THTL_EN and THTL_DTY bits, the Intel® 6300ESB ICH may force a programmed duty cycle on the STPCLK# signal. This will reduce the effective instruction rate of the processor and cut its power consumption and heat generation. See Section 8.8.3.5, “PROC_CNT—Processor Control Register” for more details on the programming of these bits.

5.11.8.5 Active Cooling

Active cooling involves fans. The GPIO signals from the Intel® 6300ESB ICH may be used to turn on/off a fan.

5.11.9 Event Input Signal Usage

The Intel® 6300ESB ICH has various input signals that trigger specific events. This section describes those signals and how they should be used.

5.11.9.1 PWRBTN# - Power Button

The Intel® 6300ESB ICH PWRBTN# signal operates as a “Fixed Power Button” as described in the ACPI specification. PWRBTN# signal has a 16 ms debounce on the input. The state transition descriptions are included in the following table.

Note: Transitions start as soon as the PWRBTN# is pressed (but after the debounce logic) and do not depend on when the Power Button is released.

Table 72. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI# or SCI generated (depending on SCI_EN)	Software will typically initiate a Sleep state.
S1-S5	PWRBTN# goes low	Wake Event. Transitions to S0 state.	Standard wakeup
G3	PWRBTN# pressed	None	No effect since no power. Not latched nor detected.
S0-S4	PWRBTN# held low for at least four consecutive seconds	Unconditional transition to S5 state.	No dependence on processor (such as Stop-Grant cycles) or any other subsystem.

Power Button Override Function

When PWRBTN# is observed active for at least four consecutive seconds, then the state machine should unconditionally transition to the G2/S5 state, regardless of present state (S0–S4). In this case, the transition to the G2/S5 state should not depend on any particular response from the processor (such as a Stop-Grant cycle), nor any similar dependency from any other subsystem.

A power button override will force a transition to S5 even when PWROK is not active.

The PWRBTN# status is readable to check when the button is currently being pressed or has been released. The status is taken after the debounce, and is readable through the PWRBTN_LVL bit.

Note: The four second PWRBTN# assertion should only be used when a system lock-up has occurred. The four second timer starts counting when the Intel® 6300ESB ICH is in a S0 state. When the PWRBTN# signal is asserted and held active when the system is in a suspend state (S1–S5), the assertion will cause a wake event. Once the system has resumed to the S0 state, the four second timer will start.

Sleep Button

The ACPI specification defines an optional Sleep button. It differs from the power button in that it only is a request to go from S0 to S1–S4 (not S5). Also, in an S5 state, the Power Button may wake the system, but the Sleep Button cannot.

Although the Intel® 6300ESB ICH does not include a specific signal designated as a Sleep Button, one of the GPIO signals may be used to create a “Control Method” Sleep Button. See the ACPI specification for implementation details.

5.11.9.2 RI# - Ring Indicate

The Ring Indicator may cause a wake event (when enabled) from the S1–S5 states. Table 73 shows when the wake event is generated or ignored in different states. When in the G0/S0/Cx states, the Intel® 6300ESB ICH will generate an interrupt based on RI# active, and the interrupt will be set up as a break event.

Table 73. Transitions Due to RI# Signal

Present State	Event	RI_EN	Event
S0	RI# Active	X	Ignored
S1-S5	RI# Active	0	Ignored
		1	Wake Event

NOTE: Filtering/Debounce on RI# will not be done in the Intel® 6300ESB ICH. Can be in modem or external.

5.11.9.3 PME# - PCI Power Management Event

The PME# signal comes from a PCI device to request that the system be restarted. The PME# signal may generate an SMI#, SCI, or optionally a Wake event. The event occurs when the PME# signal goes from high to low. No event is caused when it goes from low to high. The Intel® 6300ESB ICH supports only one PME# signal which is shared between both the PCI and the PCI-X interfaces.

In the EHCI controller, there is an internal PME_B0 bit. This is separate from the external PME# signal and may cause the same effect.

5.11.9.4 SYS_RESET# Signal

SYS_RESET# is a new pin on the Intel® 6300ESB ICH that is used to eliminate extra glue logic on the board. Before the addition of this pin, a system reset was activated by external glue forcing the PWROK signal low after the reset button was pressed. This pin eliminates the need for that glue. As such, a SYS_RESET# event should look internally to our chip and externally to the system as when PWROK had gone low.

When the SYS_RESET# pin is detected as active after the 16 ms debounce logic, the Intel® 6300ESB ICH will attempt to perform a “graceful” reset, by waiting up to 25 ms for the SMBus to go idle. When the SMBus is idle when the pin is detected active, the reset will occur immediately, otherwise the counter will start. When at any point during the count the SMBus goes idle the reset will occur. When, however, the counter expires and the SMBus is still active, a reset will be forced upon the system even though activity is still occurring.

Once a reset of this type has occurred, it cannot occur again until SYS_RESET# has been detected inactive after the debounce logic, and the system is back to a full S0 state as indicated by all of the PWROK inputs being active.

5.11.9.5 THRMTRIP# Signal

When THRMTRIP# goes active, the processor is indicating an overheat condition, and the Intel® 6300ESB ICH will immediately transition to an S5 state. However, since the processor has overheated, it will not respond to the Intel® 6300ESB ICH’s STPCLK# pin with a stop grant special cycle. Therefore, the Intel® 6300ESB ICH will not wait for one. Immediately upon seeing THRMTRIP# low, the Intel® 6300ESB ICH will initiate a transition to the S5 state, drive SLP_S3#, SLP_S4#, SLP_S5# low, and set the CTS bit. The transition will look like a power button override.

It is extremely important that when a THRMTRIP# event occurs, the Intel® 6300ESB ICH power down immediately without following the normal S0 -> S5 path. This path may be taken in parallel, but the Intel® 6300ESB ICH must immediately enter a power down state. It will do this by driving SLP_S3#, SLP_S4#, and SLP_S5# immediately after sampling THRMTRIP# active.

When the processor is running extremely hot and is heating up, it is possible (although very unlikely) that components around it, such as the Intel® 6300ESB ICH, are no longer executing cycles properly. Therefore, when THRMTRIP# fires and the Intel® 6300ESB ICH is relying on state machine logic to perform the power down, the state machine may not be working and the system will not power down.

The Intel® 6300ESB ICH will follow this flow for THRMTRIP#.

1. At boot (PXPCIRST# low), THRMTRIP# ignored.
2. After power-up (PXPCIRST# high), when THRMTRIP# sampled active, SLP_S3#, SLP_S4#, and SLP_S5# fire, and normal sequence of sleep machine starts.
3. Until sleep machine enters the S5 state, SLP_S3#, SLP_S4#, and SLP_S5# stay active, even when THRMTRIP# is now inactive. This is the equivalent of “latching” the thermal trip event.
4. When S5 state reached, go to step #1, otherwise stay here. When the Intel® 6300ESB ICH never reaches S5, the Intel® 6300ESB ICH will not reboot until power is cycled.

A Processor Thermal trip event will:

- Set the AFTERG3_EN bit
- Clear the PWRBTN_STS bit
- Clear all the GPE0_EN and GPE1_EN register bits
- Clear the SMB_WAK_STS bit only when SMB_WAK_STS was set due to SMBus slave receiving message and not set due to SMBAlert.

Note: The THRMTRIP# pin must be glitch free.

5.11.10 ALT Access Mode

Before entering a low power state, several registers from powered down parts may need to be saved. In the majority of cases, this is not an issue, as registers have read and write paths. However, several of the ISA compatible registers are either read only or write only. To get data out of write-only registers, and to restore data into read-only registers, the Intel® 6300ESB ICH implements an ALT access mode.

When the ALT access mode is entered and exited after reading the registers of the Intel® 6300ESB ICH timer (8254), the timer starts counting faster (13.5 ms). The following steps listed below may cause problems:

1. BIOS enters ALT access mode for reading the Intel® 6300ESB ICH timer related registers.
2. BIOS exits ALT access mode.
3. BIOS continues through the execution of other needed steps and passes control to the OS.

After getting control in step #3, when the OS does not reprogram the system timer again the timer ticks may be happening faster than expected. For example DOS and its associated software assume that the system timer is running at 54.6 ms and as a result the timeouts in the software may be happening faster than expected.

For some other OSs, such as DOS, the BIOS should restore the timer back to 54.6 ms before passing control to the OS. When the BIOS is entering ALT access mode before entering the suspend state it is not necessary to restore the timer contents after the exit from ALT access mode.

5.11.10.1 Write Only Registers with Read Paths in ALT Access Mode

The registers described in Table 76 have read paths in ALT access mode. The access number field in the table indicates which register will be returned per access to that port.

Table 74. Write Only Registers with Read Paths in ALT Access Mode (Sheet 1 of 2)

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
00h	2	1	DMA Chan 0 base address low byte	40h	7	1	Timer Counter 0 status, bits [5:0]
		2	DMA Chan 0 base address high byte			2	Timer Counter 0 base count low byte
01h	2	1	DMA Chan 0 base count low byte			3	Timer Counter 0 base count high byte
		2	DMA Chan 0 base count high byte			4	Timer Counter 1 base count low byte
02h	2	1	DMA Chan 1 base address low byte			5	Timer Counter 1 base count high byte
		2	DMA Chan 1 base address high byte			6	Timer Counter 2 base count low byte
03h	2	1	DMA Chan 1 base count low byte			7	Timer Counter 2 base count high byte
		2	DMA Chan 1 base count high byte	41h	1		Timer Counter 1 status, bits [5:0]
04h	2	1	DMA Chan 2 base address low byte	42h	1		Timer Counter 2 status, bits [5:0]
		2	DMA Chan 2 base address high byte	70h	1		Bit 7 = NMI Enable, Bits [6:0] = RTC Address
05h	2	1	DMA Chan 2 base count low byte	C4h	2	1	DMA Chan 5 base address low byte
		2	DMA Chan 2 base count high byte			2	DMA Chan 5 base address high byte
06h	2	1	DMA Chan 3 base address low byte	C6h	2	1	DMA Chan 5 base count low byte
		2	DMA Chan 3 base address high byte			2	DMA Chan 5 base count high byte
07h	2	1	DMA Chan 3 base count low byte	C8h	2	1	DMA Chan 6 base address low byte
		2	DMA Chan 3 base count high byte			2	DMA Chan 6 base address high byte

NOTES:
1. The OCW1 register must be read before entering ALT access mode.
2. Bits 5, 3, 1, and 0 return zero.

Table 74. Write Only Registers with Read Paths in ALT Access Mode (Sheet 2 of 2)

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
08h	6	1	DMA Chan 0-3 Command[2]	CAh	2	1	DMA Chan 6 base count low byte
		2	DMA Chan 0-3 Request			2	DMA Chan 6 base count high byte
		3	DMA Chan 0 Mode: Bits(1:0) = "00"	CCh	2	1	DMA Chan 7 base address low byte
		4	DMA Chan 1 Mode: Bits(1:0) = "01"			2	DMA Chan 7 base address high byte
		5	DMA Chan 2 Mode: Bits(1:0) = "10"	CEh	2	1	DMA Chan 7 base count low byte
		6	DMA Chan 3 Mode: Bits(1:0) = "11".			2	DMA Chan 7 base count high byte
20h	12	1	PIC ICW2 of Master controller	D0h	6	1	DMA Chan 4-7 Command[2]
		2	PIC ICW3 of Master controller			2	DMA Chan 4-7 Request
		3	PIC ICW4 of Master controller			3	DMA Chan 4 Mode: Bits(1:0) = "00"
		4	PIC OCW1 of Master controller[1]			4	DMA Chan 5 Mode: Bits(1:0) = "01"
		5	PIC OCW2 of Master controller			5	DMA Chan 6 Mode: Bits(1:0) = "10"
		6	PIC OCW3 of Master controller			6	DMA Chan 7 Mode: Bits(1:0) = "11".
		7	PIC ICW2 of Slave controller				
		8	PIC ICW3 of Slave controller				
		9	PIC ICW4 of Slave controller				
		10	PIC OCW1 of Slave controller[1]				
		11	PIC OCW2 of Slave controller				
		12	PIC OCW3 of Slave controller				

NOTES:
1. The OCW1 register must be read before entering ALT access mode.
2. Bits 5, 3, 1, and 0 return zero.

5.11.10.2 Programmable Interrupt Controller (PIC) Reserved Bits

Many bits within the PIC are reserved, and must have certain values written in order for the PIC to operate properly. Therefore, there is no need to return these values in ALT access mode. When reading PIC registers from 20h and A0h, the reserved bits shall return the values listed in the following table.

Table 75. PIC Reserved Bits Return Values

PIC Reserved Bits	Value Returned
ICW2(2:0)	000
ICW4(7:5)	000
ICW4(3:2)	00
ICW4(0)	0
OCW2(4:3)	00
OCW3(7)	0
OCW3(5)	Reflects bit 6
OCW3(4:3)	01

5.11.10.3 Read-Only Registers with Write Paths in ALT Access Mode

The registers described in Table 76 have write paths to them in ALT access mode. Software will restore these values after returning from a powered down state. These registers must be handled special by software. When in normal mode, writing to the base address/count register also writes to the current address/count register. Therefore, the base address/count must be written first, then the part is put into ALT access mode and the current address/count register is written.

Table 76. Register Write Accesses in ALT Access Mode

I/O Address	Register Write Value
08h	DMA Status Register for channels 0-3.
D0h	DMA Status Register for channels 4-7.

5.11.11 System Power Supplies, Planes, and Signals

5.11.11.1 Power Plane Control with SLP_S3#, SLP_S4# and SLP_S5#

The SLP_S3# output signal may be used to cut power to the system core supply, since it will only go active for the STR state (typically mapped to ACPI S3). Power must be maintained to system memory, the Intel® 6300ESB ICH resume well, and to any other circuits that need to generate Wake signals from the STR state.

Cutting power to the core may be done through the power supply, or by external FETs to the motherboard. The SLP_S4# or SLP_S5# output signal may be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done through the power supply, or by external FETs to the motherboard.

5.11.11.2 PWROK Signal

The PWROK input should go active based on the core supply voltages becoming valid. PWROK should go active at least 16 ms after the power is ensured valid.

Note: Please review these notes regarding the PWROK signal:

1. Traditional designs have a reset button logically ANDs with the PWROK signal from the power supply and the processor's voltage regulator module. When this is done with the Intel® 6300ESB ICH, the PWROK_FLR bit will be set. The Intel® 6300ESB ICH treats this internally as though the RSMRST# signal had gone active. However, it is not treated as a full power failure. When PWROK goes inactive and then active (but RSMRST# stays high), then the Intel® 6300ESB ICH will reboot (regardless of the state of the AFTERG3 bit). When the RSMRST# signal also goes low before PWROK goes high, then this is a full power failure and the reboot policy is controlled by the AFTERG3 bit.
2. PWROK and RSMRST# are sampled using the RTC clock. Therefore, low times that are less than one RTC clock period may not be detected by the Intel® 6300ESB ICH.

5.11.11.3 VRMPWRGD Signal

The VRMPWRGD signal is not implemented in the Intel® 6300ESB ICH. VRMPWRGD need to be pulled up to Vcc in order to disable internal legacy logic. If not pulled up, this logic may come up in an unknown state.

5.11.11.4 Controlling Leakage and Power Consumption during Low-Power States

To control leakage in the system, various signals will tri-state or go low during some low-power states.

General principles:

- All signals going to powered down planes (either internally or externally) must be either tri-stated or driven low.
- Signals with pull-up resistors should not be low during low-power states. This is to avoid the power consumed in the pull-up resistor.
- Buses should be halted (and held) in a known state to avoid a floating input (perhaps to some other device). Floating inputs may cause extra power consumption.

Based on the above principles, the following measures are taken:

- During S3 (STR), all signals attached to powered down planes will be tri-stated or driven low.

5.11.12 Clock Generators

The clock generator is expected to provide the frequencies shown in Table 77.

Table 77. Intel® 6300ESB ICH Clock Inputs

Clock Domain	Frequency	Source	Description
CLK66	66 MHz	Main Clock Generator	Clock for Hub Interface. Should be running in all Cx states. Stopped in S3 ~ S5 based on SLP_S3# assertion. This signal is not 5V tolerant.
PCICLK	33 MHz	Main Clock Generator	Free-running PCI Clock to the Intel® 6300ESB ICH. Provides timing for all transactions on the internal primary PCI bus, as well as units inside the Intel® 6300ESB ICH. This clock may be stopped in S3, S3 or S5 states. This signal is not 5V tolerant.
PXPCICLK	66 MHz	Main Clock Generator	Free-running PCI Clock to the Intel® 6300ESB ICH. Provides timing for all transactions on the internal primary PCI-X bus, as well as units inside the Intel® 6300ESB ICH. This clock may be stopped in S3, S3 or S5 states. This signal is not 5V tolerant.
CLK48	48 MHz	Main Clock Generator	Used by USB Controllers. Stopped in S3 ~ S5 based on SLP_S3# assertion.
CLK14	14.318 MHz	Main Clock Generator	Used by ACPI timers. Stopped in S3 ~ S5 based on SLP_S3# assertion.
AC_BIT_CLK	12.288 MHz	AC'97 Codec	**AC'97 Bit Clock:** 12.288 MHz serial data clock generated by the external CODECs. Integrated pull down resistor.

5.11.13 Legacy Power Management Theory of Operation

5.11.13.1 Overview

Instead of relying on ACPI software, legacy power management uses BIOS and various hardware mechanisms. The Intel® 6300ESB ICH has a greatly simplified method for legacy power management compared with previous component generations.

The scheme relies on the concept of detecting when individual subsystems are idle, detecting when the whole system is idle, and detecting when accesses are attempted to idle subsystems.

However, the OS is assumed to be at least APM enabled. Without APM calls, there is no quick way to know when the system is idle between keystrokes. The Intel® 6300ESB ICH does not support the burst modes found in previous components.

5.11.13.2 APM Feature Notes

The Intel® 6300ESB ICH has a timer that, when enabled by the 1MIN_EN bit in the SMI Control and Enable register, will generate an SMI# once per minute. The SMI handler may check for system activity by reading the DEVACT_STS register. When none of the system bits are set, the SMI handler may increment a software counter. When the counter reaches a sufficient number of consecutive minutes with no activity, the SMI handler may then put the system into a lower power state. When there is activity, various bits in the DEVACT_STS register will be set. Software clears the bits by writing a one to the bit position.

The DEVACT_STS register allows for monitoring various internal devices, or Super I/O devices (SP, PP, FDC) on LPC or PCI, keyboard controller accesses, or audio functions on LPC or PCI. Other PCI activity may be monitored by checking the PCI interrupts.

5.12 System Management (D31:F0)

5.12.1 Overview of System Management Functions

The Intel® 6300ESB ICH provides various functions to make a system easier to manage and to lower the Total Cost of Ownership (TCO) of the system. It builds on functions that have been found in prior generations of serial interface ACPI-compatible processor system monitor components products, such as the LM78 and LM80. Features and functions may be augmented through external A/D converters and GPIO, as well as an external microcontroller.

The Intel® 6300ESB ICH supports the following features and functions:

- First Hard Coded Timer to Generate SMI# after Programmable Time.
- First timeout causes SMI#. Allows for SMM-Based Recovery from OS lockup.
- OS-based software agent accesses the Intel® 6300ESB ICH to periodically reload timer.
- Ability for SMM Handler to generate “TCO” interrupt to OS.
- Allows for OS-based code augmentation.
- Ability for OS to generate SMI#.
- Call-back from OS to TCO code in SMM handler.
- Second hard coded timer to generate reboot after programmable time.

- Used only after first timeout occurs.
- Second timeout allows for system “reset and reboot” when a hardware error is detected. Various system states are preserved through this special reset to allow for possible error detection and correction.
- Reset associated with “reboot” may attempt to preserve some registers for diagnostic purposes.
- SMI# handler must periodically reload second timer to prevent “reboot” (timeout during SMI is assumed as broken processor or stuck hardware).
- Option to generate limited reset when second timeout occurs.
- Ability to detect a “Broken” processor.
- Detects when the processor fails to restart after it has been reset.
- When processor failure detected, option to pulse a GPIO or send SMBus message. The SMBus message may be used to indicate to an external LAN controller to send a distress message. The GPIO may control an LED with optional blink.
- Ability to Handle Various Errors (such as ECC Errors) Indicated by MCH.
- Can generate SMI# or TCO interrupt.
- Intruder Detect input when the system cover is removed.
- May generate TCO interrupt or SMI#.
- Ability for TCO messages to coexist with standard SMBus devices.
- Detection of bad FWH programming. Done by checking that data on the first read is not FFh.

5.12.2 TCO Signal Usage

5.12.2.1 Intruder# Signal

This signal may be used to detect the chassis being opened. The activation of this signal may be used to cause an SMI#, and is reported through the Heartbeat/Event mechanism. When SMI# is desired, the signal’s level may be read, so this may be used as a type of General Purpose Input.

5.12.2.2 Pin Straps

Some the TCO functions are decided at powerup (rising edge of PWROK). See Section 3.21, “Pin Straps” for specific assignments of the pin straps.

5.12.2.3 SMLINK Signals

The Intel® 6300ESB ICH supports TCO compatible mode connectivity. The Intel® 6300ESB ICH supports LAN controllers. A LAN controller can be used to receive or retrieve TCO message or information on Host SMBus if needed. In Legacy TCO mode messages will be driven via SMLink. For the Intel® 6300ESB ICH, messages on this link will use SMBus protocol at the rates described in Section 5.12, “System Management (D31:F0)” for TCO compatible mode.

Note: All mention of “LAN” refers to an external LAN controller.

5.12.3 TCO Theory of Operation

5.12.3.1 Overview

The System Management functions are designed to allow the system to diagnose failing subsystems. The intent of this logic is that some of the system management functionality be provided without the aid of an external microcontroller.

5.12.3.2 Detecting a System Lockup

When the processor is reset, it is expected to fetch its first instruction. When the processor fails to fetch the first instruction after reset, the TCO timer will timeout twice and the Intel® 6300ESB ICH will assert PXPCIRST#.

When TCO Reboots are not enabled, then the Intel® 6300ESB ICH will either:

- The SMLink will still send out the first 8 bits of the message. After the eighth bit, the logic will stall because there is no integrated LAN controller to send the ACK. The logic will abort the transfer. External logic may monitor the toggling and use that to drive LED.
- If an LAN controller is connected: send the appropriate message to the LAN controller.

When TCO Reboots are enabled, then the Intel® 6300ESB ICH will attempt to reboot the system.

Note: When the NO-REBOOT bit (D31:F0:Offset D4:bit 1) is set (no reboots are intended), and the SECOND_TO_STS bit (TCO I/O Offset 06h, bit 1) is set, and the DOACPU_STS bit (TCO I/O Offset 06h, bit 2), the Intel® 6300ESB ICH will indicate this in the TCO message by setting the CPU Missing bit in the message.

When the NO-REBOOT bit is not set (reboots intended), and the SECOND_TO_STS bit is set, the Intel® 6300ESB ICH will attempt to reboot. After the reboot, the SECOND_TO_STS bit will still be set. When the processor fails to fetch the first instruction, the DOA_CPU_STS bit is set, and when the TCO timer times out (actually for the third time, the first 2 times caused the SECOND_TO_STS bit to be set), the Intel® 6300ESB ICH will set the CPU MISSING EVENT bit for the TCO message.

5.12.3.3 Handling an OS Lockup

Under some conditions, the OS may lock up. To handle this, the TCO Timer is used with the following algorithm:

1. BIOS programs the TCO Timer, through the TCO_TMR register, with an initial value. Generally, this will probably be set to four seconds, but could be greater.
2. An OS-based software agent periodically writes to the TCO_RLD register to reload the timer and keep it from generating the SMI#. The software agent may read the TCO_RLD register to see when it is close to timing out, and possibly determine if the time-out should be increased.
3. When the timer reaches 0, an SMI# may be generated. This should only occur when the OS was not able to reload the timer. It is assumed that the OS will not be able to reload the timer if it has locked up.
4. Upon generating the SMI#, the TCO Timer automatically reloads with the value in the TCO_TMR register and start counting down.
5. The SMI handler may then:
   a. Read the TIMEOUT bit in the TCO_STS register to check that the SMI# was caused by the TCO timer.

b. Write to the TCO_RLD register to reload the timer to make sure the TCO timer does not reach 0 again.

c. Attempt to recover. May need to periodically reload the TCO timer.

The exact recovery algorithm will be system-specific.

When after the TIMEOUT SMI is generated, and the TCO timer again reaches 0, and reboots are enabled, the System Management logic will reset (and reboot) the system. This would be in the case where the processor or system is locked up. During every boot, BIOS should read the SECOND_TO_STS bit in the TCO_STS register to see if this is normal boot or a reboot due to the timeout.

5.12.3.4 Handling an Intruder

The Intel® 6300ESB ICH has an input signal, INTRUDER#, that may be attached to a switch that is activated by the system's case being open. This input has a two RTC clock debounce. When INTRUDER# goes active (after the debouncer), this will set the INTRD_DET bit in the TCO_STS register. The INTRD_SEL bits in the TCO_CNT register may enable the Intel® 6300ESB ICH to cause an SMI# or interrupt. The BIOS or interrupt handler may then cause a transition to the S5 state by writing to the SLP_EN bit.

The software may also directly read the status of the INTRUDER# signal (high or low) by clearing and then reading the INTRD_DET bit. This allows the signal to be used as a GPI when the intruder function is not required.

When the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, the INTRD_DET signal will go to a 0 when INTRUDER# input signal goes inactive. Note that this is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

Note: The INTRD_DET bit resides in the Intel® 6300ESB ICH's RTC well, and is set and cleared synchronously with the RTC clock. Thus, when software attempts to clear INTRD_DET (by writing a '1' to the bit location) there may be as much as two RTC clocks (about 65 µs) delay before the bit is actually cleared. Also, the INTRUDER# signal should be asserted for a minimum of 1 ms in order to ensure that the INTRD_DET bit will be set.

Note: When the INTRUDER# signal is still active when software attempts to clear the INTRD_DET bit, the bit will remain set and the SMI will be generated again immediately. The SMI handler may clear the INTRD_SEL bits to avoid further SMIs. However, when the INTRUDER# signal goes inactive and then active again, there will not be further SMIs, since the INTRD_SEL bits would select that no SMI# be generated.

5.12.3.5 Detecting Improper FWH Programming

The Intel® 6300ESB ICH may detect the case where the FWH is not programmed. This will result in the first instruction fetched to have a value of FFh. When this occurs, the Intel® 6300ESB ICH will set the BAD_BIOS bit, which may then be reported through the Heartbeat and Event reporting via an LAN Controller.

5.12.3.6 Handling an ECC Error or Other Memory Error

The Host Controller provides a message to indicate that it would like to cause an SMI#, SCI, SERR#, or NMI. The software must check the Host Controller as to the exact cause of the error.

5.12.4 Heartbeat and Event Reporting through SMLink/ SMbus

SMLink signals are implemented on the Intel® 6300ESB ICH ICH to support TCO compatible mode. Heartbeat and event reporting are accomplished via the SMLink signals.

5.12.4.1 Overview

5.12.4.1.1 TCO Compatible Mode

The Intel® 6300ESB ICH may function directly with a LAN Controller to report messages to a network management console without the aid of the system processor. This is crucial in cases where the processor is malfunctioning or cannot function due to being in a low-power state.

Note: A system that has locked up and cannot be restarted with a power button press is assumed to have broken hardware (bad power supply, short circuit on some bus, etc.), and is beyond the Intel® 6300ESB ICH's recovery mechanisms.

The basic scheme is for the Intel® 6300ESB ICH to send specific messages through the SMLink
I/F to the LAN. Upon receiving the SMLink message, the LAN has a prepared Ethernet message that it may send to a network management console. The prepared message is stored in a non-volatile memory connected directly to the LAN.

Messages will be sent by the Intel® 6300ESB ICH to a LAN either because a specific event has occurred (see Table 78), or they will be sent periodically (also known as a Heartbeat). The Event and Heartbeat messages will have exactly the same form.

Table 78. Event Transitions that Cause Messages

Event	Assertion ?	Deassertion ?	Comments
INTRUDER# pin	Yes	No	Must be in "heartbeat mode" (G1 or hung G0).
THRM# pin	Yes	Yes	Must be in "heartbeat mode". Note that the THRM# pin is isolated when the core power is off, thus preventing this event in S3-S5. Also, the THRM# pin is sampled with the PCI clock, which means that THRM# transitions cannot be detected in S1-M.
Watchdog Timer Expired	Yes	No (NA)	"Heartbeat mode" entered
SEND_NOW bit	Yes	NA	Occurs in G0
GPIO[11]/SMBALERT# pin	Yes	Yes	Must be in "heartbeat mode" (G1 or hung G0).
CPU_PWR_FLR	Yes	No	"HeartBeat mode" entered {Intel® 6300ESB ICH DCN36}.

NOTE: The GPIO[11]/SMBALERT# pin will trigger an event message (when enabled by the GPIO11_ALERT_DISABLE bit) regardless of whether it is configured as a GPI or not.

Whenever an event occurs that causes the Intel® 6300ESB ICH to send a new message, it will increment its SEQ[3:0] field. For Heartbeat messages, the sequence number will not increment.

When a triggering event occurs while a message is already being generated and sent, the new event may not appear in the current message. If not, then a second message will be generated, with the SEQ[3:0] field incremented, to report the new event.

The following rules/steps apply when the system is in a G0 state and the policy is for the Intel® 6300ESB ICH to reboot the system after a hardware lockup:

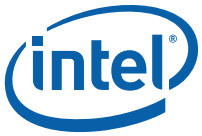
1. Upon detecting the lockup the SECOND_TO_STS bit will be set. The Intel® 6300ESB ICH may send up to 1 Event message to the LAN. The Intel® 6300ESB ICH will then attempt to reboot the processor.
2. When the reboot at step 1 is successful then the BIOS should clear the SECOND_TO_STS bit. This will prevent any further Heartbeats from being sent. The BIOS may then perform addition recovery/boot steps.

Warning: It is important that the BIOS clears the SECOND_TO_STS bit, as the messages (alerts) will interfere with the LAN device driver from working properly. The alerts reset part of the LAN and would prevent an OS's device driver from sending or receiving some messages.

3. When the reboot attempt in step 1 is not successful, then the timer will timeout a third time. At this point the system has locked up and was unsuccessful in rebooting. The Intel® 6300ESB ICH will not attempt to automatically reboot again. The Intel® 6300ESB ICH will start sending a message every heartbeat period (30-32 seconds). The heartbeats will continue until some external intervention occurs (reset, power failure, etc.).
4. After step 3 (unsuccessful reboot after third timeout), when the user does a Power Button Override, the system will go to an S5 state. The Intel® 6300ESB ICH will continue sending the messages every heartbeat period.
5. After step 4 (power button override after unsuccessful reboot) when the user presses the Power Button again, the system should wake to an S0 state and the processor should start executing the BIOS.
6. When step 5 (power button press) is successful in waking the system, the Intel® 6300ESB ICH will continue sending messages every heartbeat period until the BIOS clears the SECOND_TO_STS bit.
7. When step 5 (power button press) is unsuccessful in waking the system, the Intel® 6300ESB ICH will continue sending a message every heartbeat period. The Intel® 6300ESB ICH will not attempt to automatically reboot again. The Intel® 6300ESB ICH will start sending a message every heartbeat period (30-32 seconds). The heartbeats will continue until some external intervention occurs (reset, power failure, etc.).
8. After step 3 (unsuccessful reboot after third timeout), when a reset is attempted (using a button that pulses PWROK low or through the message on the SMBus slave I/F), the Intel® 6300ESB ICH will attempt to reset the system.
9. After step 8 (reset attempt), when the reset is successful, then the BIOS will be run. The Intel® 6300ESB ICH will continue sending a message every heartbeat period until the BIOS clears the SECOND_TO_STS bit.
10. After step 8 (reset attempt), when the reset is unsuccessful, then the Intel® 6300ESB ICH will continue sending a message every heartbeat period. The Intel® 6300ESB ICH will not attempt to reboot the system again without external intervention.

The following rules/steps apply when the system is in a G0 state and the policy is for the Intel® 6300ESB ICH to **not** reboot the system after a hardware lockup:

1. Upon detecting the lockup the SECOND_TO_STS bit will be set. The Intel® 6300ESB ICH will send a message with the Watchdog (WD) Event status bit set (and any other bits that must also be set). This message will be sent as soon as the lockup is detected, and will be sent with the next (incremented) sequence number.
2. After step 1, the Intel® 6300ESB ICH will send a message every heartbeat period until some external intervention occurs.
3. Rules/steps 4-10 apply when no user intervention (resets, power button presses, SMBus reset messages) occur after a third timeout of the watchdog timer. When the intervention occurs before the third timeout, then jump to rule/step11.
4. After step 3 (third timeout), when the user does a Power Button Override, the system will go to an S5 state. The Intel® 6300ESB ICH will continue sending heartbeats at this point.
5. After step 4 (power button override), when the user presses the power button again, the system should wake to an S0 state and the processor should start executing the BIOS.
6. When step 5 (power button press) is successful in waking the system, the Intel® 6300ESB ICH will continue sending heartbeats until the BIOS clears the SECOND_TO_STS bit.
7. When step 5 (power button press) is unsuccessful in waking the system, the Intel® 6300ESB ICH will continue sending heartbeats. The Intel® 6300ESB ICH will not attempt to reboot the system again until some external intervention occurs (reset, power failure, etc.).
8. After step 3 (third timeout), when a reset is attempted (using a button that pulses PWROK low or through the message on the SMBus slave I/F), the Intel® 6300ESB ICH will attempt to reset the system.
9. When step 8 (reset attempt) is successful, then the BIOS will be run. The Intel® 6300ESB ICH will continue sending heartbeats until the BIOS clears the SECOND_TO_STS bit.
10. When step 8 (reset attempt), is unsuccessful, then the Intel® 6300ESB ICH will continue sending heartbeats. The Intel® 6300ESB ICH will not attempt to reboot the system again without external intervention.
11. This and the following rules/steps apply when the user intervention (power button press, reset, SMBus message, etc.) occur prior to the third timeout of the watchdog timer.
12. After step 1 (second timeout), when the user does a Power Button Override, the system will go to an S5 state. The Intel® 6300ESB ICH will continue sending heartbeats at this point.
13. After step 12 (power button override), when the user presses the power button again, the system should wake to an S0 state and the processor should start executing the BIOS.
14. When step 13 (power button press) is successful in waking the system, the Intel® 6300ESB ICH will continue sending heartbeats until the BIOS clears the SECOND_TO_STS bit.
15. When step 13 (power button press) is unsuccessful in waking the system, the Intel® 6300ESB ICH will continue sending heartbeats. The Intel® 6300ESB ICH will not attempt to reboot the system again until some external intervention occurs (reset, power failure, etc.).
16. After step 1 (second timeout), when a reset is attempted (using a button that pulses PWROK low or through the message on the SMBus slave I/F), the Intel® 6300ESB ICH will attempt to reset the system.
17. When step 16 (reset attempt) is successful, then the BIOS will be run. The Intel® 6300ESB ICH will continue sending heartbeats until the BIOS clears the SECOND_TO_STS bit.
18. When step 16 (reset attempt), is unsuccessful, then the Intel® 6300ESB ICH will continue sending heartbeats. The Intel® 6300ESB ICH will not attempt to reboot the system again without external intervention.



The following rules will apply when the system is in a G1 (S1-S4) state:

1. The Intel® 6300ESB ICH will send a Heartbeat message every Heartbeat Period (30-32 seconds).
2. When an event occurs prior to the system being shut down, the Intel® 6300ESB ICH will immediately send another Event message with the next (incremented) sequence number.
3. After the event, it will resume sending Heartbeat messages.

Note: There is a boundary condition when a hardware event (event or heartbeat) happens right as the system is transitioning into a G0 state. In this condition, the hardware will send messages even though the system will be in a G0 state (and the status bits could potentially indicate that). Normally the Intel® 6300ESB ICH will not send heartbeats in the G0 state (except in the case of a lockup).

Note: A spurious alert could occur in the following sequence:

a. The processor has initiated an alert using the SEND_NOW bit

b. During the alert, the THRM#, INTRUDER# or GPI[11] changes state

c. The system then goes to a non-S0 state.

Once the system transitions to the non-S0 state, it may send a single alert with an incremented SEQUENCE number.

Note: An inaccurate alert message may be generated in the following scenario:

a. The system successfully boots after a second watchdog Timeout occurs.

b. PWROK goes low (typically due to a reset button press) or a power button override occurs (before the SECOND_TO_STS bit is cleared).

c. An alert message indicating that the processor is missing or locked up is generated with a new sequence number.

5.13 General Purpose I/O

5.13.1 GPIO Mapping

Table 79. GPIO Implementation (Sheet 1 of 2)

GPIO	Type	Alternate Function	Power Well	Tolerant	Notes
GPI[0]	Input Only	PXREQ[2]#	Core	5.0 V	GPIO_USE_SEL bit 0 enables REQ/GNT[A]# pair. Input active status read from GPE0_STS register bit 16. Input active high/low set through GPI_INV register bit 0. **NOTE:** GPEO_STS register, Section 8.8.3.7 GPI_INV register, Section 8.10.6
GPI[1]	Input Only	PXREQ[3]#	Core	5.0 V	GPIO_USE_SEL bit 1 enables REQ/GNT[B]# pair. Input active status read from GPE0_STS register bit 17 Input active high/low set through GPI_INV register bit 1.
GPI[2:5]	Input Only	PIRQ[E:H]#	Core	5.0 V	GPIO_USE_SEL bits [2:5] enable PIRQ[E:H]#. Input active status read from GPE0_STS reg. bits [18:21]. Input active high/low set through GPI_INV reg. bit [2:5].
GPI[6]	Input Only	Unmuxed	Core	3.3 V	Input active status read from GPE0_STS register bit 22. Input active high/low set through GPI_INV register bit 6.
GPI[7]	Input Only	Unmuxed	Core	3.3 V	Input active status read from GPE0_STS register bit 23. Input active high/low set through GPI_INV register bit 7
GPI[8]	Input Only	Unmuxed	Resume	3.3 V	Input active status read from GPE0_STS register bit 24. Input active high/low set through GPI_INV register bit 8.
GPI[11]	Input Only	SMBALERT#	Resume	3.3 V	GPIO_USE_SEL bit 11 enables SMBALERT# Input active status read from GPE0_STS register bit 27. Input active high/low set through GPI_INV register bit 11.
GPI[12]	Input Only	Unmuxed	Resume	3.3 V	Input active status read from GPE0_STS register bit 28. Input active high/low set through GPI_INV register bit 12.
GPI[13]	Input Only	Unmuxed	Resume	3.3 V	Input active status read from GPE0_STS register bit 29. Input active high/low set through GPI_INV register bit 13.

NOTES:
1. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:43] are in the core well.
2. GPIO[8:13] and GPIO[24:28] are in the suspend well.
3. Core-well GPIO are 5V tolerant, except for GPIO[7:6] and [32:43].
4. Resume-well GPIO are not 5V tolerant.
5. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.

Table 79. GPIO Implementation (Sheet 2 of 2)

GPIO	Type	Alternate Function	Power Well	Tolerant	Notes
GPO[16]	Output Only	PXGNT[2]#	Core	5.0 V	Output controlled through GP_LVL register bit 16. TTL driver output **NOTE:** GP_LVL register, Section 8.10.4
GPO[17]	Output Only	PXGNT[3]#	Core	5.0 V	Output controlled through GP_LVL register bit 17. TTL driver output
GPO[18]	Output Only	Unmuxed	Core	5.0 V	Output controlled through GP_LVL register bits [18:19]. TTL driver output
GPO[19]	Output Only	Unmuxed	Core	5.0 V	Output controlled through GP_LVL register bits [18:19]. TTL driver output
GPO[20	Output Only	Unmuxed	Core	5.0 V	Output controlled through GP_LVL register bit 20. TTL driver output
GPO[21]	Output Only	Unmuxed	Core	5.0 V	Output controlled through GP_LVL register bit 21. TTL driver output
GPIO[23	Output Only	Unmuxed	Core	5.0 V	Output controlled through GP_LVL register bit [23]. TTL driver output
GPIO[24]	I/O	Unmuxed	Resume	3.3 V	Input active status read from GP_LVL register bit 24. Output controlled through GP_LVL register bit 24. TTL driver output
GPIO[25]	I/O	Unmuxed	Resume	3.3 V	Blink enabled through GPO_BLINK register bit 25. Input active status read from GP_LVL register bit 25 Output controlled through GP_LVL register bit 25. TTL driver output
GPIO[27:28]	I/O	Unmuxed	Resume	3.3 V	Input active status read from GP_LVL register bits [27:28] Output controlled through GP_LVL register bits [27:28] TTL driver output
GPIO[32]	I/O	WDT_TOUT#	Core	3.3 V	
GPIO[36:33]	I	PXIRQ[3:0]#	Core	3.3 V	
GPIO[39:37]	I/O	Unmuxed	Core	3.3 V	
GPIO[43:40]	I/O	Unmuxed	Core	3.3 V	These GPIOs have high strength output capability (for driving LEDs)
GPIO[57:56]	OD	Unmuxed	Resume and RTC	3.3 V	

NOTES:
1. GPIO[0:7], GPIO[16:21, 23], and GPIO[32:43] are in the core well.
2. GPIO[8:13] and GPIO[24:28] are in the suspend well.
3. Core-well GPIO are 5V tolerant, except for GPIO[7:6] and [32:43].
4. Resume-well GPIO are not 5V tolerant.
5. GPIO[56:57] pads are in the suspend well, the register bits are in the RTC well.

5.13.2 Power Wells

Some GPIOs exist in the resume power plane. Care must be taken to make sure GPIO signals are not driven high into powered-down planes.

Some Intel® 6300ESB ICH GPIOs may be connected to pins on devices that exist in the core well. When these GPIOs are outputs, there is a danger that a loss of core power (PWROK low) or a Power Button Override event will result in the Intel® 6300ESB ICH driving a pin to a logic ‘1’ to another device that is powered down.

5.13.3 SMI# and SCI Routing

The routing bits for GPIO[0:15] allow an input to be routed to SMI# or SCI, or neither. See Section 8.8.3.3 for the routing register

Note: A bit may be routed to either an SMI# or an SCI, but not both.

5.13.4 Triggering

GPIO[0:15] have “sticky” bits on the input. See Section 8.8.3.7 for the GPE0_STS register. As long as the signal goes active for at least 2 clocks, the Intel® 6300ESB ICH will keep the sticky status bit active. The active level (high or low) can be selected via the GP_INV register.

If the system is in an S0 or S1-D state, the GPI are sampled at 33 MHz, so the signal only needs to be active for about 60 ns to be latched. In the S3-S5 states, the GPI are sampled at 32.768 KHz, and thus must be active for at least 61 microseconds to be latched.

Note: GPIs that are in the core well are not capable of waking the system from sleep states where the core well is not powered.

If the input signal is still active when the latch is cleared, it will again be set (another edge is not required). This makes these signals ìlevelî triggered inputs.

5.14 IDE Controller (D31:F1)

5.14.1 Overview

The Intel® 6300ESB ICH IDE controller features two sets of interface signals (Primary and Secondary) that may be independently enabled, tri-stated or driven low.

The Intel® 6300ESB ICH IDE controller supports both legacy mode and native mode IDE interface. In native mode, the IDE controller is a fully PCI compliant software interface and does not use any legacy I/O or interrupt resources.

The IDE interfaces of the Intel® 6300ESB ICH may support several types of data transfers:

Programmed I/O (PIO): Processor is in control of the data transfer.

8237 style DMA: DMA protocol that resembles the DMA on the ISA bus, although it does not use the 8237 in the Intel® 6300ESB ICH. This protocol off loads the processor from moving data. This allows higher transfer rate of up to 16 Mbytes/s.

Ultra ATA/33: DMA protocol that redefines signals on the IDE cable to allow both host and target throttling of data and transfer rates of up to 33 Mbytes/s.

Ultra ATA/66: DMA protocol that redefines signals on the IDE cable to allow both host and target throttling of data and transfer rates of up to 66 Mbytes/s.

Ultra ATA/100: DMA protocol that redefines signals on the IDE cable to allow both host and target throttling of data and transfer rates of up to 100 Mbytes/s.

5.14.2 PIO Transfers

5.14.2.1 Overview

The Intel® 6300ESB ICH IDE controller includes both compatible and fast timing modes. The fast timing modes may be enabled only for the IDE data ports. All other transactions to the IDE registers are run in single transaction mode with compatible timings.

Up to two IDE devices may be attached per IDE connector (drive 0 and drive 1). The IDETIM and SIDETIM Registers permit different timing modes to be programmed for drive 0 and drive 1 of the same connector.

The Ultra ATA/33/66/100 synchronous DMA timing modes may also be applied to each drive by programming the IDE I/O Configuration register and the Synchronous DMA Control and Timing registers. When a drive is enabled for synchronous DMA mode operation, the DMA transfers are executed with the synchronous DMA timings. The PIO transfers are executed using compatible timings or fast timings when also enabled.

5.14.2.2 IDE Port Decode

The Command and Control Block registers are accessed differently depending on the decode mode, which is selected by the Programming Interface configuration register (Offset 09h).

Note: The primary and secondary channels are controlled by separate bits, allowing one to be in native mode and the other in legacy mode simultaneously.

5.14.2.3 IDE Legacy Mode and Native Mode

The Intel® 6300ESB ICH IDE controller supports both legacy mode and PCI native mode. In legacy mode, the Command and Control Block registers are accessible at fixed I/O addresses, may not be accessed through the I/O BARs. These blocks are decoded when I/O space is enabled through the P-ATA function's configuration space and ATA decode is enabled through the PTIM/STIM registers, bit 15. An access to these addresses results in the assertion of the appropriate chip select (CS1#/CS3#) and the command strobes (DIOR#, DIOW#).

There are two I/O ranges for each IDE cable: the Command Block, which corresponds to the CS1P#/CS1S# chip select, and the Control Block, which corresponds to the CS3P#/CS3S# chip select. The Command Block is an 8 byte range, while the control block is a 4 byte range.

- **Command Block Offset:** 01F0h for Primary, 0170h for Secondary
- **Control Block Offset:** 03F4h for Primary, 0374h for Secondary

Table 80 and Table 81 specify the registers and transaction timings as they affect the Intel® 6300ESB ICH hardware definition.

Note: The Data Register (I/O Offset 00h) should be accessed using 16-bit or 32-bit I/O instructions. All other registers should be accessed using 8-bit I/O instructions.

Note: These registers are implemented in the IDE device. Therefore, accesses to these I/O registers cause corresponding accesses on the IDE interface.

Table 80. IDE Legacy I/O Ports: Command Block Registers (CS1x# Chip Select)

I/O Offset	Register Function (Read)	Register Function (Write)
00h	Data	Data
01h	Error	Features
02h	Sector Count	Sector Count
03h	Sector Number	Sector Number
04h	Cylinder Low	Cylinder Low
05h	Cylinder High	Cylinder High
06h	Drive	Head
07h	Status	Command

NOTE: For accesses to the Alt Status register in the Control Block, the Intel® 6300ESB ICH must always force the upper address bit (PDA[2] or SDA[2]) to 1 in order to ensure proper native mode decode by the IDE device. Unlike the legacy mode fixed address location, the native mode address for this register may contain a 0 in address bit 2 when it is received by the Intel® 6300ESB ICH.

In native mode, the Intel® 6300ESB ICH will not decode the legacy ranges. The same offsets are used as in Table 80, however, the base addresses are selected using the PCI BARs, rather than fixed I/O locations.

For accesses to the Alt Status register in the Control Block, the P-ATA host controller must always force the upper address bit (PDA[2] or SDA[2]) to 1 in order to ensure proper decode by the P-ATA device. Unlike the Legacy Mode fixed address location, the Native Mode address for this register may contain a 0 in address bit 2 when it is received by the P-ATA host controller.

5.14.2.4 PIO IDE Timing Modes

IDE data port transaction latency consists of startup latency, cycle latency, and shutdown latency.

Startup latency is incurred when a PCI master cycle targeting the IDE data port is decoded and the DA[2:0] and CSxx# lines are not set up. Startup latency provides the setup time for the DA[2:0] and CSxx# lines prior to assertion of the read and write strobes (DIOR# and DIOW#).

Cycle latency consists of the I/O command strobe assertion length and recovery time. Recovery time is provided so that transactions may occur back-to-back on the IDE interface (without incurring startup and shutdown latency) without violating minimum cycle periods for the IDE interface. The command strobe assertion width for the enhanced timing mode is selected by the IDE_TIM Register and may be set to 2, 3, 4, or 5 PCI clocks. The recovery time is selected by the IDE_TIM Register and may be set to 1, 2, 3, or 4 PCI clocks.

When IORDY is asserted when the initial sample point is reached, no wait-states are added to the command strobe assertion length. When IORDY is negated when the initial sample point is reached, additional wait-states are added. Since the rising edge of IORDY must be synchronized, at least two additional PCI clocks are added.

Shutdown latency is incurred after outstanding scheduled IDE data port transactions (either a non-empty write post buffer or an outstanding read prefetch cycles) have completed and before other transactions may proceed. It provides hold time on the DA[2:0] and CSxx# lines with respect to the read and write strobes (DIOR# and DIOW#). Shutdown latency is two PCI clocks in duration.

The IDE timings for various transaction types are shown in Table 81.

Note that bit 2 (16-bit I/O recovery enable) of the ISA I/O Recovery Timer Register does not add wait-states to IDE data port read accesses when any of the fast timing modes are enabled.

Table 81. IDE Transaction Timings (PCI Clocks)

IDE Transaction Type	Startup Latency	IORDY Sample Point (ISP)	Recovery Time (RCT)	Shutdown Latency
Non-Data Port Compatible	4	11	22	2
Data Port Compatible	3	6	14	2
Fast Timing Mode	2	2	1	2

5.14.2.5 IORDY Masking

The IORDY signal may be ignored and assumed asserted at the first IORDY Sample Point (ISP) on a drive by drive basis through the IDETIM Register.

5.14.2.6 PIO 32-Bit IDE Data Port Accesses

A 32-bit PCI transaction run to the IDE data address (01F0h primary, 0170h secondary) results in two back to back 16-bit transactions to the IDE data port. The 32-bit data port feature is enabled for all timings, not just enhanced timing. For compatible timings, a shutdown and startup latency is incurred between the two 16-bit halves of the IDE transaction. This ensures that the chip selects will be deasserted for at least two PCI clocks between the two cycles.

5.14.2.7 PIO IDE Data Port Prefetching and Posting

The Intel® 6300ESB ICH may be programmed through the IDETIM registers to allow data to be posted to and prefetched from the IDE data ports.

Data pre fetching is initiated when a data port read occurs. The read prefetch eliminates latency to the IDE data ports and allows them to be performed back to back for the highest possible PIO data transfer rates. The first data port read of a sector is called the demand read. Subsequent data port reads from the sector are called prefetch reads. The demand read and all prefetch reads much be of the same size (16 or 32 bits).

Data posting is performed for writes to the IDE data ports. The transaction is completed on the PCI bus after the data is received by the Intel® 6300ESB ICH. The Intel® 6300ESB ICH will then run the IDE cycle to transfer the data to the drive. When the Intel® 6300ESB ICH write buffer is non-empty and an unrelated (non-data or opposite channel) IDE transaction occurs, that transaction will be stalled until all current data in the write buffer is transferred to the drive.

5.14.3 Bus Master Function

The Intel® 6300ESB ICH may act as a PCI Bus master on behalf of an IDE slave device. Two PCI Bus master channels are provided, one channel for each IDE connector (primary and secondary). By performing the IDE data transfer as a PCI Bus master, the Intel® 6300ESB ICH off-loads the processor and improves system performance in multitasking environments. Both devices attached to a connector may be programmed for bus master transfers, but only one device per connector may be active at a time.

5.14.3.1 Physical Region Descriptor Format

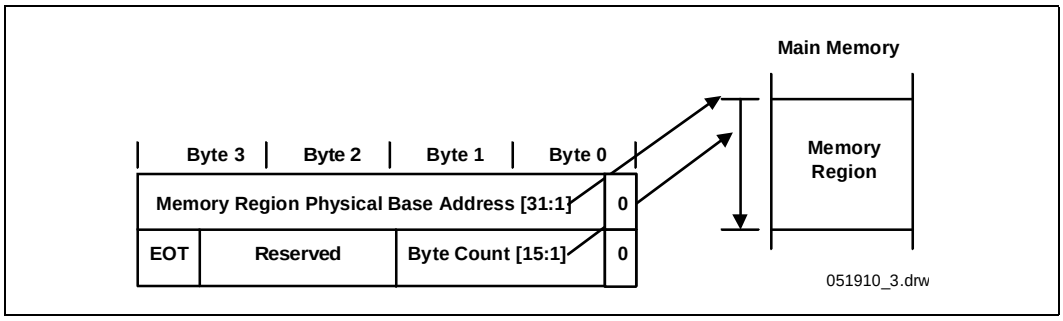
The physical memory region to be transferred is described by a Physical Region Descriptor (PRD). The PRDs are stored sequentially in a Descriptor Table in memory. The data transfer proceeds until all regions described by the PRDs in the table have been transferred. Note that the Intel® 6300ESB ICH bus master IDE function does not support memory regions or descriptor tables located on ISA.

Descriptor Tables must not cross a 64-Kbyte boundary. Each PRD entry in the table is 8 bytes in length. The first 4 bytes specify the byte address of a physical memory region. This memory region must be DWORD aligned and must not cross a 64-Kbyte boundary. The next two bytes specify the size or transfer count of the region in bytes (64-Kbyte limit per region). A value of zero in these two bytes indicates 64 Kbytes (thus the minimum transfer count is 1). When bit 7 (EOT) of the last byte is a 1, it indicates that this is the final PRD in the Descriptor table. Bus master operation terminates when the last descriptor has been retired.

When the Bus Master IDE controller is reading data from the memory regions, bit 1 of the Base Address is masked and byte enables are asserted for all read transfers. When writing data, bit 1 of the Base Address is not masked and if set, will cause the lower WORD byte enables to be deasserted for the first DWORD transfer. The write to PCI will typically consist of a 32-byte cache line. When valid data ends prior to end of the cache line, the byte enables will be deasserted for invalid data.

The total sum of the byte counts in every PRD of the descriptor table must be equal to or greater than the size of the disk transfer request. When greater than the disk transfer request, the driver must terminate the bus master transaction (by setting bit 0 in the Bus Master IDE Command Register to zero) when the drive issues an interrupt to signal transfer completion.

Figure 16. Physical Region Descriptor Table Entry



5.14.3.2 Line Buffer

A single line buffer exists for the Intel® 6300ESB ICH Bus master IDE interface. This buffer is not shared with any other function. The buffer is maintained in either the read state or the write state. Memory writes are typically 4-DWORD bursts and invalid DWORDs have C/BE[3:0]#=0Fh. The line buffer allows burst data transfers to proceed at peak transfer rates.

The Bus Master IDE Active bit in Bus Master IDE Status register is reset automatically when the controller has transferred all data associated with a Descriptor Table (as determined by EOT bit in last PRD). The IDE Interrupt Status bit is set when the IDE device generates an interrupt. These events may occur prior to line buffer emptying for memory writes. When either of these conditions exist, all PCI Master non-Memory read accesses to the Intel® 6300ESB ICH are retried until all data in the line buffers has been transferred to memory.

5.14.3.3 Bus Master IDE Timings

The timing modes used for Bus Master IDE transfers are identical to those for PIO transfers. The DMA Timing Enable Only bits in IDE Timing register may be used to program fast timing mode for DMA transactions only. This is useful for IDE devices whose DMA transfer timings are faster that its PIO transfer timings. The IDE device DMA request signal is sampled on the same PCI clock that DIOR# or DIOW# is deasserted. When inactive, the DMA Acknowledge signal is deasserted on the next PCI clock and no more transfers take place until DMA request is asserted again.

5.14.3.4 Interrupts

Legacy Mode:

The Intel® 6300ESB ICH is connected to IRQ14 for the primary interrupt and IRQ15 for the secondary interrupt. This connection is done from the ISA pin, before any mask registers. This implies the following:

- Bus Master IDE is operating under an interrupt based driver. Therefore, it will not operate under environments where the IDE device drives an interrupt but the interrupt is masked in the system.
- Bus Master IDE devices are connected directly off of the Intel® 6300ESB ICH. IDE interrupts cannot be communicated through PCI devices or the serial stream.

Caution: In this mode, the Intel® 6300ESB ICH will not drive the PCI Interrupt associated with this function. That is only used in native mode.

Native Mode:

In this case both the Primary and Secondary channels share an interrupt. It will be internally connected to PIRQ[C]# (IRQ18 in APIC mode). The interrupt will be active-low and shared.

Behavioral notes in native mode

- The IRQ14 and IRQ15 pins do not affect the internal IRQ14 and IRQ15 inputs to the interrupt controllers. The IDE logic forces these signals inactive in such a way that the Serial IRQ source may be used.
- The IRQ14 and IRQ15 inputs (not external IRQ[14:15] pins) to the interrupt controller may come from other sources (Serial IRQ, PIRQx).
- The IRQ14 and IRQ15 pins are inverted from active-high to the active-low PIRQ.

- When switching the IDE controller to native mode, the IDE Interrupt Pin Register will be masked (see Section 9.1.19, "Offset 3Dh: INTR_PN—Interrupt Pin Register (IDE—D31:F1)"). When an interrupt occurs while the masking is in place and the interrupt is still active when the masking ends, the interrupt will be allowed to be asserted.
- The active-low PIRQ must be masked by hardware when the IOSE bit is cleared in order to allow other interrupts that are shared with this pin to be delivered and serviced. When the IOSE bit is 0, software may not clear the IDE interrupt status bits. When in Native Mode, a '1' in the Bus Master Interrupt status bit (bit 2 of BMISP/BMISS) forces the interrupt asserted. This bit must be cleared in order to deassert the interrupt. This implementation is different from the Legacy Mode.

5.14.3.5 Bus Master IDE Operation

To initiate a bus master transfer between memory and an IDE device, the following steps are required:

1. Software prepares a PRD Table in system memory. The PRD Table must be DWORD aligned and must not cross a 64-Kbyte boundary.
2. Software provides the starting address of the PRD Table by loading the PRD Table Pointer Register. The direction of the data transfer is specified by setting the Read/Write Control bit. The interrupt bit and Error bit in the Status register are cleared.
3. Software issues the appropriate DMA transfer command to the disk device.
4. The bus master function is engaged by software writing a '1' to the Start bit in the Command Register. The first entry in the PRD table is fetched and loaded into two registers which are not visible by software, the Current Base and Current Count registers. These registers hold the current value of the address and byte count loaded from the PRD table. The value in these registers is only valid when there is an active command to an IDE device.
5. Once the PRD is loaded internally, the IDE device will receive a DMA acknowledge.
6. The controller transfers data to/from memory responding to DMA requests from the IDE device. The IDE device and the host controller may or may not throttle the transfer several times. When the last data transfer for a region has been completed on the IDE interface, the next descriptor is fetched from the table. The descriptor contents are loaded into the Current Base and Current Count registers.
7. At the end of the transfer the IDE device signals an interrupt.
8. In response to the interrupt, software resets the Start/Stop bit in the command register. It then reads the controller status followed by the drive status to determine when the transfer completed successfully.

The last PRD in a table has the End of List (EOL) bit set. The PCI bus master data transfers will terminate when the physical region described by the last PRD in the table has been completely transferred. The active bit in the Status Register will be reset and the DDRQ signal will be masked.

The buffer is flushed (when in the write state) or invalidated (when in the read state) when a terminal count condition exists; that is, the current region descriptor has the EOL bit set and that region has been exhausted. The buffer is also flushed (write state) or invalidated (read state) when the Interrupt bit in the Bus Master IDE Status register is set. Software that reads the status register and finds the Error bit reset, and either the Active bit reset or the Interrupt bit set, may be assured that all data destined for system memory has been transferred and that data is valid in system memory. Table 82 describes how to interpret the Interrupt and Active bits in the Status Register after a DMA transfer has started.

During concurrent DMA or Ultra ATA transfers, the Intel® 6300ESB ICH IDE interface will arbitrate between the primary and secondary IDE cables when a PRD expires.

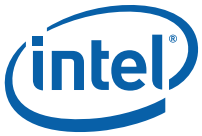


Table 82. Interrupt/Active Bit Interaction Definition

Interrupt	Active	Description
0	1	DMA transfer is in progress. No interrupt has been generated by the IDE device.
1	0	The IDE device generated an interrupt. The controller exhausted the Physical Region Descriptors. This is the normal completion case where the size of the physical memory regions was equal to the IDE device transfer size.
1	1	The IDE device generated an interrupt. The controller has not reached the end of the physical memory regions. This is a valid completion case where the size of the physical memory regions was larger than the IDE device transfer size.
0	0	This bit combination signals an error condition. When the Error bit in the status register is set, the controller has some problem transferring data to/from memory. Specifics of the error have to be determined using bus-specific information. When the Error bit is not set, then the PRD's specified a smaller size than the IDE transfer size.

5.14.3.6 Error Conditions

IDE devices are sector based mass storage devices. The drivers handle errors on a sector basis; either a sector is transferred successfully or it is not. A sector is 512 bytes.

When the IDE device does not complete the transfer due to a hardware or software error, the command will eventually be stopped by the driver setting Command Start bit to zero when the driver times out the disk transaction. Information in the IDE device registers help isolate the cause of the problem.

When the controller encounters an error while doing the bus master transfers, it will stop the transfer (i.e., reset the Active bit in the Command register) and set the Error bit in the Bus Master IDE Status register. The controller does not generate an interrupt when this happens. The device driver may use device specific information (PCI Configuration Space Status register and IDE Drive Register) to determine what caused the error.

Whenever a requested transfer does not complete properly, information in the IDE device registers (Sector Count) may be used to determine how much of the transfer was completed and to construct a new PRD table to complete the requested operation. In most cases the existing PRD table may be used to complete the operation.

5.14.3.7 8237-Like Protocol

8237 mode DMA is similar in form to DMA used on the ISA bus. This mode uses pins familiar to the ISA bus, namely a DMA Request, a DMA Acknowledge, and I/O read/write strobes. These pins have similar characteristics to their ISA counterparts in terms of when data is valid relative to strobe edges, and the polarity of the strobes, however the Intel® 6300ESB ICH does not use the 8237 for this mode.

5.14.4 Ultra ATA/33 Protocol

Ultra ATA/33 is enabled through configuration register 48h in Device 31:Function 1 for each IDE device. The IDE signal protocols are significantly different under this mode than for the 8237 mode. These differences allow the following enhancements to the transfer:

- A source synchronous protocol to allow higher data transfer rates of up to 33 Mbytes/s. The device that drives the data lines also drives the data strobe signal.
- Both the source and destination may pause the transfer. The source pauses the burst by not toggling its strobe signal, while the destination pauses the burst by deasserting a redefined signal, DMARDY#.
- 16 bit wide CRC error checking, sent from the Intel® 6300ESB ICH to the IDE device on DDACK# deassertion.

Ultra ATA/33 is a physical protocol used to transfer data between a Ultra ATA/33 capable IDE controller such as the Intel® 6300ESB ICH and one or more Ultra ATA/33 capable IDE devices. It utilizes the standard Bus Master IDE functionality and interface to initiate and control the transfer. Ultra ATA/33 utilizes a “source synchronous” signaling protocol to transfer data at rates up to 33 Mbytes/s. The Ultra ATA/33 definition also incorporates a Cyclic Redundancy Checking (CRC-16) error checking protocol.

5.14.4.1 Signal Descriptions

The Ultra ATA/33 protocol requires no extra signal pins on the IDE connector. It does redefine a number of the standard IDE control signals when in Ultra ATA/33 mode. These redefinitions are shown in the following table. Read cycles are defined as transferring data from the IDE device to the Intel® 6300ESB ICH. Write cycles are defined as transferring data from the Intel® 6300ESB ICH to IDE device.

Table 83. UltraATA/33 Control Signal Redefinitions

Standard IDE Signal Definition	Ultra ATA/33 Read Cycle Definition	Ultra ATA/33 Write Cycle Definition	Intel® 6300ESB ICH Primary Channel Signal	Intel® 6300ESB ICH Secondary Channel Signal
DIOW#	STOP	STOP	PDIOW#	SDIOW#
DIOR#	DMARDY#	STROBE	PDIOR#	SDIOR#
IORDY	STROBE	DMARDY#	PIORDY	SIORDY

The DIOW# signal is redefined as STOP for both read and write transfers. This is always driven by the Intel® 6300ESB ICH and is used to request that a transfer be stopped or as an acknowledgment to stop a request from the IDE device.

The DIOR# signal is redefined as DMARDY# for transferring data from the IDE device to the Intel® 6300ESB ICH (read). It is used by the Intel® 6300ESB ICH to signal when it is ready to transfer data and to add wait-states to the current transaction. The DIOR# signal is redefined as STROBE for transferring data from the Intel® 6300ESB ICH to the IDE device (write). It is the data strobe signal driven by the Intel® 6300ESB ICH on which data is transferred during each rising and falling edge transition.

The IORDY signal is redefined as STROBE for transferring data from the IDE device to the Intel® 6300ESB ICH (read). It is the data strobe signal driven by the IDE device on which data is transferred during each rising and falling edge transition. The IORDY signal is redefined as DMARDY# for transferring data from the Intel® 6300ESB ICH to the IDE device (write). It is used by the IDE device to signal when it is ready to transfer data and to add wait-states to the current transaction.

All other signals on the IDE connector retain their functional definitions during Ultra ATA/33 operation.

5.14.4.2 Operation

Initial setup programming consists of enabling and performing the proper configuration of the Intel® 6300ESB ICH and the IDE device for Ultra ATA/33 operation. For the Intel® 6300ESB ICH, this consists of enabling synchronous DMA mode and setting up appropriate Synchronous DMA timings.

When ready to transfer data to or from an IDE device, the Bus Master IDE programming model is followed. Once programmed, the drive and Intel® 6300ESB ICH control the transfer of data through the Ultra ATA/33 protocol. The actual data transfer consists of three phases, a start-up phase, a data transfer phase, and a burst termination phase.

The IDE device begins the start-up phase by asserting DMARQ signal. When ready to begin the transfer, the Intel® 6300ESB ICH will assert DMACK# signal. When DMACK# signal is asserted, the host controller will drive CS0# and CS1# inactive, DA0–DA2 low. For write cycles, the Intel® 6300ESB ICH will deassert STOP, wait for the IDE device to assert DMARDY#, and then drive the first data word and STROBE signal. For read cycles, the Intel® 6300ESB ICH will tri-state the DD lines, deassert STOP, and assert DMARDY#. The IDE device will then send the first data word and STROBE.

The data transfer phase continues the burst transfers with the data transmitter (Intel® 6300ESB ICH - writes, IDE device - reads) providing data and toggling STROBE. Data is transferred (latched by receiver) on each rising and falling edge of STROBE. The transmitter may pause the burst by holding STROBE high or low, resuming the burst by again toggling STROBE. The receiver may pause the burst by deasserting DMARDY# and resumes the transfers by asserting DMARDY#. The Intel® 6300ESB ICH will pause a burst transaction in order to prevent an internal line buffer over or under flow condition, resuming once the condition has cleared. It may also pause a transaction when the current PRD byte count has expired, resuming once it has fetched the next PRD.

Warning: The current burst may be terminated by either the transmitter or receiver. A burst termination consists of a Stop Request, Stop Acknowledge and transfer of CRC data. The Intel® 6300ESB ICH may stop a burst by asserting STOP, with the IDE device acknowledging by deasserting DMARQ. The IDE device stops a burst by deasserting DMARQ and the Intel® 6300ESB ICH acknowledges by asserting STOP. The transmitter then drives the STROBE signal to a high level. The Intel® 6300ESB ICH will then drive the CRC value onto the DD lines and deassert DMACK#. The IDE device will latch the CRC value on rising edge of DMACK#. The Intel® 6300ESB ICH will terminate a burst transfer when it needs to service the opposite IDE channel, when a Programmed I/O (PIO) cycle is executed to the IDE channel currently running the burst, or upon transferring the last data from the final PRD.

5.14.4.3 CRC Calculation

Cyclic Redundancy Checking (CRC-16) is used for error checking on Ultra ATA/33 transfers. The CRC value is calculated for all data by both the Intel® 6300ESB ICH and the IDE device over the duration of the Ultra ATA/33 burst transfer segment. This segment is defined as all data transferred with a valid STROBE edge from DDACK# assertion to DDACK# deassertion. At the end of the transfer burst segment, the Intel®

6300ESB ICH will drive the CRC value onto the DD[15:0] signals. It is then latched by the IDE device on deassertion of DDACK#. The IDE device compares the Intel® 6300ESB ICH CRC value to its own and reports an error if there is a mismatch.

5.14.5 Ultra ATA/66 Protocol

In addition to Ultra ATA/33, the Intel® 6300ESB ICH supports the Ultra ATA/66 protocol. The Ultra ATA/66 protocol is enabled through config bits 3:0 at offset 54h. The two protocols are similar, and are intended to be device driver compatible. The Ultra ATA/66 logic may achieve transfer rates of up to 66Mbytes/s.

In order to achieve the higher data rate, the timings are shortened and the quality of the cable is improved to reduce reflections, noise, and inductive coupling. Note that the improved cable is required and will still plug into the standard IDE connector.

The Ultra ATA/66 protocol also supports a 44 Mbytes/s mode.

5.14.6 Ultra ATA/100 Protocol

When the ATA_FAST bit is set for any of the four IDE devices, then the timings for the transfers to and from the corresponding device run at a higher rate. The Intel® 6300ESB ICH Ultra ATA/100 logic may achieve read transfer rates up to 100 Mbytes/s, and write transfer rates up to 88.9 Mbytes/s.

The cable improvements required for Ultra ATA/66 are sufficient for Ultra ATA/100, so no further cable improvements are required when implementing Ultra ATA/100.

5.14.7 Ultra ATA/33/66/100 Timing

The timings for Ultra ATA/33/66/100 modes are programmed through the Synchronous DMA Timing Register and the IDE Configuration Register. Different timings may be programmed for each drive in the system. The Base Clock frequency for each drive is selected in the IDE Configuration Register. The Cycle Time (CT) and Ready to Pause (RP) time (defined as multiples of the Base Clock) are programmed in the Synchronous DMA Timing Register. The Cycle Time represents the minimum pulse width of the data strobe (STROBE) signal. The Ready to Pause time represents the number of Base Clock periods that the Intel® 6300ESB ICH will wait from deassertion of DMARDY# to the assertion of STOP when it desires to stop a burst read transaction.

Note: The internal Base Clock for Ultra ATA/100 (Mode 5) runs at 133 MHz, and the Cycle Time (CT) must be set for three Base Clocks. The Intel® 6300ESB ICH will thus toggle the write strobe signal every 22.5 ns, transferring two bytes of data on each strobe edge. This means that the Intel® 6300ESB ICH will perform Mode 5 write transfers at a maximum rate of 88.9 Mbytes/s. For read transfers, the read strobe will be driven by the ATA/100 device, and the Intel® 6300ESB ICH supports reads at the maximum rate of 100 Mbytes/s.

5.15 SATA Host Controller (D31:F2)

5.15.1 Overview

The Intel® 6300ESB ICH SATA controller features two sets of interface signals that may be independently enabled, tri-stated or driven low. Each interface is supported by an independent DMA controller.

The Intel® 6300ESB ICH SATA controller interacts with an attached mass storage device through a register interface that is equivalent to that presented by a traditional IDE host adapter. The host software follows existing standards and conventions when accessing the register interface and follows standard command protocol conventions.

5.15.2 Theory of Operation

5.15.2.1 Standard ATA Emulation

The Intel® 6300ESB ICH contains a set of registers that shadow the contents of the legacy IDE registers. The behavior of the Command and Control Block registers, PIO and DMA data transfers, resets, and interrupts are all emulated.

5.15.2.2 48-bit LBA Operation (Logical Block Addressing)

The SATA host controller supports 48-bit LBA through the host-to-device register FIS, Frame Information Structure, when accesses are performed through writes to the task file. The SATA host controller will ensure that the correct data is put into the correct byte of the host-to-device FIS.

There are special considerations when reading from the task file to support 48-bit LBA operation. Software may need to read all 16 bits. Since the registers are only 8 bits wide and act as a FIFO, a bit must be set in the device/control register, which is at offset 3F6h for primary and 376h for secondary (or their native counterparts).

When software clears bit 7 of the control register before performing a read, the last item written will be returned from the FIFO. When software sets bit 7 of the control register before performing a read, the first item written will be returned from the FIFO.

5.15.3 Hot Plug Operation

Dynamic hot plug (such as surprise removal) is not supported by the SATA Host controller. However, using the SPC register configuration bits, and power management flows, a device may be powered down by software, and the port may then be powered off, allowing removal and insertion of a new device.

5.15.4 Power Management Operation

Power management of the Intel® 6300ESB ICH SATA Controller and ports will cover operations of the host controller and the SATA wire.

5.15.4.1 Power State Mappings

The following PCI power management states for devices are supported by the Intel® 6300ESB ICH SATA Controller:

D0 – working

D3 – very deep sleep. This state is split into two sub-states, $D3_{HOT}$ (may respond to PCI configuration accesses) and $D3_{COLD}$ (cannot respond to PCI configuration accesses). These two sub-states are considered the same, where $D3_{HOT}$ has V_{CC}, but $D3_{COLD}$ does not. This is the only state allowed for the host controller when the system is in an S1-S5 state.

SATA devices may also have multiple power states. From parallel ATA, three device states are supported through ACPI. They are:

D0 – Device is working and instantly available.

D1 – device enters when it receives a STANDBY IMMEDIATE command. Exit latency from this state is in seconds

D3 – from the SATA device's perspective, no different than a D1 state, in that it is entered through the STANDBY IMMEDIATE command. However, an ACPI method is also called which will reset the device and then cut its power.

Each of these device states are subsets of the host controller's D0 state.

Finally, SATA defines three PHY layer power states, which have no equivalent mappings to parallel ATA. They are:

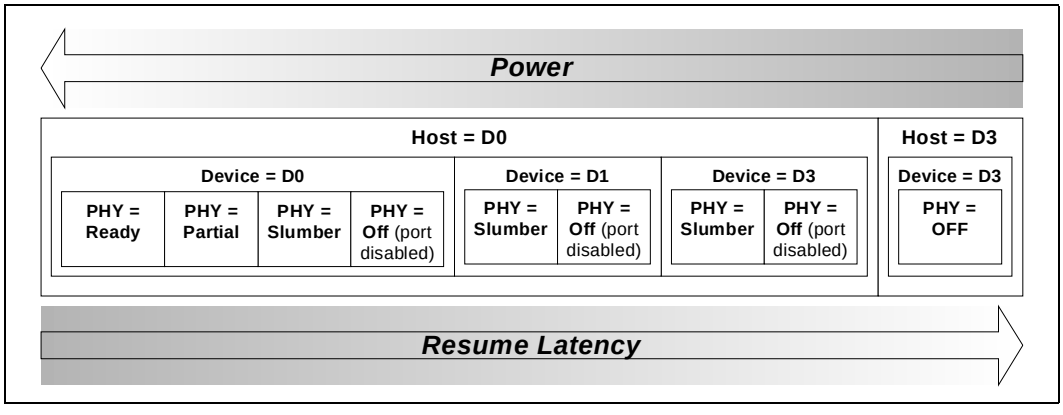
PHY READY – PHY logic and PLL are both on and active

Partial – PHY logic is powered, but in a reduced state. Exit latency is no longer than 10 ns

Slumber – PHY logic is powered, but in a reduced state. Exit latency may be up to 10 ms.

Since these states have much lower exit latency than the ACPI D1 and D3 states, the SATA Controller defines these states as sub-states of the device D0 state.

Figure 17. SATA Power States



5.15.4.2 Power State Transitions

5.15.4.2.1 Partial and Slumber State Entry/Exit

The partial and slumber states save interface power when the interface is idle. It would be most analogous to PCI CLKRUN# (in power savings, not in mechanism), where the interface may have power saved while no commands are pending.

The SATA Controller defines PHY layer power management (as performed through primitives) as a driver operation from the host side, and a device proprietary mechanism on the device side. The SATA Controller will accept device transition types, but will not issue any transitions as a host. All received requests from a SATA device will be ACKed.

When an operation is performed to the SATA Controller such that it needs to use the SATA cable, the controller must check whether the link is in the Partial or Slumber states, and if so, must issue a COM_WAKE to bring the link back online. Similarly, the SATA device must perform the same action.

5.15.4.2.2 Device D1, D3 States

These states are entered after some period of time when software has determined that no commands will be sent to this device for some time. The mechanism for putting a device in these states does not involve any work on the host controller, other then sending commands over the interface to the device. The command most likely to be used in ATA/ATAPI is the “STANDBY IMMEDIATE” command.

5.15.4.2.3 Host Controller D3 state

After the interface and device have been put into a low power state, the host controller may be put into a low power state. This is performed through the PCI power management registers in configuration space.

There are two very important aspects to note when using PCI power management.

1. When the power state is D3, only accesses to configuration space are allowed. Any attempt to access the memory or I/O spaces must result in master abort.
2. When the power state is D3, no interrupts may be generated, even when they are enabled. When an interrupt status bit is pending when the controller transitions to D0, an interrupt may be generated.

When the controller is put into D3, it is assumed that software has properly shut down the device and disabled the ports. Therefore, there is no need to sustain any values on the port wires. The interface will be treated as though no device is present on the cable, and power will be minimized.

5.15.4.3 SMI Trapping (APM)

Offset 48h, bits 3:0 in the power management I/O space contain control for generating SMI# on accesses to the IDE I/O spaces. These bits map to the legacy ranges only (1f0-1f7h, 3f6h, 170-177h, and 376h). When the SATA controller is in legacy mode and is using these addresses, accesses to one of these ranges with the appropriate bit set will cause the cycle to not be forwarded to the SATA controller, and an SMI# is generated.

To block accesses to the native IDE ranges, software must use the generic Power Management control registers described in Section 8.8.1.7, “Offset C4h, C6h, C8h, CAh: MON[n]_TRP_RNG—I/O Monitor [4:7] Trap Range Register for Devices 4-7 (PM—D31:F0)”.

5.15.5 SATA Interrupts

The following table summarizes interrupt behavior for MSI and wire-modes. In the table "bits" refers to the 4 possible interrupt bits in I/O space, which are: BMISP.PRDIS (offset 02h, bit 7), BMISP.I (offset 02h, bit 2), BMISS.PRDIS (offset 0Ah, bit 7), and BMISS.I (offset 0Ah, bit 2). See Section 20.2, "Bus Master IDE I/O Registers (D31:F2)" for I/O space register details.

Table 84. SATA MSI vs. PCI IRQ Actions

Interrupt Register	Wire-Mode Action	MSI Action
All bits are '0'.	Wire Inactive	No Action
One or more bits set to '1'.	Wire Active	Send Message
One or more bits set to '1', new bit gets set to '1'.	Wire Active	Send Message
One or more bits set to '1', software clears some (but not all) bits.	Wire Active	Send Message
One or more bits set to '1', software clears all bits.	Wire Inactive	No Action
Software clears one or more bits, and one or more bits is set simultaneously.	Wire Active	Send Message

5.15.6 SATALED#

The SATALED# pin is driven low to indicate SATA drive activity. When SATALED# is asserted, the LED is active.

5.16 Multimedia Event Timers

5.16.1 Overview

This function provides a set of timers that may be used by the operating system. The timers are defined such that in the future, the OS may be able to assign specific timers to be used directly by specific applications. Each timer may be configured to cause a separate interrupt. This specification allows for a block of 32 timers, with support for up to eight blocks, for a total of 256 timers. However, specific implementations may include only a subset of these timers.

The Intel® 6300ESB ICH provides three timers. The three timers are implemented as a single counter each with its own comparator and value register. Each timer's counter increases monotonically. Each individual timer may generate an interrupt when the value in its value register matches the value in the main counter. Some of the timers may be enabled to generate a periodic interrupt.

The registers associated with these timers are mapped to a memory space (much like the I/O APIC). However, it is not implemented as a standard PCI function. The BIOS reports to the operating system the location of the register space. The hardware may support an assignable decode space, however the BIOS will set this space prior to handing it over to the OS (see Section 6.4, "Memory Map"). It is not expected that the OS will move the location of these timers once it is set by the BIOS.

In the Intel® 6300ESB ICH, one timer block is implemented. The timer block has one counter and three timers (comparators). Future devices may have a different number of implemented timers. Various capabilities registers indicate the number of timers and the capabilities of each.

5.16.2 Timer Accuracy

1. The timers are accurate over any 1 ms period to within 0.005% of the time specified in the timer resolution fields.
2. Within any 100 ms period, the timer will report a time that is up to two ticks too early or too late. Each tick is less than or equal to 100 ns, so this represents an error of less than 0.2%.
3. The timer is monotonic. It will not return the same value on two consecutive reads (unless the counter has rolled over and reached the same value).

The main counter will be clocked by the 14.31818 MHz clock, synchronized into the 66.666 MHz domain. This will result in a non-uniform duty cycle on the synchronized clock, but does have the correct average period. The main counter will be as accurate as the 14.3818 MHz clock.

5.16.3 Interrupt Mapping

Mapping Option #1: Legacy Option

In this case, the Legacy Rout bit (LEG_RT_CNF) will be set. This will force the mapping found in Table 85. See Section 15.1.3, "Offset 010-017h: General Config Register" for LEG_RT_CNF details.

Table 85. Legacy Routing

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	In this case, the 8254 timer will not cause any interrupts.
1	IRQ8	IRQ8	In this case, the RTC will not cause any interrupts.
2	As per IRQ Routing Field	As per IRQ Routing Field	

Mapping Option #2: Standard Option

In this case, the Legacy Rout bit (LEG_RT_CNF) will be zero. Each timer has its own routing control. The supported interrupt values are IRQ 20, 21, 22, and 23. See Section 15.1.3, "Offset 010-017h: General Config Register" for LEG_RT_CNF details.

5.16.4 Periodic vs. Non-Periodic Modes

Non-Periodic Mode

When a timer is set up for non-periodic mode, it will generate a value in the main counter which matches the value in the timer's comparator register. When the timer is set up for 32-bit mode, it will generate another interrupt when the main counter wraps around and matches this same value again. Timer 0 is configurable to 32 (default) or 64-bit mode, whereas Timers 1 and 2 only support 32-bit mode.

During run-time, the value in the timer's comparator value register will not be changed by the hardware. Software may change the value.

Warning: Software must be careful when programming the comparator registers. When the value written to the register is not sufficiently far in the future, the counter

may pass the value before it reaches the register and the interrupt will be missed.

All three timers support non-periodic mode.

Periodic Mode

Timer 0 is the only timer that supports periodic mode. When Timer 0 is set up for periodic mode, the software writes a value into the timer's comparator value register. When the main counter value matches the value in the timer's comparator value register, an interrupt may be generated. The hardware will then automatically increase the value in the comparator value register by the last value written to that register.

To make the periodic mode work properly, the main counter is typically written with a value of zero so that the first interrupt occurs at the right point for the comparator. When the main counter is not set to zero, interrupts may not occur as expected.

During run-time, the value in the timer's comparator value register may be read by software to find out when the next periodic interrupt will be generated (not the rate at which it generates interrupts). Software is expected to remember the last value written to the comparator's value register (the rate at which interrupts are generated).

When software wants to change the periodic rate, it should write a new value to the comparator value register. At the point when the timer's comparator indicates a match, this new value will be added to derive the next matching point.

When the software resets the main counter, the value in the comparator's value register needs to be reset as well. This may be done by setting the TIMER0_VAL_SET_CNF bit. Again, to avoid race conditions, this should be done with the main counter halted. See Section 15, "Multimedia Timer Registers" for register and bits details.

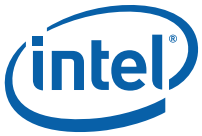
The following usage model is expected:

1. Software clears the ENABLE_CNF bit to prevent any interrupts
2. Software Clears the main counter by writing a value of 00h to it.
3. Software sets the TIMER0_VAL_SET_CNF bit.
4. Software writes the new value in the TIMER0_COMPARATOR_VAL register
5. Software sets the ENABLE_CNF bit to enable interrupts.

Warning: As the timer period approaches zero, the interrupts associated with the periodic timer may not get completely serviced before the next timer match occurs. Interrupts may get lost and/or system performance may be degraded in this case.

The Timer 0 Comparator Value register cannot be programmed reliably by a single 64-bit write in a 32-bit environment except when only the periodic rate is being changed during run-time. When the actual Timer 0 Comparator Value needs to be reinitialized, the following software solution will always work regardless of the environment:

1. Set TIMER0_VAL_SET_CNF bit
2. Set the lower 32 bits of the Timer0 Comparator Value register
3. Set TIMER0_VAL_SET_CNF bit
4. Set the upper 32 bits of the Timer0 Comparator Value register



5.16.5 Enabling the Timers

The BIOS or OS PnP code should route the interrupts. This includes the Legacy Rout bit, Interrupt Rout bit (for each timer), interrupt type (to select the edge or level type for each timer)

The Device Driver code should do the following for an available timer:

1. Set the Overall Enable bit (Offset 04h, bit 0).
2. Set the timer type field (selects one-shot or periodic).
3. Set the interrupt enable.
4. Set the comparator value.

5.16.6 Interrupt Levels

Interrupts directed to the internal 8259s are active high. See Section 5.7, “Advanced Interrupt Controller (APIC) (D29:F5)” for information regarding the polarity programming of the
I/O APIC for detecting internal interrupts. When the interrupts are mapped to the I/O APIC and set for level-triggered mode, they may be shared with PCI interrupts, although it is unlikely for the OS to attempt this. When more than one timer is configured to share the same IRQ using the TIMERn_INT_ROUT_CNF fields, the software must configure the timers to level-triggered mode. Edge-triggered interrupts cannot be shared.

5.16.7 Handling Interrupts

When each timer has a unique interrupt and the timer has been configured for edge-triggered mode, no specific steps are required. No read is required to process the interrupt.

When a timer has been configured to level-triggered mode, its interrupt must be cleared by the software. This is done by reading the interrupt status register and writing a 1 back to the bit position for the interrupt to be cleared.

Independent of the mode, software may read the value in the main counter to see how much time has passed between when the interrupt was generated and when it was first serviced.

When Timer 0 is set up to generate a periodic interrupt, the software may check to see how much time remains until the next interrupt by checking the timer value register.

5.16.8 Issues Related to 64-bit Timers with 32-bit Processors

A 32-bit timer may be read directly using processors that are capable of 32-bit or 64-bit instructions. However, a 32-bit processor may not be able to directly read 64-bit timer. A race condition comes up when a 32-bit processor reads the 64-bit register using two separate 32-bit reads. The danger is that just after reading one half, the other half rolls over and changes the first half.

When a 32-bit processor needs to access a 64-bit timer, it must first halt the timer before reading both the upper and lower 32-bits of the timer.

When a 32-bit processor does not want to halt the timer, it may use the 64-bit timer as a 32-bit timer by setting the TIMERn_32MODE_CNF bit. This will cause the timer to behave as a 32-bit timer. The upper 32 bits will always be 0.

5.17 USB UHCI Controllers (D29:F0 and F1)

5.17.1 Overview

The Intel® 6300ESB ICH contains two USB UHCI Host Controllers. Each Host Controller includes a root hub with two separate USB ports each, for a total of four USB ports. The Intel® 6300ESB ICH Host Controllers support the standard *Universal Host Controller Interface (UHCI) Specification,* Rev 1.1.

Overcurrent detection on all four USB ports is supported. The overcurrent inputs are 5V-tolerant and may be used as GPIs when not needed.

The Intel® 6300ESB ICH's USB UHCI controllers are arbitrated differently from standard PCI devices to improve arbitration latency.

The USB UHCI controllers use the Analog Front End (AFE) embedded cell that allows support for USB High-speed signaling rates instead of USB I/O buffers.

5.17.2 Data Structures in Main Memory

This section describes the details of the data structures used to communicate control, status, and data between software and the Intel® 6300ESB ICH: Frame Lists, Transfer Descriptors, and Queue Heads. Frame Lists are aligned on 4-Kbyte boundaries. Transfer Descriptors and Queue Heads are aligned on 16-byte boundaries.

5.17.2.1 Frame List Pointer

The frame list pointer contains a link pointer to the first data object to be processed in the frame, as well as the control bits defined in Table 86.

Table 86. Frame List Pointer Bit Description

Bit	Description
31:4	**Frame List Pointer (FLP):** This field contains the address of the first data object to be processed in the frame and corresponds to memory address signals [31:4], respectively.
3:2	Reserved. These bits must be written as zero.
1	**QH/TD Select (Q):** This bit indicates to the hardware whether the item referenced by the link pointer is a TD (Transfer Descriptor) or a QH (Queue Head). This allows the Intel® 6300ESB ICH to perform the proper type of processing on the item after it is fetched. 1 = QH 0 = TD
0	**Terminate (T):** This bit indicates to the Intel® 6300ESB ICH whether the schedule for this frame has valid entries in it. 1 = Empty Frame (pointer is invalid). 0 = Pointer is valid (points to a QH or TD).

5.17.2.2 Transfer Descriptors (TD)

Transfer Descriptors (TDs) express the characteristics of the transaction requested on USB by a client. TDs are always aligned on 16-byte boundaries, and the elements of the TD are shown in Figure 18. The four different USB transfer types are supported by a small number of control bits in the descriptor that the Intel® 6300ESB ICH interprets during operation. All Transfer Descriptors have the same basic, 32-byte structure. During operation, the Intel® 6300ESB ICH hardware performs consistency checks on some fields of the TD. When a consistency check fails, the Intel® 6300ESB ICH halts immediately and issues an interrupt to the system. This interrupt cannot be masked within the Intel® 6300ESB ICH.

Figure 18. Transfer Descriptor

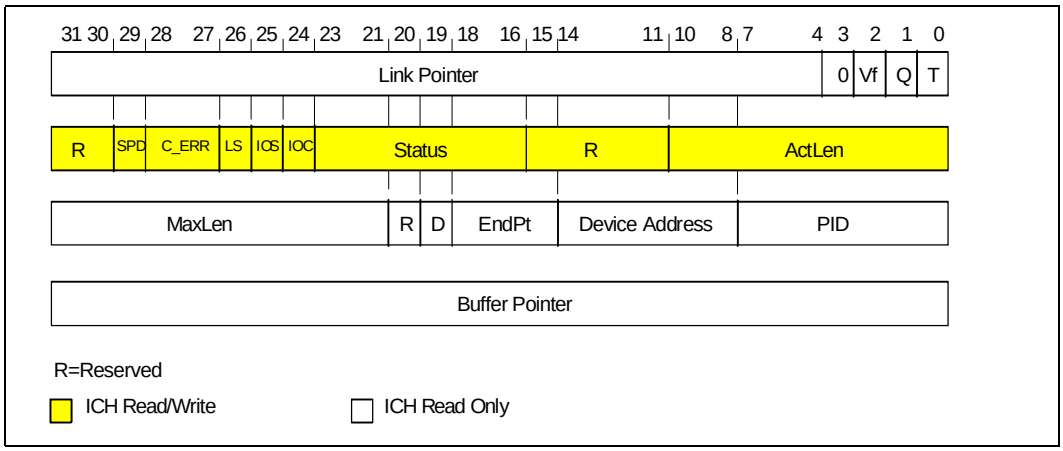


Table 87. TD Link Pointer

Bit	Description
31:4	**Link Pointer (LP):** Bits [31:4] Correspond to memory address signals [31:4], respectively. This field points to another TD or QH.
3	Reserved. Must be 0 when writing this field.
2	**Depth/Breadth Select (VF):** This bit is only valid for queued TDs and indicates to the hardware whether it should process in a depth first or breadth first fashion. When set to depth first, it informs the Intel® 6300ESB ICH to process the next transaction in the queue rather than starting a new queue. 0 = Breadth first 1 = Depth first
1	**QH/TD Select (Q):** This bit informs the Intel® 6300ESB ICH whether the item referenced by the link pointer is another TD or a QH. This allows the Intel® 6300ESB ICH to perform the proper type of processing on the item after it is fetched. 0 = TD 1 = QH
0	**Terminate (T):** This bit informs the Intel® 6300ESB ICH that the link pointer in this TD does not point to another valid entry. When encountered in a queue context, this bit indicates to the Intel® 6300ESB ICH that there are no more valid entries in the queue. A TD encountered outside of a queue context with the T bit set informs the Intel® 6300ESB ICH that this is the last TD in the frame. 0 = Link Pointer field is valid. 1 = Link Pointer field not valid.

Table 88. TD Control and Status (Sheet 1 of 3)

Bit	Description
31:30	Reserved.
29	**Short Packet Detect (SPD):** When a packet has this bit set to 1 and the packet is an input packet, is in a queue; and successfully completes with an actual length less than the maximum length then the TD is marked inactive, the Queue Header is not updated and the USBINT status bit (Status Register) is set at the end of the frame. In addition, when the interrupt is enabled, the interrupt will be sent at the end of the frame. Note that any error (e.g., babble or FIFO error) prevents the short packet from being reported. The behavior is undefined when this bit is set with output packets or packets outside of queues. 0 = Disable 1 = Enable
28:27	**Error Counter (C_ERR):** This field is a 2-bit down counter that keeps track of the number of Errors detected while executing this TD. When this field is programmed with a non zero value during setup, the Intel® 6300ESB ICH decrements the count and writes it back to the TD when the transaction fails. When the counter counts from one to zero, the Intel® 6300ESB ICH marks the TD inactive, sets the “STALLED” and error status bit for the error that caused the transition to zero in the TD. An interrupt will be generated to Host Controller Driver (HCD) when the decrement to zero was caused by Data Buffer error, Bit stuff error, or when enabled, a CRC or Timeout error. When HCD programs this field to zero during setup, the Intel® 6300ESB ICH will not count errors for this TD and there will be no limit on the retries of this TD. **Bits[28:27]Interrupt After** 00 No Error Limit 01 1 Error 10 2 Errors 11 3 Errors **Error** / **Decrement Counter** / **Error** / **Decrement Counter** CRC Error — Yes; Data Buffer Error — Yes Timeout Error — Yes; Stalled — No[1] NAK Received — No; Bit stuff Error — Yes Babble Detected — No† † Detection of Babble or Stall automatically deactivates the TD. Thus, count is not decremented.
26	**Low Speed Device (LS):** This bit indicates that the target device (USB data source or sink) is a low speed device, running at 1.5 Mb/s, instead of at full speed (12 Mb/s). There are special restrictions on schedule placement for low speed TDs. When an Intel® 6300ESB ICH root hub port is connected to a full speed device and this bit is set to a 1 for a low speed transaction, the Intel® 6300ESB ICH sends out a low speed preamble on that port before sending the PID. No preamble is sent when a Intel® 6300ESB ICH root hub port is connected to a low speed device. 0 = Full Speed Device 1 = Low Speed Device
25	**Isochronous Select (IOS):** The field specifies the type of the data structure. When this bit is set to a 1, then the TD is an isochronous transfer. Isochronous TDs are always marked inactive by the hardware after execution, regardless of the results of the transaction. 0 = Non-isochronous Transfer Descriptor 1 = Isochronous Transfer Descriptor
24	**Interrupt on Complete (IOC):** This specifies that the Intel® 6300ESB ICH should issue an interrupt on completion of the frame in which this Transfer Descriptor is executed. Even when the Active bit in the TD is already cleared when the TD is fetched (no transaction will occur on USB), an IOC interrupt is generated at the end of the frame. 1 = Issue IOC

Table 88. TD Control and Status (Sheet 2 of 3)

Bit	Description
23	**Active:** For Intel® 6300ESB ICH schedule execution operations, see Section 5.17.3, "Data Transfers to/from Main Memory", Data Transfers to/from Main Memory. 0 = When the transaction associated with this descriptor is completed, the Intel® 6300ESB ICH sets this bit to 0 indicating that the descriptor should not be executed when it is next encountered in the schedule. The Active bit is also set to 0 when a stall handshake is received from the endpoint. 1 = Set to 1 by software to enable the execution of a message transaction by the Intel® 6300ESB ICH.
22	**Stalled:** 1 = Set to a 1 by the Intel® 6300ESB ICH during status updates to indicate that a serious error has occurred at the device/endpoint addressed by this TD. This may be caused by babble, the error counter counting down to zero, or reception of the STALL handshake from the device during the transaction. Any time that a transaction results in the Stalled bit being set, the Active bit is also cleared (set to 0). When a STALL handshake is received from a SETUP transaction, a time-out error will also be reported.
21	**Data Buffer Error (DBE):** 1 = Set to a 1 by the Intel® 6300ESB ICH during status update to indicate that the Intel® 6300ESB ICH is unable to keep up with the reception of incoming data (overrun) or is unable to supply data fast enough during transmission (underrun). When this occurs, the actual length and Max Length field of the TD will not match. In the case of an underrun, the Intel® 6300ESB ICH will transmit an incorrect CRC (thus invalidating the data at the endpoint) and leave the TD active (unless error count reached zero). When an overrun condition occurs, the Intel® 6300ESB ICH will force a timeout condition on the USB, invalidating the transaction at the source.
20	**Babble Detected (BABD):** 1 = Set to a 1 by the Intel® 6300ESB ICH during status update when "babble" is detected during the transaction generated by this descriptor. Babble is unexpected bus activity for more than a preset amount of time. In addition to setting this bit, the Intel® 6300ESB ICH also sets the "STALLED" bit (bit 22) to a 1. Since "babble" is considered a fatal error for that transfer, setting the "STALLED" bit to a 1 insures that no more transactions occur as a result of this descriptor. Detection of babble causes immediate termination of the current frame. No further TDs in the frame are executed. Execution resumes with the next frame list index.
19	**Negative Acknowledgment (NAK) Received (NAKR):** 1 = Set to a 1 by the Intel® 6300ESB ICH during status update when the Intel® 6300ESB ICH receives a "NAK" packet during the transaction generated by this descriptor. When a NAK handshake is received from a SETUP transaction, a time-out error will also be reported.
18	**CRC/time-out error (CRC_TOUT):** 1 = Set to a 1 by the Intel® 6300ESB ICH as follows: During a status update in the case that no response is received from the target device/endpoint within the time specified by the protocol chapter of the USB specification. During a status update when a Cyclic Redundancy Check (CRC) error is detected during the transaction associated with this transfer descriptor. In the transmit case (OUT or SETUP Command), this is in response to the Intel® 6300ESB ICH detecting a timeout from the target device/endpoint. In the receive case (IN Command), this is in response to the Intel® 6300ESB ICH's CRC checker circuitry detecting an error on the data received from the device/endpoint or a NAK or STALL handshake being received in response to a SETUP transaction.
17	**Bit stuff Error (BSE):** 1 = This bit is set to a 1 by the Intel® 6300ESB ICH during status update to indicate that the receive data stream contained a sequence of more than 6 ones in a row.

Table 88. TD Control and Status (Sheet 3 of 3)

Bit	Description
16	**Bus Turn Around Time-out (BTTO):** 1 = This bit is set to a 1 by the Intel® 6300ESB ICH during status updates to indicate that a bus time-out condition was detected for this USB transaction. This time-out is specially defined as not detecting an IDLE-to 'K' state Start of Packet (SOP) transition from 16 to 18 bit times after the SE0-to-IDE transition of previous End of Packet (EOP).
15:11	Reserved
10:0	**Actual Length (ACTLEN):** The Actual Length field is written by the Intel® 6300ESB ICH at the conclusion of a USB transaction to indicate the actual number of bytes that were transferred. It may be used by the software to maintain data integrity. The value programmed in this register is encoded as n-1 (see Maximum Length field description in the TD Token).

Table 89. TD Token

Bit	Description
31:21	**Maximum Length (MAXLEN):** The Maximum Length field specifies the maximum number of data bytes allowed for the transfer. The Maximum Length value does not include protocol bytes, such as Packet ID (PID) and CRC. The maximum data packet is 1280 bytes. The 1280 packet length is the longest packet theoretically ensured to fit into a frame. Actual packet maximum lengths are set by HCD according to the type and speed of the transfer. Note that the maximum length allowed by the USB specification is 1023 bytes. The valid encodings for this field are: 0x000 = 1 byte 0x001 = 2 bytes .... 0x3FE = 1023 bytes 0x3FF = 1024 bytes .... 0x4FF = 1280 bytes 0x7FF = 0 bytes (null data packet) Note that values from 500h to 7FEh are illegal and cause a consistency check failure. In the transmit case, the Intel® 6300ESB ICH uses this value as a terminal count for the number of bytes it fetches from host memory. In most cases, this is the number of bytes it will actually transmit. In rare cases, the Intel® 6300ESB ICH may be unable to access memory (e.g., due to excessive latency) in time to avoid underrunning the
20	Reserved.
19	**Data Toggle (D):** This bit is used to synchronize data transfers between a USB endpoint and the host. This bit determines which data PID is sent or expected (0=DATA0 and 1=DATA1). The Data Toggle bit provides a 1-bit sequence number to check whether the previous packet completed. This bit must always be 0 for Isochronous TDs.
18:15	**Endpoint (ENDPT):** This 4-bit field extends the addressing internal to a particular device by providing 16 endpoints. This permits more flexible addressing of devices in which more than one sub-channel is required.
14:8	**Device Address:** This field identifies the specific device serving as the data source or sink.
7:0	**Packet Identification (PID):** This field contains the Packet ID to be used for this transaction. Only the IN (69h), OUT (E1h), and SETUP (2Dh) tokens are allowed. Any other value in this field causes a consistency check failure resulting in an immediate halt of the Intel® 6300ESB ICH. Bits [3:0] are complements of bits [7:4].

Table 90. TD Buffer Pointer

Bit	Description
31:0	**Buffer Pointer (BUFF_PNT):** Bits [31:0] corresponds to memory address [31:0], respectively. It points to the beginning of the buffer that will be used during this transaction. This buffer must be at least as long as the value in the Maximum Length field described int the TD token. The data buffer may be byte-aligned.

5.17.2.3 Queue Head (QH)

Queue heads are special structures used to support the requirements of Control, Bulk, and Interrupt transfers. Since these TDs are not automatically retired after each use, their maintenance requirements may be reduced by putting them into a queue. Queue Heads must be aligned on a 16-byte boundary, and the elements are shown in Table 91.

Table 91. Queue Head Block

Bytes	Description	Attributes
00-03	Queue Head Link Pointer	RO
04-07	Queue Element Link Pointer	R/W

Table 92. Queue Head Link Pointer

Bit	Description
31:4	**Queue Head Link Pointer (QHLP):** This field contains the address of the next data object to be processed in the horizontal list and corresponds to memory address signals [31:4], respectively.
3:2	Reserved. These bits must be written as zeros.
1	**QH/TD Select (Q):** This bit indicates to the hardware whether the item referenced by the link pointer is another TD or a QH. 0 = TD 1 = QH
0	**Terminate (T):** This bit indicates to the Intel® 6300ESB ICH that this is the last QH in the schedule. When there are active TDs in this queue, they are the last to be executed in this frame. 0 = Pointer is valid (points to a QH or TD). 1 = Last QH (pointer is invalid).

Table 93. Queue Element Link Pointer (Sheet 1 of 2)

Bit	Description
31:4	**Queue Element Link Pointer (QELP):** This field contains the address of the next TD or QH to be processed in this queue and corresponds to memory address signals [31:4], respectively.

Table 93. Queue Element Link Pointer (Sheet 2 of 2)

Bit	Description
3:2	Reserved.
1	**QH/TD Select (Q):** This bit indicates to the hardware whether the item referenced by the link pointer is another TD or a QH. For entries in a queue, this bit is typically set to zero. 0 = TD 1 = QH
0	**Terminate (T):** This bit indicates to the Intel® 6300ESB ICH that there are no valid TDs in this queue. When HCD has new queue entries it overwrites this value with a new TD pointer to the queue entry. 0 = Pointer is valid. 1 = Terminate (No valid queue entries).

5.17.3 Data Transfers to/from Main Memory

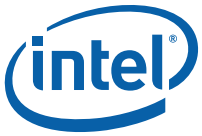
The following sections describe the details on how HCD and the Intel® 6300ESB ICH communicate through the schedule data structures. The discussion is organized in a top-down manner, beginning with the basics of walking the Frame List, followed by a description of generic processing steps common to all transfer descriptors, and finally a discussion on Transfer Queuing.

5.17.3.1 Executing the Schedule

Software programs the Intel® 6300ESB ICH with the starting address of the Frame List and the Frame List index, then causes the Intel® 6300ESB ICH to execute the schedule by setting the Run/Stop bit in the Control register to Run. The Intel® 6300ESB ICH processes the schedule one entry at a time; the next element in the frame list is not fetched until the current element in the frame list is retired.

Schedule execution proceeds in the following fashion:

1. The Intel® 6300ESB ICH first fetches an entry from the Frame List. This entry has three fields. Bit 0 indicates whether the address pointer field is valid. Bit 1 indicates whether the address points to a Transfer Descriptor or to a queue head. The third field is the pointer itself.
2. When isochronous traffic is to be moved in a given frame, the Frame List entry points to a Transfer Descriptor. When no isochronous data is to be moved in that frame, the entry points to a queue head or the entry is marked invalid and no transfers are initiated in that frame.
3. When the Frame List entry indicates that it points to a Transfer Descriptor, the Intel® 6300ESB ICH fetches the entry and begins the operations necessary to initiate a transaction on USB. Each TD contains a link field that points to the next entry, as well as indicating whether it is a TD or a QH.
4. When the Frame List entry contains a pointer to a QH, the Intel® 6300ESB ICH processes the information from the QH to determine the address of the next data object that it should process.
5. The TD/QH process continues until the millisecond allotted to the current frame expires. At this point, the Intel® 6300ESB ICH fetches the next entry from the Frame List. When the Intel® 6300ESB ICH is not able to process all of the transfer descriptors during a given frame, those descriptors are retired by software without having been executed.



5.17.3.2 Processing Transfer Descriptors

The Intel® 6300ESB ICH executes a TD using the following generalized algorithm. These basic steps are common across all modes of TDs. Subsequent sections present processing steps unique to each TD mode.

1. The Intel® 6300ESB ICH fetches TD or QH from the current Link Pointer.
2. When a QH, go to 1 to fetch from the Queue Element Link Pointer. When inactive, go to 12.
3. Build token, actual bits are in TD token.
4. When (Host-to-Function) then
   [*PCI Access*] issue request for data, (referenced through TD.BufferPointer)
   wait for first chunk data arrival
   end if
5. [*Begin USB Transaction*] Issue token (from token built in 2, above) and begin data transfer.
   if (Host-to-Function) then Go to 6
   else Go to 7
   end if
6. Fetch data from memory (through TD BufferPointer) and transfer over USB until TD Max-Length bytes have been read and transferred. [*Concurrent system memory and USB Accesses*]. Go to 8.
7. Wait for data to arrive (from USB). Write incoming bytes into memory beginning at TD BufferPointer. Internal HC buffer should signal end of data packet. Number of bytes received must be TD Max-Length; The length of the memory area referenced by TD BufferPointer.
8. Issue handshake based on status of data received (Ack or Time-out). Go to 10.
9. Wait for handshake, when required [*End of USB Transaction*].
10. Update Status [*PCI Access*] (TD.Status and TD.ActualLength).
    When the TD was an isochronous TD, mark the TD inactive. Go to 12.
    When not an isochronous TD, and TD completed successfully, mark the TD inactive. Go to 11.
    When not successful, and the error count has not been reached, leave the TD active. When the error count has been reached, mark the TD inactive. Go to 12.
11. Write the link pointer from the current TD into the element pointer field of the QH structure. When the Vf bit is set in the TD link pointer, go to 2.
12. Proceed to next entry.

5.17.3.3 Command Register, Status Register, and TD Status Bit Interaction

Table 94. Command Register, Status Register and TD Status Bit Interaction

Condition	Intel® 6300ESB ICH USB Status Register Actions	TD Status Register Actions
CRC/time-out error	Set USB Error Int bit[1], Clear HC Halted bit	Clear Active bit[1] and set Stall bit[1].
Illegal PID, PID Error, Max Length (illegal)	Clear Run/Stop bit in command register Set HC Process Error and HC Halted bits	
PCI Master/Target Abort	Clear Run/Stop bit in command register Set Host System Error and HC Halted bits	
Suspend Mode	Clear Run/Stop bit in command register[2] Set HC Halted bit	
Resume Received and Suspend Mode = 1	Set Resume received bit	
Run/Stop = 0	Clear Run/Stop bit in command register Set HC Halted bit	
Config Flag Set	Set Config Flag in command register	
HC Reset/Global Reset	Clear Run/Stop and Config Flag in command register Clear USB Int, USB Error Int, Resume received, Host System Error, HC Process Error, and HC Halted bits	
IOC = 1 in TD Status	Set USB Int bit	
Stall	Set USB Error Int bit	Clear Active bit[1] and set Stall bit.
Bit Stuff/Data Buffer Error	Set USB Error Int bit[1].	Clear Active bit[1] and set Stall bit[1].
Short Packet Detect	Set USB Int bit	Clear Active bit

NOTES:
1. Only when error counter counted down from 1 to 0.
2. Suspend mode may be entered only when Run/Stop bit is 0.

Note that when a NAK or STALL response is received from a SETUP transaction, a Time-Out Error will be reported. This will cause the Error counter to decrement and the CRC/Time-Out Error status bit to be set within the TD Control and Status DWORD during write back. When the Error counter changes from 1 to 0, the Active bit will be reset to 0 and Stalled bit to 1 as normal.

5.17.3.4 Transfer Queuing

Transfer Queues are used to implement an ensured data delivery stream to a USB Endpoint. Transfer Queues are composed of two parts: a Queue Header (QH) and a linked list. The linked list of TDs and QHs has an indeterminate length (0 to n).

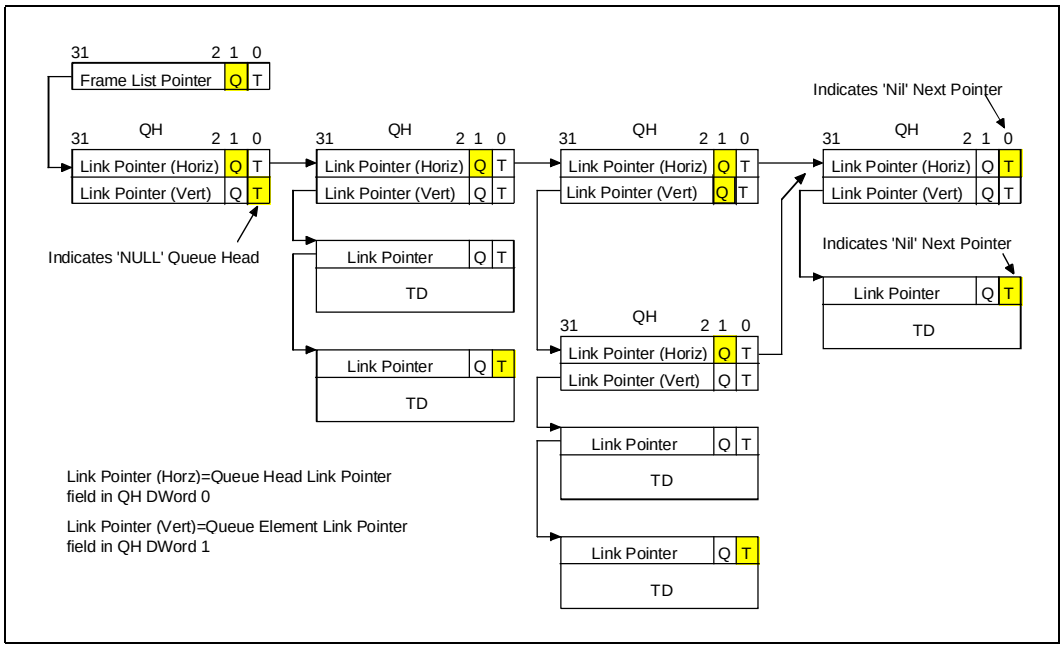
The QH contains two link pointers and is organized as two contiguous DWORDs. The first DWORD is a horizontal pointer (Queue Head Link Pointer), used to link a single transfer queue with either another transfer queue, or a TD (target data structure depends on Q bit). When the T bit is set, this QH represents the last data structure in the current Frame. The T bit informs the Intel® 6300ESB ICH that no further processing is required until the beginning of the next frame. The second DWORD is a vertical pointer (Queue Element Link Pointer) to the first data structure (TD or QH) being managed by this QH. When the T bit is set, the queue is empty. This pointer may reference a TD or another QH.

Figure 19 illustrates four example queue conditions. The first QH (on far left) is an example of an “empty” queue; the termination bit (T Bit), in the vertical link pointer field, is set to 1. The horizontal link pointer references another QH. The next queue is the expected typical configuration. The horizontal link pointer references another QH, and the vertical link pointer references a valid TD.

Typically, the vertical pointer in a QH points to a TD. However, as shown in Figure 19 (third example from left side of figure) the vertical pointer could point to another QH. When this occurs, a new Q Context is entered and the Q Context just exited is NULL (The Intel® 6300ESB ICH will not update the vertical pointer field).

The far right QH is an example of a frame ‘termination’ node. Since its horizontal link pointer has its termination bit set, the Intel® 6300ESB ICH assumes there is no more work to complete for the current Frame.

Figure 19. Example Queue Conditions



Transfer Queues are based on the following characteristics:

- A QH’s vertical link pointer (Queue Element Link Pointer) references the ‘Top’ queue member. A QH’s horizontal link pointer (Queue Head Link Pointer) references the “next” work element in the Frame.
- Each queue member’s link pointer references the next element within the queue.

In the simplest model, the Intel® 6300ESB ICH follows vertical link point to a queue element, then executes the element. When the completion status of the TD satisfies the advance criteria as shown in Table 95, the Intel® 6300ESB ICH advances the queue

by writing the just-executed TD's link pointer back into the QH's Queue Element link pointer. The next time the queue head is traversed, the next queue element will be the Top element.

The traversal has two options: Breadth first, or Depth first. A flag bit in each TD (Vf - Vertical Traversal Flag) controls whether traversal is Breadth or Depth first. The default mode of traversal is Breadth-First. For Breadth-First, the Intel® 6300ESB ICH only executes the top element from each queue. The execution path is shown below:

1. QH (Queue Element Link Pointer)
2. TD
3. Write-Back to QH (Queue Element Link Pointer)
4. QH (Queue Head Link pointer).

Breadth-First is also performed for every transaction execution that fails the advance criteria. This means that when a queued TD fails, the queue does not advance and the Intel® 6300ESB ICH traverses the QH's Queue Head Link Pointer.

In a Depth-first traversal, the top queue element must complete successfully to satisfy the *advance criteria* for the queue. When the Intel® 6300ESB ICH is currently processing a queue, and the advance criteria are met, and the Vf bit is set, the Intel® 6300ESB ICH follows the TD's link pointer to the next schedule work item.

Note that regardless of traversal model, when the advance criteria are met, the successful TD's link pointer is written back to the QH's Queue Element link pointer.

When the Intel® 6300ESB ICH encounters a QH, it caches the QH internally, and sets internal state to indicate it is in a Q-context. It needs this state to update the correct QH (for auto advancement) and also to make the correct decisions on how to traverse the Frame List.

Restricting the advancement of queues to advancement criteria implements an ensured data delivery stream.

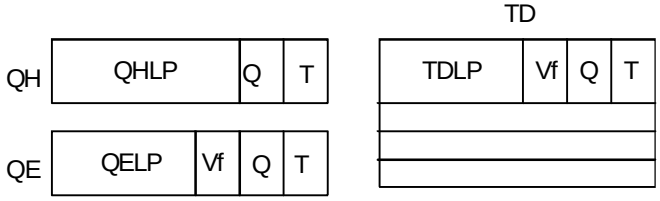
A queue is **never** advanced on an error completion status (even in the event the error count was exhausted).

Table 95 lists the general queue advance criteria, which are based on the execution status of the TD at the "Top" of a currently "active" queue.

Table 95. Queue Advance Criteria

Function-to-Host (IN)			Host-to-Function (OUT)		
Non-NULL	**NULL**	**Error/NAK**	**Non-NULL**	**NULL**	**Error/NAK**
Advance Q	Advance Q	Retry Q Element	Advance Q	Advance Q	Retry Q Element

Table 96 is a decision table illustrating the valid combinations of link pointer bits and the valid actions taken when advancement criteria for a queued transfer descriptor are met. The column headings for the link pointer fields are encoded, based on the following list:



Legend:

QH.LP = Queue Head Link Pointer (or Horizontal Link Pointer)
QE.LP = Queue Element Link Pointer (or Vertical Link Pointer)
TD.LP = TD Link Pointer
QH.Q = Q bit in QH
QH.T = T bit in QH
QE.Q = Q bit in QE
QE.T = T bit in QE
TD. Vf = Vf bit in TD
TD.Q = Q bit in TD
TD. T = T bit in TD

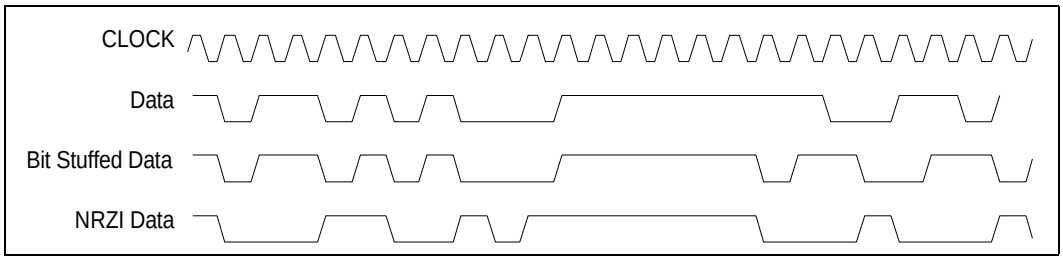
Table 96. USB Schedule List Traversal Decision Table

Q Context	QH.Q	QH.T	QE.Q	QE.T	TD.Vf	TD.Q	TD.T	Description
0	-	-	-	-	x	0	0	Not in Queue - execute TD. Use TD.LP to get next TD
0	-	-	-	-	x	x	1	Not in Queue - execute TD. End of Frame
0	-	-	-	-	x	1	0	Not in Queue - execute TD. Use TD.LP to get next (QH+QE). Set Q Context to 1.
1	0	0	0	0	0	x	x	In Queue. Use QE.LP to get TD. Execute TD. Update QE.LP with TD.LP. Use QH.LP to get next TD.
1	x	x	0	0	1	0	0	In Queue. Use QE.LP to get TD. Execute TD. Update QE.LP with TD.LP. Use TD.LP to get next TD.
1	x	x	0	0	1	1	0	In Queue. Use QE.LP to get TD. Execute TD. Update QE.LP with TD.LP. Use TD.LP to get next (QH+QE).
1	0	0	x	1	x	x	x	In Queue. Empty queue. Use QH.LP to get next TD
1	x	x	1	0	-	-	-	In Queue. Use QE.LP to get (QH+QE)
1	x	1	0	0	0	x	x	In Queue. Use QE.LP to get TD. Execute TD. Update QE.LP with TD.LP. End of Frame
1	x	1	x	1	x	x	x	In Queue. Empty queue. End of Frame
1	1	0	0	0	0	x	x	In Queue. Use QE.LP to get TD. Execute TD. Update QE.LP with TD.LP. Use QH.LP to get next (QH+QE).
1	1	0	x	1	x	x	x	In Queue. Empty queue. Use QH.LP to get next (QH+QE)

5.17.4 Data Encoding and Bit Stuffing

The USB employs NRZI data encoding (Non-Return to Zero Inverted) when transmitting packets. In NRZI encoding, a 1 is represented by no change in level and a 0 is represented by a change in level. A string of zeros causes the NRZI data to toggle each bit time. A string of ones causes long periods with no transitions in the data. In order to ensure adequate signal transitions, bit stuffing is employed by the transmitting device when sending a packet on the USB. A zero is inserted after every six consecutive ones in the data stream before the data is NRZI encoded to force a transition in the NRZI data stream. This gives the receiver logic a data transition at least once every seven bit times to ensure the data and clock lock. A waveform of the data encoding is shown in Figure 20.

Figure 20. USB Data Encoding



Bit stuffing is enabled beginning with the Sync Pattern and throughout the entire transmission. The data “one” that ends the Sync Pattern is counted as the first one in a sequence. Bit stuffing is always enforced, without exception. When required by the bit stuffing rules, a zero bit will be inserted even when it is the last bit before the end-of-packet (EOP) signal.

5.17.5 Bus Protocol

5.17.5.1 Bit Ordering

Bits are sent out onto the bus least significant bit (LSb) first, followed by next LSb, through to the most significant bit (MSb) last.

5.17.5.2 SYNC Field

All packets begin with a synchronization (SYNC) field, which is a coded sequence that generates a maximum edge transition density. The SYNC field appears on the bus as IDLE followed by the binary string “KJKJKJKK,” in its NRZI encoding. It is used by the input circuitry to align incoming data with the local clock and is defined to be eight bits in length. SYNC serves only as a synchronization mechanism and is not shown in the following packet diagrams. The last two bits in the SYNC field are a marker that is used to identify the first bit of the PID. All subsequent bits in the packet must be indexed from this point.

5.17.5.3 Packet Field Formats

Field formats for the token, data, and handshake packets are described in the following section. The effects of NRZI coding and bit stuffing have been removed for the sake of clarity. All packets have distinct start and end of packet delimiters.

Table 97. PID Format

Bit	Data Sent		Bit	Data Sent
0	PID 0		4	NOT(PID 0)
1	PID 1		5	NOT(PID 1)
2	PID 2		6	NOT(PID 2)
3	PID 3		7	NOT(PID 3)

Packet Identifier Field

A packet identifier (PID) immediately follows the SYNC field of every USB packet. A PID consists of a four bit packet type field followed by a four-bit check field as shown in Table 97. The PID indicates the type of packet and, by inference, the format of the packet and the type of error detection applied to the packet. The four-bit check field of the PID insures reliable decoding of the PID so that the remainder of the packet is interpreted correctly. The PID check field is generated by performing a ones complement of the packet type field.

Any PID received with a failed check field or which decodes to a non-defined value is assumed to be corrupted and the remainder of the packet is assumed to be corrupted and is ignored by the receiver. PID types, codes, and descriptions are listed in Table 98.

Table 98. PID Types

PID Type	PID Name	PID[3:0]	Description
Token	OUT	b0001	Address + endpoint number in host -> function transaction
	IN	b1001	Address + endpoint number in function -> host transaction
	SOF	b0101	Start of frame marker and frame number
	SETUP	b1101	Address + endpoint number in host -> function transaction for setup to a control endpoint
Data	DATA0	b0011	Data packet PID even
	DATA1	b1011	Data packet PID odd
Handshake	ACK	b0010	Receiver accepts error free data packet
	NAK	b1010	Rx device cannot accept data or Tx device cannot send data
	STALL	b1110	Endpoint is stalled
Special	PRE	b1100	Host-issued preamble. Enables downstream bus traffic to low speed devices.

PIDs are divided into four coding groups: token, data, handshake, and special, with the first two transmitted PID bits (PID[1:0]) indicating which group. This accounts for the distribution of PID codes.

5.17.5.4 Address Fields

Function endpoints are addressed using two fields: the function address field and endpoint field.

Table 99. Address Field

Bit	Data Sent		Bit	Data Sent
0	ADDR 0		4	ADDR 4
1	ADDR 1		5	ADDR 5
2	ADDR 2		6	ADDR 6
3	ADDR 3			

Address Field

The function address (ADDR) field specifies the function, through its address, that is either the source or destination of a data packet, depending on the value of the token PID. As shown in Table 99, a total of 128 addresses are specified as ADDR[6:0]. The ADDR field is specified for IN, SETUP, and OUT tokens.

Endpoint Field

An additional four-bit endpoint (ENDP) field, shown in Table 100, permits more flexible addressing of functions in which more than one sub-channel is required. Endpoint numbers are function specific. The endpoint field is defined for IN, SETUP, and OUT token PIDs only.

Table 100. Endpoint Field

Bit	Data Sent
0	ENDP 0
1	ENDP 1
2	ENDP 2
3	ENDP 3

5.17.5.5 Frame Number Field

The frame number field is an 11-bit field that is incremented by the host on a per frame basis. The frame number field rolls over upon reaching its maximum value of x7FF, and is sent only for SOF tokens at the start of each frame.

5.17.5.6 Data Field

The data field may range from zero to 1023 bytes and must be an integral numbers of bytes. Data bits within each byte are shifted out LSB first.

5.17.5.7 Cyclic Redundancy Check (CRC)

CRC is used to protect the all non-PID fields in token and data packets. In this context, these fields are considered to be protected fields. The PID is not included in the CRC check of a packet containing CRC. All CRCs are generated over their respective fields in the transmitter before bit stuffing is performed. Similarly, CRCs are decoded in the receiver after stuffed bits have been removed. Token and data packet CRCs provide 100% coverage for all single and double bit errors. A failed CRC is considered to indicate that one or more of the protected fields is corrupted and causes the receiver to ignore those fields, and, in most cases, the entire packet.

5.17.6 Packet Formats

5.17.6.1 Token Packets

Table 101 shows the field formats for a token packet. A token consists of a PID, specifying either IN, OUT, or SETUP packet type, and ADDR and ENDP fields. For OUT and SETUP transactions, the address and endpoint fields uniquely identify the endpoint that will receive the subsequent data packet. For IN transactions, these fields uniquely identify which endpoint should transmit a data packet. Only the Intel® 6300ESB ICH may issue token packets. IN PIDs define a data transaction from a function to the Intel® 6300ESB ICH. OUT and SETUP PIDs define data transactions from the Intel® 6300ESB ICH to a function.

Token packets have a five-bit CRC which covers the address and endpoint fields as shown above. The CRC does not cover the PID, which has its own check field. Token and SOF packets are delimited by an EOP after three bytes of packet field data. When a packet decodes as an otherwise valid token or SOF but does not terminate with an EOP after three bytes, it must be considered invalid and ignored by the receiver.

Table 101. Token Format

Packet	Width
PID	8 bits
ADDR	7 bits
ENDP	4 bits
CRC5	5 bits

5.17.6.2 Start of Frame Packets

Table 102 shows a start of frame (SOF) packet. SOF packets are issued by the host at a nominal rate of once every 1.00 ms +/- 0.05. SOF packets consist of a PID indicating packet type followed by an 11-bit frame number field.

The SOF token comprises the token-only transaction that distributes a start of frame marker and accompanying frame number at precisely timed intervals corresponding to the start of each frame. All full speed functions, including hubs, must receive and decode the SOF packet. The SOF token does not cause any receiving function to generate a return packet; therefore, SOF delivery to any given function cannot be ensured. The SOF packet delivers two pieces of timing information. A function is informed that a start of frame has occurred when it detects the SOF PID. Frame timing sensitive functions, which do not need to keep track of frame number, need only decode the SOF PID; they may ignore the frame number and its CRC. When a function needs to track frame number, it must comprehend both the PID and the time stamp.

Table 102. SOF Packet

Packet	Width
PID	8 bits
Frame Number	11 bits
CRC5	5 bits

5.17.6.3 Data Packets

A data packet consists of a PID, a data field, and a CRC as shown in Table 103. There are two types of data packets, identified by differing PIDs: DATA0 and DATA1. Two data packet PIDs are defined to support data toggle synchronization.

Data must always be sent in integral numbers of bytes. The data CRC is computed over only the data field in the packet and does not include the PID, which has its own check field.

Table 103. Data Packet Format

Packet	Width
PID	8 bits
DATA	0-1023 bytes
CRC16	16 bits

5.17.6.4 Handshake Packets

Handshake packets consist of only a PID. Handshake packets are used to report the status of a data transaction and may return values indicating successful reception of data, flow control, and stall conditions. Only transaction types that support flow control may return handshakes. Handshakes are always returned in the handshake phase of a transaction and may be returned, instead of data, in the data phase. Handshake packets are delimited by an EOP after one byte of packet field. When a packet is decoded as an otherwise valid handshake but does not terminate with an EOP after one byte, it must be considered invalid and ignored by the receiver.

There are three types of handshake packets:

- **ACK** indicates that the data packet was received without bit stuff or CRC errors over the data field and that the data PID was received correctly. An ACK handshake is applicable only in transactions in which data has been transmitted and where a handshake is expected. ACK may be returned by the host for IN transactions and by a function for OUT transactions.
- **NAK** indicates that a function was unable to accept data from the host (OUT) or that a function has no data to transmit to the host (IN). NAK may only be returned by functions in the data phase of IN transactions or the handshake phase of OUT transactions. The host may not issue a NAK. NAK is used for flow control purposes to indicate that a function is temporarily unable to transmit or receive data, but will eventually be able to do so without need of host intervention. NAK is also used by interrupt endpoints to indicate that no interrupt is pending.
- **STALL** is returned by a function in response to an IN token or after the data phase of an OUT. STALL indicates that a function is unable to transmit or receive data, and that the condition requires host intervention to remove the stall. Once a function's endpoint is stalled, the function must continue returning STALL until the condition causing the stall has been cleared through host intervention. The host is not permitted to return a STALL under any condition.

5.17.6.5 Handshake Responses

IN Transaction

A function may respond to an IN transaction with a STALL or NAK. When the token received was corrupted, the function will issue no response. When the function may transmit data, it will issue the data packet. The Intel® 6300ESB ICH, as the USB host, may return only one type of handshake on an IN transaction, an ACK. When it receives a corrupted data, or cannot accept data due to a condition such as an internal buffer overrun, it discards the data and issues no response.

OUT Transaction

A function may respond to an OUT transaction with a STALL, ACK, or NAK. When the transaction contained corrupted data, it will issue no response.

SETUP Transaction

Setup defines a special type of host to function data transaction which permits the host to initialize an endpoint's synchronization bits to those of the host. Upon receiving a Setup transaction, a function must accept the data. Setup transactions cannot be STALLed or NAKed and the receiving function must accept the Setup transfer's data. When a non-control endpoint receives a SETUP PID, it must ignore the transaction and return no response.

5.17.7 USB Interrupts

5.17.7.1 Overview

There are two general groups of USB interrupt sources, those resulting from execution of transactions in the schedule, and those resulting from an Intel® 6300ESB ICH operation error. All transaction-based sources may be masked by software through the Intel® 6300ESB ICH's Interrupt Enable register. Additionally, individual transfer descriptors may be marked to generate an interrupt on completion.

When the Intel® 6300ESB ICH drives an interrupt for USB, it internally drives the PIRQ[A]# pin for USB function #0, PIRQ[D]# pin for USB function #1 until all sources of the interrupt are cleared. In order to accommodate some operating systems, the Interrupt Pin register must contain a different value for each function of this new multi-function device.

5.17.7.2 Transaction Based Interrupts

These interrupts are not signaled until after the status for the last complete transaction in the frame has been written back to host memory. This ensures that software may safely process through (Frame List Current Index -1) when it is servicing an interrupt.

CRC Error / Time-Out

A CRC/Time-Out error occurs when a packet transmitted from the Intel® 6300ESB ICH to a USB device or a packet transmitted from a USB device to the Intel® 6300ESB ICH generates a CRC error. The Intel® 6300ESB ICH is informed of this event by a time-out from the USB device or by the Intel® 6300ESB ICH's CRC checker generating an error on reception of the packet. Additionally, a USB bus time-out occurs when USB devices do not respond to a transaction phase within 19-bit times of an EOP. Either of these conditions will cause the C_ERR field of the TD to decrement.

When the C_ERR field decrements to zero, the following occurs:

- The Active bit in the TD is cleared
- The Stalled bit in the TD is set
- The CRC/Time-out bit in the TD is set.
- At the end of the frame, the USB Error Interrupt bit is set in the HC status register.

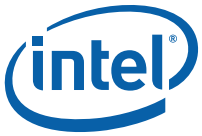
When the CRC/Time out interrupt is enabled in the Interrupt Enable register, a hardware interrupt will be signaled to the system.

Interrupt on Completion

Transfer Descriptors contain a bit that may be set to cause an interrupt on their completion. The completion of the transaction associated with that block causes the USB Interrupt bit in the HC Status Register to be set at the end of the frame in which the transfer completed. When a TD is encountered with the IOC bit set to 1, the IOC bit in the HC Status register is set to 1 at the end of the frame when the active bit in the TD is set to 0 (even when it was set to zero when initially read).

When the IOC Enable bit of Interrupt Enable register (bit 2 of I/O offset 04h) is set, a hardware interrupt is signaled to the system. The USB Interrupt bit in the HC status register is set either when the TD completes successfully or because of errors. When the completion is because of errors, the USB Error bit in the HC status register is also set.

Short Packet Detect



A transfer set is a collection of data which requires more than one USB transaction to completely move the data across the USB interface. An example might be a large print file which requires numerous TDs in multiple frames to completely transfer the data. Reception of a data packet that is less than the endpoint's Max Packet size during Control, Bulk or Interrupt transfers signals the completion of the transfer set, even when there are active TDs remaining for this transfer set. Setting the SPD bit in a TD indicates to the HC to set the USB Interrupt bit in the HC status register at the end of the frame in which this event occurs. This feature streamlines the processing of input on these transfer types. When the Short Packet Interrupt Enable bit in the Interrupt Enable register is set, a hardware interrupt is signaled to the system at the end of the frame where the event occurred.

Serial Bus Babble

When a device transmits on the USB for a time greater than its assigned Max Length, it is said to be babbling. Since isochrony may be destroyed by a babbling device, this error results in the Active bit in the TD being cleared to 0 and the Stalled and Babble bits being set to one. The C_ERR field is not decremented for a babble. The USB Error Interrupt bit in the HC Status register is set to 1 at the end of the frame. A hardware interrupt is signaled to the system.

When an EOF babble was caused by the Intel® 6300ESB ICH (due to incorrect schedule for instance), the Intel® 6300ESB ICH will force a bit stuff error followed by an EOP and the start of the next frame.

Stalled

This event indicates that a device/endpoint returned a STALL handshake during a transaction or that the transaction ended in an error condition. The TDs Stalled bit is set and the Active bit is cleared. Reception of a STALL does not decrement the error counter. A hardware interrupt is signaled to the system.

Data Buffer Error

This event indicates that an overrun of incoming data or a under-run of outgoing data has occurred for this transaction. This would generally be caused by the Intel® 6300ESB ICH not being able to access required data buffers in memory within necessary latency requirements. Either of these conditions will cause the C_ERR field of the TD to be decremented.

When C_ERR decrements to zero, the Active bit in the TD is cleared, the Stalled bit is set, the USB Error Interrupt bit in the HC Status register is set to 1 at the end of the frame and a hardware interrupt is signaled to the system.

Bit Stuff Error

A bit stuff error results from the detection of a sequence of more that 6 ones in a row within the incoming data stream. This will cause the C_ERR field of the TD to be decremented. When the C_ERR field decrements to 0, the Active bit in the TD is cleared to 0, the Stalled bit is set to one, the USB Error Interrupt bit in the HC Status register is set to 1 at the end of the frame and a hardware interrupt is signaled to the system.

5.17.7.3 Non-Transaction Based Interrupts

When an Intel® 6300ESB ICH process error or system error occurs, the Intel® 6300ESB ICH halts and immediately issues a hardware interrupt to the system.

Resume Received

This event indicates that the Intel® 6300ESB ICH received a RESUME signal from a device on the USB bus during a global suspend. When this interrupt is enabled in the Interrupt Enable register, a hardware interrupt will be signaled to the system allowing the USB to be brought out of the suspend state and returned to normal operation.

Intel® 6300ESB ICH Process Error

The HC monitors certain critical fields during operation to ensure that it does not process corrupted data structures. These include checking for a valid PID and verifying that the MaxLength field is less than 1280. When it detects a condition that would indicate that it is processing corrupted data structures, it immediately halts processing, sets the HC Process Error bit in the HC Status register and signals a hardware interrupt to the system.

This interrupt cannot be disabled through the Interrupt Enable register.

Host System Error

The Intel® 6300ESB ICH sets this bit to 1 when a PCI Parity error, PCI Master Abort, or PCI Target Abort occur. When this error occurs, the Intel® 6300ESB ICH clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs. This interrupt cannot be disabled through the Interrupt Enable register.

5.17.8 USB Power Management

The Host Controller may be put into a suspended state and its power may be removed. This requires that certain bits of information are retained in the resume power plane of the Intel® 6300ESB ICH so that a device on a port may wake the system. Such a device may be a fax-modem, which will wake up the machine to receive a fax or take a voice message. The settings of the following bits in I/O space will be maintained when the Intel® 6300ESB ICH enters the S3, S4 or S5 states:

Table 104. Bits Maintained in Low Power States

Register	Offset	Bit	Description
Command	00h	3	Enter Global Suspend Mode (EGSM)
Status	02h	2	Resume Detect
Port Status and Control	10h and 12h	2	Port Enabled/Disabled
		6	Resume Detect
		8	Low Speed Device Attached
		12	Suspend

When the Intel® 6300ESB ICH detects a resume event on any of its ports, it will set the corresponding USB_STS bit in ACPI space. When USB is enabled as a wake/break event, the system will wake up and an SCI will be generated.

5.17.9 USB Legacy Keyboard Operation

Typically when a USB keyboard is plugged into the system, and a standard keyboard is not, the system may not boot, and DOS legacy software will not run, because the keyboard will not be identified. In an Intel® 6300ESB ICH system Port 60/64 emulation will allow the USB keyboard and DOS legacy software to run. Port 60/64 emulation registers may be enabled by BIOS typically in a pre-OS environment and may be disabled during run time.

The Intel® 6300ESB ICH implements a series of trapping operations that will snoop accesses that typically go to the keyboard controller and put the expected data from the USB keyboard into ports 60/64.

The following table summarizes the implementation of the bits in the USB Legacy Keyboard/Mouse Control Registers.

Table 105. USB Legacy Keyboard/Mouse Control Register Bit Implementation (Sheet 1 of 2)

Bit #	Bit Name	Summary	Details
15	SMI Caused by End of Pass-Through	Logically 1 bit for all controllers	Note this bit in all host controllers will be set at the same time and cleared at the same time. It is cleared whenever software writes a one to this bit in any of the three host controllers. This bit may either be implemented separately for each controller or shared and aliased.
13	PCI Interrupt Enable	Independent enable	Each bit provides individual host control.
12	SMI Caused by USB Interrupt	Independent status	Individual status bits for each controller.
11	SMI Caused by Port 64 Write	Logically 1 bit for all controllers	Note this bit in all host controllers will be set at the same time and cleared at the same time. It is cleared whenever software writes a one to this bit in any of the three host controllers. This bit may either be implemented separately for each controller or shared and aliased.
10	SMI Caused by Port 64 Read	Logically 1 bit for all controllers	Note this bit in all host controllers will be set at the same time and cleared at the same time. It is cleared whenever software writes a one to this bit in any of the three host controllers. This bit may either be implemented separately for each controller or shared and aliased.
9	SMI Caused by Port 60 Write	Logically 1 bit for all controllers	Note this bit in all host controllers will be set at the same time and cleared at the same time. It is cleared whenever software writes a one to this bit in any of the three host controllers. This bit may either be implemented separately for each controller or shared and aliased.
8	SMI Caused by Port 60 Read	Logically 1 bit for all controllers	Note this bit in all host controllers will be set at the same time and cleared at the same time. It is cleared whenever software writes a one to this bit in any of the three host controllers. This bit may either be implemented separately for each controller or shared and aliased.
7	SMI at End of Pass-Through Enable	Separate enables ORed together	This bit enables the generation of the SMI based on bit 15 within the same function. When bit 15 is implemented as a shared/aliased bit across all functions, the bit 7's from all three controllers are ORed together and used to enable the SMI based on bit 15.
6	Pass Through State	Logically 1 bit for all controllers	This bit in all host controllers reflects the state of the Pass-Through state machine. Software may force this bit to zero by clearing the A20Gate Pass-Through Enable (bit 5) in *all* of the host controllers.
5	A20Gate Pass-Through Enable	ORed together to enable the pass-through state machine	When any of these bits in the three host controllers is set, the Intel® 6300ESB ICH will enable the Legacy Keyboard A20Gate Pass-through sequence. This prevents the SMI status bits (11:8) from asserting in all three controllers when the specific sequence of I/O cycles is observed.

Table 105. USB Legacy Keyboard/Mouse Control Register Bit Implementation (Sheet 2 of 2)

Bit #	Bit Name	Summary	Details
4	SMI on USB IRQ	Independent Enable	Each bit provides individual host control.
3	SMI on Port 64 Writes Enable	Separate enables ORed together	Each bit enables SMI generation when the corresponding bit 11 is set. When bit 11 is implemented as a shared/aliased bit across all functions, then the bit 3's from all three controllers are ORed together and used to enable the SMI based on bit 11.
2	SMI on Port 64 Reads Enable	Separate enables OR'ed together	Each bit enables SMI generation if the corresponding bit 10 is set. If bit 10 is implemented as a shared/aliased bit across all functions, then the bit 2's from all three controllers are OR'ed together and used to enable the SMI based on bit 10.
1	SMI on Port 60 Writes Enable	Separate enables OR'ed together	Each bit enables SMI generation if the corresponding bit 9 is set. If bit 9 is implemented as a shared/aliased bit across all functions, then the bit 1's from all three controllers are OR'ed together and used to enable the SMI based on bit 9.
0	SMI on Port 60 Reads Enable	Separate enables OR'ed together	Each bit enables SMI generation if the corresponding bit 8 is set. If bit 8 is implemented as a shared/aliased bit across all functions, then the bit 0's from all three controllers are OR'ed together and used to enable the SMI based on bit 8.
NOTE: If bit 7 of the Extended Test Mode Register 1 (D31:F0, offset F4h ETR1, section 9.1.36) is set. Port 60/64h Reads and Writes from an external PCI agent will not affect set bits 8-11 and will not cause an SMI independent of the setting of bits 0-3			

Note: The scheme described below assumes that the keyboard controller (8042 or equivalent) is on the LPC bus.

This legacy operation is performed through SMM space.

Figure 21 shows the Enable and Status path. The latched SMI source (60R, 60W, 64R, 64W) is available in the Status Register. Since the enable is after the latch, it is possible to check for other events that didn't necessarily cause an SMI. It is the software's responsibility to logically AND the value with the appropriate enable bits.

Note also that the SMI is generated before the PCI cycle completes (e.g., before TRDY# goes active) to ensure that the processor doesn't complete the cycle before the SMI is observed. This method is used on MPIIX and has been validated.

The logic will also need to block the accesses to the 8042. When there is an external 8042, then this is simply accomplished by not activating the 8042 CS. This is simply done by logically ANDing the four enables (60R, 60W, 64R, 64W) with the four types of accesses to determine when 8042CS should go active. An additional term is required for the "Pass-through" case.

The state table for the diagram is shown in Table 106.

Figure 21. USB Legacy Keyboard Flow Diagram

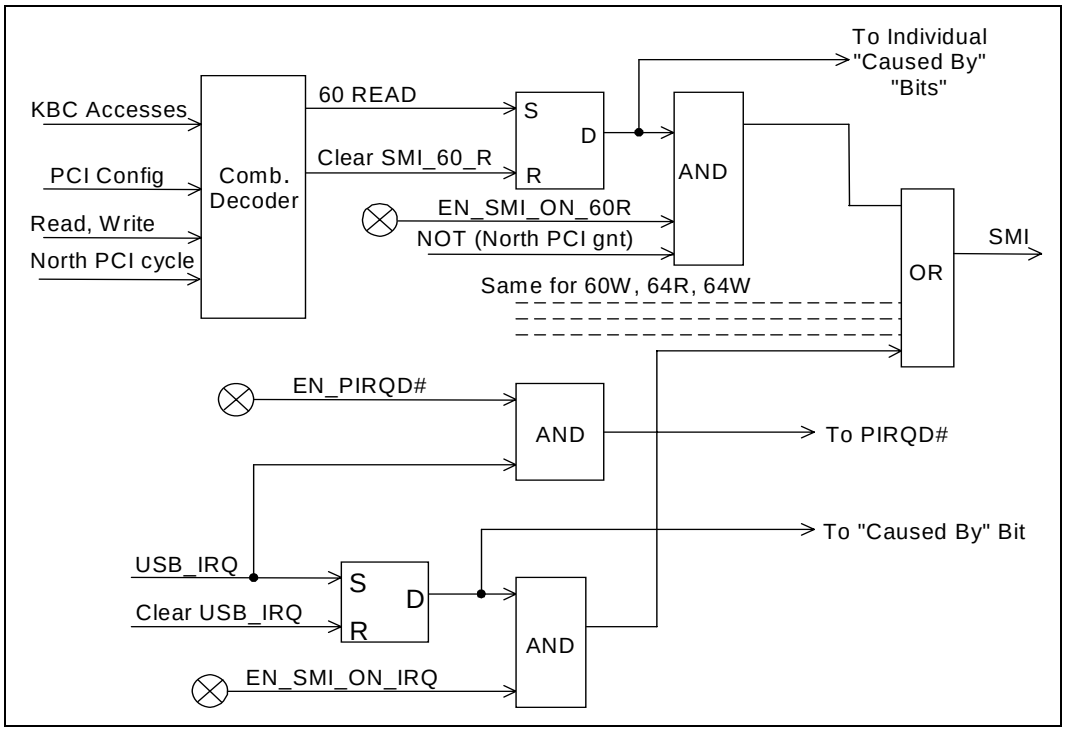


Table 106. USB Legacy Keyboard State Transitions

Current State	Action	Data Value	Next State	Comment
IDLE	64h / Write	D1h	GateState 1	Standard D1 command. Cycle passed through to 8042. SMI# doesn't go active. PSTATE (offset C0, bit 6) goes to 1.
IDLE	64h / Write	Not D1h	IDLE	Bit 3 in Config Register determines if cycle passed through to 8042 and if SMI# generated.
IDLE	64h / Read	N/A	IDLE	Bit 2 in Config Register determines if cycle passed through to 8042 and if SMI# generated.
IDLE	60h / Write	Don't Care	IDLE	Bit 1 in Config Register determines if cycle passed through to 8042 and if SMI# generated.
IDLE	60h / Read	N/A	IDLE	Bit 0 in Config Register determines if cycle passed through to 8042 and if SMI# generated.
GateState1	60h / Write	XXh	GateState 2	Cycle passed through to 8042, even if trap enabled in Bit 1 in Config Register. No SMI# generated. PSTATE remains 1. When data value is not DFh or DDh then the 8042 may chose to ignore it.
GateState1	64h / Write	D1h	GateState 1	Cycle passed through to 8042, even if trap enabled through Bit 3 in Config Register. No SMI# generated. PSTATE remains 1. Stay in GateState1 because this is part of the double-trigger sequence.
GateState1	64h / Write	Not D1h	IDLE	Bit 3 in Config space determines if cycle passed through to 8042 and if SMI# generated. PSTATE goes to 0. When Bit 7 in Config Register is set, then SMI# should be generated.
GateState1	60h / Read	N/A	IDLE	This is an invalid sequence. Bit 0 in Config Register determines if cycle passed through to 8042 and if SMI# generated. PSTATE goes to 0. When Bit 7 in Config Register is set, then SMI# should be generated.
GateState1	64h / Read	N/A	GateState 1	Just stay in same state. Generate an SMI# when enabled in Bit 2 of Config Register. PSTATE remains 1.
GateState2	64 / Write	FFh	IDLE	Standard end of sequence. Cycle passed through to 8042. PSTATE goes to 0. Bit 7 in Config Space determines if SMI# should be generated.
GateState2	64h / Write	Not FFh	IDLE	Improper end of sequence. Bit 3 in Config Register determines if cycle passed through to 8042 and if SMI# generated. PSTATE goes to 0. When Bit 7 in Config Register is set, then SMI# should be generated.
GateState2	64h / Read	N/A	GateState 2	Just stay in same state. Generate an SMI# when enabled in Bit 2 of Config Register. PSTATE remains 1.

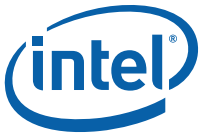


Table 106. USB Legacy Keyboard State Transitions

Current State	Action	Data Value	Next State	Comment
GateState2	60h / Write	XXh	IDLE	Improper end of sequence. Bit 1 in Config Register determines if cycle passed through to 8042 and if SMI# generated. PSTATE goes to 0. When Bit 7 in Config Register is set, then SMI# should be generated.
GateState2	60h / Read	N/A	IDLE	Improper end of sequence. Bit 0 in Config Register determines if cycle passed through to 8042 and if SMI# generated. PSTATE goes to 0. When Bit 7 in Config Register is set, then SMI# should be generated.

NOTES:
1. If bit 7 of the Extended Test Mode Register 1 (D31:F0, offset F4h ETR1, section 9.1.36) is set. Port 60/64h Reads and Writes from an external PCI agent will not affect set bits 8-11 and will not cause an SMI independent of the setting of bits 0-3
2. System Software should ensure that the host controller and an external PCI agent are not simultaneously executing keyboard accesses including an A20Gate Pass-through sequence to Port 60h & 64h. This is not supported and the results may be indeterminate.

5.18 USB EHCI Controller (D29:F7)

5.18.1 Overview

The Intel® 6300ESB ICH contains an Enhanced Host Controller Interface (EHCI) compliant host controller which supports up to four High-speed USB 2.0 Specification compliant root ports. High-speed USB 2.0 allows data transfers up to 480 Mbps using the same pins as the four Full-speed and Low-speed USB Universal Host Controller Interface (UHCI) ports. The Intel® 6300ESB ICH contains port-routing logic that determines whether a USB port is controlled by one of the UHCI controllers or by the EHCI controller. A USB 2.0 based Debug Port is also implemented in the Intel® 6300ESB ICH.

A summary of the key architectural differences between the USB UHCI host controllers and the USB EHCI host controller are shown in the table below:

Table 107. UHCI vs. EHCI

Feature	USB 1.1 UHCI	USB 2.0 EHCI
Accessible by	I/O space	Memory Space
Memory Data Structure	Single linked list	Separated into Periodic and Asynchronous lists.
Differential Signaling Voltage	3.3 V	400 mV
Ports per Controller	2	4

5.18.2 EHC Initialization

The following descriptions step through the expected Intel® 6300ESB ICH Enhanced Host Controller (EHC) initialization sequence in chronological order, beginning with a complete power cycle in which the suspend well and core well have been off.

5.18.2.1 Power On

The suspend well is a “deeper” power plane than the core well, which means that the suspend well is always functional when the core well is functional but the core well may not be functional when the suspend well is. Therefore, the suspend well reset pin (RSMRST#) deasserts before the core well reset pin (PWROK) rises.

1. The suspend well reset deasserts, leaving all registers and logic in the suspend well in the default state. However, it is not possible to read any registers until after the core well reset deasserts. Note that normally the suspend well reset will only occur when a desktop system is unplugged or the battery is removed from a mobile system. In other words, suspend well resets are not easily achieved by software or the end-user. This step will typically not occur immediately before the remaining steps.
2. The core well reset deasserts, leaving all registers and logic in the core well in the default state. The EHC configuration space is accessible at this point. Note that the core well reset may (and typically does) occur without the suspend well reset asserting. This means that all of the Configure Flag and Port Status and Control bits (and any other suspend-well logic) may be in any valid state at this time.

5.18.2.2 BIOS Initialization

BIOS performs a number of platform customization steps after the core well has powered up as described in the Intel® 6300ESB ICH *BIOS Specification*.

5.18.2.3 Driver Initialization

See Chapter 4 of the *Enhanced Host Controller Interface Specification for Universal Serial Bus, Revision 0.96*.

5.18.2.4 EHC Resets

In addition to the standard Intel® 6300ESB ICH hardware resets, portions of the EHC are reset by the HCRESET bit and the transition from the D3hot device power management state to the D0 state. The effects of each of these resets are:

Table 108. EHC Resets

Reset	Does Reset	Does not Reset	Comments
HCRESET bit set	Memory space registers except Structural Parameters (which is written by BIOS).	Configuration registers	The HCRESET must only affect registers that the EHCI driver controls. PCI Configuration space and BIOS-programmed parameters cannot be reset. See Section 11.2.2.1, "Offset CAPLENGTH + 00 - 03h: USB EHCI CMD—USB EHCI Command Register" for information regarding offset 00h, bit 1.
Software writes the Device Power State from D3 hot (11b) to D0 (00b).	Core well registers (except BIOS-programmed registers).	Suspend well registers; BIOS-programmed core well registers	The D3-to-D0 transition must not cause wake information (suspend well) to be lost. It also must not clear BIOS-programmed registers because BIOS may not be invoked following the D3-to-D0 transition. See Section 11.1.17, "Offset 54 - 55h: Power Management Control/Status" for more information.

When the detailed register descriptions give exceptions to these rules, those exceptions override these rules. This summary is provided to help explain the reasons for the reset policies.

5.18.3 Data Structures in Main Memory

See Section 3 and Appendix B of the *Enhanced Host Controller Interface Specification for Universal Serial Bus,* Revision 0.96.

5.18.4 USB 2.0 Enhanced Host Controller DMA

The Intel® 6300ESB ICH USB 2.0 EHC implements three sources of USB packets. They are, in order of priority on USB during each microframe:

1. USB 2.0 Debug Port (see Section 5.18.11, “USB 2.0 EHCI Based Debug Port”)
2. Periodic DMA engine
3. Asynchronous DMA engine

The Intel® 6300ESB ICH always performs any currently-pending debug port transaction at the beginning of a microframe, followed by any pending periodic traffic for the current microframe. When there is time left in the microframe, then the EHC performs any pending asynchronous traffic until the end of the microframe (EOF1). Note that the debug port traffic is only presented on one port (Port #0), while the other ports are idle during this time.

The following subsections describe the policies of the periodic and asynchronous DMA engines.

5.18.4.1 Periodic List Execution

The Periodic DMA engine contains buffering for two control structures (two transactions). By implementing two entries, the EHC is able to pipeline the memory accesses for the next transaction while executing the current transaction on the USB ports. Note that a multiple-packet, High-Bandwidth transaction occupies one of these buffer entries, which means that up to six 1-Kbyte data packets may be associated with the two buffered control structures.

5.18.4.1.1 Read Policies for Periodic DMA

The Periodic DMA engine performs reads for the following structures.

Table 109. Read Policies for Periodic DMA

Memory Structure	Size (DWORDs)	Comments
Periodic Frame List entry	1	The EHC reads the entry for each microframe. The frame list is not internally cached across microframes.
iTD	23	Only the 64-bit addressing format is supported.
siTD	9	Only the 64-bit addressing format is supported.
qTD	13	Only the 64-bit addressing format is supported.
Queue Head	17	Only the 64-bit addressing format is supported.
Out Data	Up to 257	The Intel® 6300ESB ICH breaks large read requests down into smaller aligned read requests based on the setting of the Read Request Max Length field.
Frame Span Transversal Node	2	

The EHC Periodic DMA Engine (PDE) does not generate accesses to main memory unless all three of the following conditions are met.

- The HCHalted bit is 0 (memory space, offset 04h, bit 12). Software clears this bit indirectly by setting the RUN/STOP bit to 1. See Section 11.2.2.2, "Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status" for more information.
- The Periodic Schedule Status bit is 1 (memory space, offset 04h, bit 14). Software sets this bit indirectly by setting the Periodic Schedule Enable Bit to 1. See Section 11.2.2.2, "Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status" for more information.
- The Bus Master Enable bit is 1 (configuration space, offset 04h, bit 2). See Section 11.1.1, "Offset 04 - 05h: Command Register" for more information.

Note: Prefetching is limited to the current and next microframes only.

Note: Once the PDE checks the length of a periodic packet against the remaining time in the microframe (late-start check) and decides that there is not enough time to run it on the wire, then the EHC switches over to run asynchronous traffic.

5.18.4.1.2 Write Policies for Periodic DMA

The Periodic DMA engine performs writes for the following reasons.

Table 110. Write Policies for Periodic DMA

Memory Structure	Size (DWORDs)	Comments
iTD Status Write	1	Only the DWORD that corresponds to the just-executed microframe's status is written. All bytes of the DWORD are written.
siTD Status Write	3	DWORDs 0C:17h are written. IOC and Buffer Pointer fields are re-written with the original value.
Interrupt Queue Head Overlay	14	Only the 64-bit addressing format is supported. DWORDs 0C:43h are written.
Interrupt Queue Head Status Write	5	DWORDs 14:27h are written.
Interrupt qTD Status Write	3	DWORDs 04:0Fh are written. PID Code, IOC, Buffer Pointers, and Alt. Next qTD Pointers are re-written with the original value.
In Data	Up to 257	The Intel® 6300ESB ICH breaks data writes down into 16 DWORD aligned chunks.

NOTES:
1. The Periodic DMA Engine (PDE) will only generate writes after a transaction is executed on USB.
2. Status writes are always performed after In Data writes for the same transaction.

5.18.4.2 Asynchronous List Execution

The Asynchronous DMA engine contains buffering for two control structures (two transactions). By implementing two entries, the EHC is able to pipeline the memory accesses for the next transaction while executing the current transaction on the USB ports.

5.18.4.2.1 Read Policies for Asynchronous DMA

The Asynchronous DMA engine performs reads for the following structures.

Table 111. Read Policies for Asynchronous DMA

Memory Structure	Size (DW)	Comments
qTD	13	Only the 64-bit addressing format is supported.
Queue Head	17	Only the 64-bit addressing format is supported.
Out Data	Up to 129	The Intel® 6300ESB ICH breaks large read requests down into smaller aligned read requests based on the setting of the Read Request Max Length field.

The EHC Asynchronous DMA Engine (ADE) does not generate accesses to main memory unless all four of the following conditions are met. (Note that the ADE may be active when the periodic schedule is actively executed, unlike the description in the EHCI specification; since the EHC contains independent DMA engines, the ADE may perform memory accesses interleaved with the PDE accesses.)

- The HCHalted bit is 0 (memory space, offset 04h, bit 12). Software clears this bit indirectly by setting the RUN/STOP bit to 1.
- The Asynchronous Schedule Status bit is 1 (memory space, offset 04h, bit 15). Software sets this bit indirectly by setting the Asynchronous Schedule Enable Bit to 1. See Section 11.2.2.2, “Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status” for more information.
- The Bus Master Enable bit is 1 (configuration space, offset 04h, bit 2). See Section 11.1.1, “Offset 04 - 05h: Command Register” for more information.
- The ADE is not sleeping due to the detection of an empty schedule. There is not one single bit that indicates this state. However, the sleeping state is entered when the Queue Head with the H bit set is encountered when the Reclamation bit in the USB EHCI Status register is 0. See Section 11.2.2.2, “Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status” for information regarding offset 04h, bit 13.

Note: The ADE does not fetch data when a QH is encountered in the Ping state. An Ack handshake in response to the Ping results in the ADE writing the QH to the Out state, which results in the fetching and delivery of the Out Data on the next iteration through the asynchronous list.

Note: Once the ADE checks the length of an asynchronous packet against the remaining time in the microframe (late-start check) and decides that there is not enough time to run it on the wire, then the EHC stops all activity on the USB ports for the remainder of that microframe.

Note: Once the ADE detects an “empty” asynchronous schedule as described in Section 4 of the EHCI specification, it implements a waking mechanism like the one in the example. The amount of time that the ADE “sleeps” is 10 µs ± 30 ns.

5.18.4.2.2 Write Policies for Asynchronous DMA

The Asynchronous DMA engine performs writes for the following reasons.

Table 112. Write Policies for Asynchronous DMA

Memory Structure	Size (DWORDs)	Comments
Asynchronous Queue Head Overlay	14	Only the 64-bit addressing format is supported. DWORDs 0C:43h are written.
Asynchronous Queue Head Status Write	34	DWORDs 14:1Fh are written.
Asynchronous qTD Status Write	3	DWORDs 04:0Fh are written. PID Code, IOC, Buffer Pointer (Page 0), and Alt. Next qTD Pointers are re-written with the original value.
In Data	Up to 1297	The Intel® 6300ESB ICH breaks data writes down into 16 DWORD aligned chunks.

NOTES:
1. The Asynchronous DMA Engine (ADE) will only generate writes after a transaction is executed on USB.
2. Status writes are always performed after In Data writes for the same transaction.

5.18.5 Data Encoding and Bit Stuffing

See Chapter 8 of the *Universal Serial Bus Revision 2.0 Specification*.

5.18.6 Packet Formats

See Chapter 8 of the *Universal Serial Bus Revision 2.0 Specification*.

5.18.7 USB EHCI Interrupts and Error Conditions

Section 4 of the EHCI specification goes into detail on the EHC interrupts and the error conditions that cause them. All error conditions that the EHC detects may be reported through the EHCI Interrupt status bits. Only Intel® 6300ESB ICH-specific interrupt and error-reporting behavior is documented in this section. The EHCI Interrupts Section must be read first, followed by this section of the EDS to fully comprehend the EHC interrupt and error-reporting functionality.

- Based on the EHC's Buffer sizes and buffer management policies, the Data Buffer Error may not occur on the Intel® 6300ESB ICH.
- Master Abort and Target Abort responses from Hub Interface on EHC-initiated read packets will be treated as Fatal Host Errors. The EHC halts when these conditions are encountered.
- The Intel® 6300ESB ICH may assert the interrupts which are based on the interrupt threshold as soon as the status for the last complete transaction in the interrupt interval has been posted in the internal write buffers. The requirement in the EHCI Specification (that the status is written to memory) is met internally, even though the write may not be seen on the Hub Interface before the interrupt is asserted.
- Since the Intel® 6300ESB ICH supports the 1024-element Frame List size, the Frame List Rollover interrupt occurs every 1024 milliseconds.

- The Intel® 6300ESB ICH delivers interrupts using PIRQ#[H].
- The Intel® 6300ESB ICH does not modify the CERR count on an Interrupt IN when the “Do Complete-Split” execution criteria are not met.
- For complete-split transactions in the Periodic list, the “Missed Microframe” bit does not get set on a control-structure-fetch that fails the late-start test. When subsequent accesses to that control structure do not fail the late-start test, then the “Missed Microframe” bit will get set and written back.

5.18.7.1 Aborts on USB EHCI-Initiated Memory Reads

When a read initiated by the EHC is aborted, the EHC treats it as a fatal host error. The following actions are taken when this occurs:

- The Host System Error status bit is set. See Section 11.2.2.2, “Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status” for information regarding offset 04h, bit 4.
- The DMA engines are halted after completing up to one more transaction on the USB interface.
- When enabled (by the Host System Error Enable), an interrupt is generated. See Section 11.2.2.3, “Offset CAPLENGTH + 08 - 0Bh: USB EHCI INTR—USB EHCI Interrupt Enable” for information regarding offset 08h, bit 4.
- When the status is Master Abort, the Received Master Abort bit in configuration space is set. Section 11.1.2, “Offset 06 - 07h: Device Status” for information regarding offset 06h, bit 13.
- When the status is Target Abort, the Received Target Abort bit in configuration space is set. Section 11.1.2, “Offset 06 - 07h: Device Status” for information regarding offset 06h, bit 12.
- When enabled (by the SERR Enable bit in the function’s configuration space, see Section 11.1.1, “Offset 04 - 05h: Command Register”, offset 04h, bit 8), the Signaled System Error bit is set by the Intel® 6300ESB ICH when it signals SERR# (internally). Section 11.1.2, “Offset 06 - 07h: Device Status” for information regarding offset 06h,
bit 14.

5.18.8 USB EHCI Power Management

5.18.8.1 Pause Feature

This feature allows platforms, especially mobile systems, to dynamically enter low-power states during brief periods when the system is idle, i.e., between keystrokes. This is useful for enabling power management features like C3, C4, and Intel® SpeedStep® technology in the Intel® 6300ESB ICH. The policies for entering these states typically are based on the recent history of system bus activity to incrementally enter deeper power management states. Normally, when the EHC is enabled, it regularly accesses main memory while traversing the DMA schedules looking for work to do; this activity is viewed by the power management software as a non-idle system, thus preventing the power managed states to be entered. Suspending all of the enabled ports may prevent the memory accesses from occurring, but there is an inherent latency overhead with entering and exiting the suspended state on the USB ports that makes this unacceptable for the purpose of dynamic power management. As a result, the EHCI software drivers are allowed to pause the EHC’s DMA engines when it knows that the traffic patterns of the attached devices may afford the delay. The pause only prevents the EHC from generating memory accesses; the SOF packets continue to be generated on the USB ports (unlike the suspended state).

5.18.8.2 Suspend Feature

The EHCI Specification describes the details of Port Suspend and Resume in detail in Section 4.3.

5.18.8.3 ACPI Device States

The USB EHCI function only supports the D0 and D3 PCI Power Management states. Notes regarding the Intel® 6300ESB ICH implementation of the Device States:

1. The EHC hardware does not inherently consume any more power when it is in the D0 state than it does in the D3 state. However, software is required to suspend or disable all ports prior to entering the D3 state such that the maximum power consumption is reduced.
2. In the D0 state, all implemented EHC features are enabled.
3. In the D3 state, accesses to the EHC memory-mapped I/O range will master abort. Note that, since the Debug Port uses the same memory range, the Debug Port is only operational when the EHC is in the D0 state.
4. In the D3 state, the EHC interrupt must never assert for any reason. The internal PME# signal is used to signal wake events, etc.
5. When the Device Power State field is written to D0 from D3, an internal reset is generated. See section EHC Resets for general rules on the effects of this reset.
6. Attempts to write any other value into the Device Power State field other than 00b (D0 state) and 11b (D3 state) will complete normally without changing the current value in this field. See Section 11.1.17, “Offset 54 - 55h: Power Management Control/Status” for information regarding offset 54h, bits [1:0].

5.18.8.4 ACPI System States

The EHC behavior as it relates to other power management states in the system is summarized in the following list:

- The System is always in the S0 state when the EHC is in the D0 state. However, when the EHC is in the D3 state, the system may be in any power management state (including S0).
- When in D0, the Pause feature (See Section 5.18.8.1, “Pause Feature”) enables dynamic processor low-power states to be entered.
- All core well logic is reset in the S3/S4/S5 states (core power turns off).

5.18.8.5 Low-power system Considerations

The Intel® 6300ESB ICH USB EHCI implementation does not behave differently in low power configurations. However, some features may be especially useful for the low power configurations.

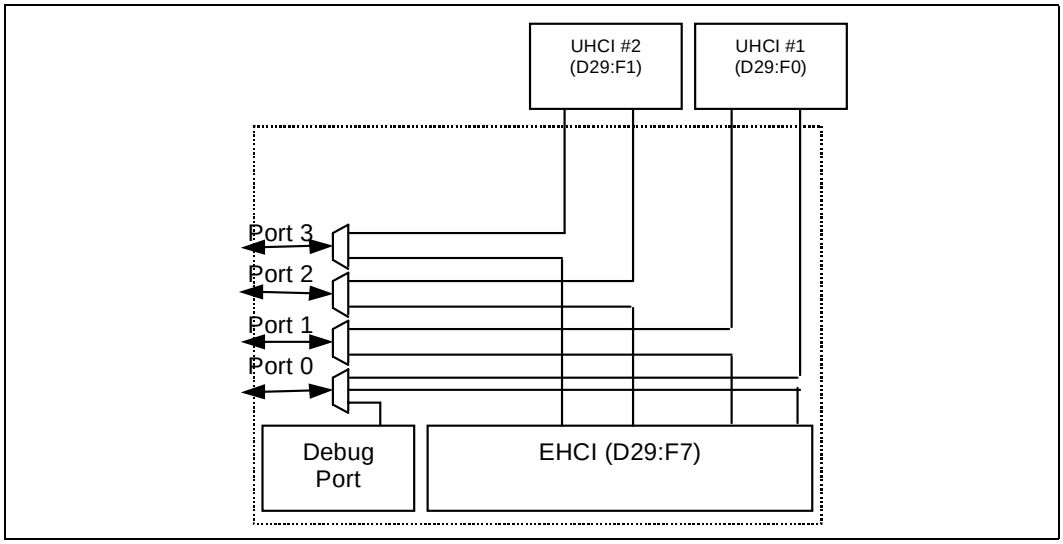
- Low-power systems are not likely to use all four of the USB ports that are provided on the Intel® 6300ESB ICH. With this in mind, the Intel® 6300ESB ICH provides mechanisms for changing the structural parameters of the EHC and hiding unused USB UHCI controllers. See Intel® 6300ESB ICH BIOS Specification on how BIOS should configure the Intel® 6300ESB ICH.
- Low-power systems may want to minimize the conditions that will wake the system. The Intel® 6300ESB ICH implements the “Wake Enable” bits in the Port Status and Control registers, as specified in the EHCI spec, for this purpose.
- Low-power systems may want to cut suspend well power to some or all USB ports when in a low-power state. The Intel® 6300ESB ICH implements the optional Port Wake Capability Register in the EHC Configuration Space for this platform-specific information to be communicated to software.

5.18.9 Interaction with Classic Host Controllers

The Enhanced Host Controller shares the four USB ports with two UHCI Host Controllers in the Intel® 6300ESB ICH. The USB UHCI Controller at D29:F0 shares ports 0 and 1 and the USB UHCI Controller at D29:F1 shares ports 2 and 3 with the EHCI Controller. There is very little interaction between the USB EHCI and the USB UHCI controllers other than the muxing control which is provided as part of the EHCI Controller.

Figure 22 depicts the USB Port Connections at a conceptual level. The dashed rectangle indicates all of the logic that is part of the Enhanced Host Controller cluster.

Figure 22. Intel® 6300ESB ICH-USB Port Connections



5.18.9.1 Port-Routing Logic

Integrated into the EHC functionality is port-routing logic, which performs the muxing between the USB UHCI and USB EHCI host controllers. The Intel® 6300ESB ICH conceptually implements this logic as described in Section 4.2 of the EHCI Specification. When a device is connected that is not capable of USB 2.0's High-Speed signaling protocol or when the EHCI software drivers are not present as indicated by the Configured Flag, the USB UHCI Controller owns the port. Owning the port means that the differential output is driven by the owner and the input stream is only visible to the owner. The host controller that is not the owner of the port internally sees a disconnected port.

Note that the port-routing logic is the only block of logic within the Intel® 6300ESB ICH that observes the physical (real) connect/disconnect information. The port status logic inside each of the host controllers observes the electrical connect/disconnect information that is generated by the port-routing logic.

Only the differential signal pairs are muxed/demuxed between the USB UHCI and USB EHCI host controllers. The other USB functional signals are handled as follows: The overcurrent inputs (OC#[3:0]) are directly routed to both controllers. An overcurrent event is recorded in both controllers' status registers.

The Port-Routing logic is implemented in the Suspend power well so that re-enumeration and remapping of the USB ports is not required following entering and exiting a system sleep state in which the core power is turned off.

The Intel® 6300ESB ICH also allows the USB Debug Port traffic to be routed in and out of Port #0. When in this mode, the Enhanced Host Controller is the owner of Port #0.

5.18.9.2 Device Connects

Section 4.2 of the EHCI Specification describes the details of handling Device Connects. There are four general scenarios that are summarized below. See Section 11.2.2.8, "Offset CAPLENGTH + 40 - 43h: CONFIGFLAG—Configure Flag Register".

1. Configure Flag = 0 and a USB Full-speed/Low-speed -only Device is connected

   In this case, the USB UHCI Controller is the owner of the port both before and after the connect occurs. The EHC (except for the port-routing logic) never sees the connect occur. The UHCI driver handles the connection and initialization process.

2. Configure Flag = 0 and an USB High-speed-capable Device is connected

   In this case, the USB UHCI Controller is the owner of the port both before and after the connect occurs. The EHC (except for the port-routing logic) never sees the connect occur. The UHCI driver handles the connection and initialization process. Since the USB UHCI Controller does not perform the high-speed chirp handshake, the device operates in compatible mode.

3. Configure Flag = 1 and a USB Full-speed/Low-speed-only Device is connected

   In this case, the USB EHCI Controller is the owner of the port before the connect occurs. The EHCI driver handles the connection and performs the port reset. After the reset process completes, the EHC hardware has cleared (not set) the Port Enable bit in the EHC's PORTSC register. The EHCI driver then writes a 1 to the Port Owner bit in the same register, causing the USB UHCI Controller to see a connect event and the EHC to see an "electrical" disconnect event. The UHCI driver and hardware handle the connection and initialization process from that point on. The EHCI driver and hardware handle the perceived disconnect.

4. Configure Flag = 1 and an USB High-speed-capable Device is connected

   In this case, the USB EHCI Controller is the owner of the port before, and remains the owner after, the connect occurs. The EHCI driver handles the connection and performs the port reset. After the reset process completes, the EHC hardware has set the Port Enable bit in the EHC's PORTSC register. The port is functional at this point. The USB UHCI Controller continues to see an unconnected port.

5.18.9.3 Device Disconnects

Section 4.2 of the EHCI Specification describes the details of handling Device Connects. There are three general scenarios that are summarized below. See Section 11.2.2.8, "Offset CAPLENGTH + 40 - 43h: CONFIGFLAG—Configure Flag Register".

1. Configure Flag = 0 and the device is disconnected.

   In this case, the USB UHCI Controller is the owner of the port both before and after the disconnect occurs. The EHC (except for the port-routing logic) never sees a device attached. The UHCI driver handles disconnection process.

2. Configure Flag = 1 and a USB Full-speed/Low-speed-capable Device is disconnected.

   In this case, the USB UHCI Controller is the owner of the port before the disconnect occurs. The disconnect is reported by the USB UHCI Controller

and serviced by the associated UHCI driver. The port-routing logic in the EHC cluster forces the Port Owner bit to 0, indicating that the EHC owns the unconnected port.

3. Configure Flag = 1 and a USB High-speed-capable Device is disconnected.
   In this case, the USB EHCI Controller is the owner of the port before, and remains the owner after, the disconnect occurs. The EHCI hardware and driver handle the disconnection process. The USB UHCI Controller never sees a device attached.

5.18.9.4 Effect of Resets on Port-Routing Logic

As mentioned above, the Port Routing logic is implemented in the Suspend power well so that reenumeration and remapping of the USB ports is not required following entering and exiting a system sleep state in which the core power is turned off.

Table 113. Effect of Resets on Port-Routing Logic

Reset Event	Effect on Configure Flag	Effect on Port Owner Bits
Suspend Well Reset	cleared (0)	set (1)
Core Well Reset	no effect	no effect
D3-to-D0 Reset	no effect	no effect
HCRESET	cleared (0)	set (1)

5.18.10 USB 2.0 Legacy Keyboard Operation

The Intel® 6300ESB ICH must support the possibility of a keyboard downstream from either a USB UHCI or a USB EHCI port. The description of the legacy keyboard support is unchanged from USB UHCI (See Section 5.17.9, “USB Legacy Keyboard Operation”).

The EHC provides the basic ability to generate SMIs on an interrupt event, along with more sophisticated control of the generation of SMIs.

5.18.11 USB 2.0 EHCI Based Debug Port

The Intel® 6300ESB ICH supports the elimination of the legacy COM ports by providing the ability for new debugger software to interact with devices on a USB EHCI port.

High-level restrictions and features are:

- Must be operational before USB EHCI drivers are loaded.
- Must work even when the port is disabled.
- Must work even though non-configured port is default-routed to the classic controller. Note that the Debug Port cannot be used to debug an issue that requires a classic USB device on Port #0 using the UHCI drivers.
- Must allow normal system USB EHCI traffic in a system that may only have one USB port.
- Debug Port device (DPD) must be High-Speed capable and connect to a High-Speed port on Intel® 6300ESB ICH systems.
- Debug Port FIFO must always make forward progress (a bad status on USB is simply presented back to software)
- The Debug Port FIFO is only given one USB access per microframe

5.18.11.1 Overview

The Debug port facilitates OS and device driver debug. It allows the software to communicate with an external console using a USB EHCI connection. Since the interface to this link does not go through the normal USB EHCI stack, it allows communication with the external console during cases where the OS is not loaded, the USB EHCI software is broken, or where the USB EHCI software is being debugged.

Specific features of this implementation of a debug port are:

- Only works with an external USB 2.0 debug device (console)
- Implemented for a specific port on the host controller
- Operational anytime the port is not suspended AND the host controller is in D0 power state.
- Capability is interrupted when port is driving USB RESET

5.18.11.2 Theory of Operation

There are two operational modes for the USB debug port:

1. Mode 1 is when the USB port is in a disabled state from the viewpoint of a standard host controller driver. In Mode 1, the Debug Port controller is required to generate a "keepalive" packets less than 2 ms apart to keep the attached debug device from suspending. The keepalive packet should be a standalone 32-bit SYNC field.
2. Mode 2 is when the host controller is running (i.e., Host controller's *Run/Stop#* bit is 1). In Mode 2, the normal transmission of SOF packets will keep the debug device from suspending.

Detail for the registers mentioned in the next sections can be found in Section 11.2.3, "USB 2.0-Based Debug Port Register" and in Section 11.2, "Memory-Mapped I/O Registers".

Behavioral Rules:

1. In both modes 1 and 2, the Debug Port controller must check for software requested debug transactions at least every 125 microseconds.
2. When the debug port is enabled by the debug driver, and the standard host controller driver resets the USB port, USB debug transactions are held off for the duration of the reset and until after the first SOF is sent.
3. When the standard host controller driver suspends the USB port, then USB debug transactions are held off for the duration of the suspend/resume sequence and until after the first SOF is sent.
4. The ENABLED_CNT bit in the debug register space is independent of the similar port control bit in the associated Port Status and Control register.

Table 114 describes the debug port behavior related to the state of bits in the debug registers as well as bits in the associated Port Status and Control register.

Table 114. USB Debug Port Behavior (Sheet 1 of 2)

OWNER_CNT	ENABLED_CNT	Port Enable	Run / Stop	Suspend	Debug Port Behavior
0	X	X	X	X	Debug port is not being used. Normal operation.
1	0	X	X	X	Debug port is not being used. Normal operation.
1	1	0	0	X	Debug port in Mode 1. SYNC keep alives sent plus debug traffic.

Table 114. USB Debug Port Behavior (Sheet 2 of 2)

OWNER_CNT	ENABLED_CNT	Port Enable	Run / Stop	Suspend	Debug Port Behavior
1	1	0	1	X	Debug port in Mode 2. SOF (and only SOF) is sent as keepalive. Debug traffic is also sent. Note that no other normal traffic is sent out this port, because the port is not enabled.
1	1	1	0	0	Illegal. Host controller driver should never put controller into this state (enabled, not running and not suspended).
1	1	1	0	1	Port is suspended. No debug traffic sent.
1	1	1	1	0	Debug port in Mode 2. Debug traffic is interspersed with normal traffic.
1	1	1	1	1	Port is suspended. No debug traffic sent.

5.18.11.2.1 OUT Transactions

An Out transaction sends data to the debug device. It may occur only when the following are true:

1. The debug port is enabled.
2. The debug software sets the GO_CNT bit.
3. The WRITE_READ#_CNT bit is set.

The sequence of the transaction is listed below.

1. Software sets the appropriate values in these bits:
   - USB_ADDRESS_CNF
   - USB_ENDPOINT_CNF
   - DATA_BUFFER[63:0]
   - TOKEN_PID_CNT[7:0]
   - SEND_PID_CNT[15:8]
   - DATA_LEN_CNT
   - WRITE_READ#_CNT (**Note:** This will always be 1 for OUT transactions.)
   - GO_CNT (**Note:** This will always be 1 to initiate the transaction.)

2. The debug port controller sends a token packet consisting of:
   a. SYNC
   b. TOKEN_PID_CNT field
   c. USB_ADDRESS_CNT field
   d. USB_ENDPOINT_CNT field
   e. 5-bit CRC field
3. After sending the token packet, the debug port controller sends a data packet consisting of
   a. SYNC
   b. SEND_PID_CNT field
   c. The number of data bytes indicated in DATA_LEN_CNT from the DATA_BUFFER
   d. 16-bit CRC

Note: A DATA_LEN_CNT value of zero is valid in which case no data bytes would be included in the packet.

4. After sending the data packet, the controller waits for a handshake response from the debug device.

- When a handshake is received, the debug port controller:
  a. Places the received PID in the RECEIVED_PID_STS field
  b. Resets the ERROR_GOOD#_STS bit
  c. Sets the DONE_STS bit
- When no handshake PID is received, the debug port controller:
  a. Sets the EXCEPTION_STS field to 001b
  b. Sets the ERROR_GOOD#_STS bit
  c. Sets the DONE_STS bit

5.18.11.2.2 IN Transactions

An IN transaction receives data from the debug device. It may occur only when the following are true:

1. The debug port is enabled
2. The debug software sets the GO_CNT bit
3. The WRITE_READ#_CNT bit is reset

The sequence of the transaction is:

1. Software sets the appropriate values in the following bits:
   — USB_ADDRESS_CNF
   — USB_ENDPOINT_CNF
   — TOKEN_PID_CNT[7:0]
   — DATA_LEN_CNT
   — WRITE_READ#_CNT (**Note:** This will always be 0 for IN transactions.)
   — GO_CNT (**Note:** This will always be 1 to initiate the transaction.)
2. The debug port controller sends a token packet consisting of:
   a. SYNC
   b. TOKEN_PID_CNT field

c. USB_ADDRESS_CNT field
d. USB_ENDPOINT_CNT field
e. 5-bit CRC field.

3. After sending the token packet, the debug port controller waits for a response from the debug device. When a response is received:
   a. The received PID is placed into the RECEIVED_PID_STS field
   b. Any subsequent bytes are placed into the DATA_BUFFER
   c. The DATA_LEN_CNT field is updated to show the number of bytes that were received after the PID.
4. When valid packet was received from the device that was one byte in length (indicating it was a handshake packet), then the debug port controller:
   a. Resets the ERROR_GOOD#_STS bit
   b. Sets the DONE_STS bit
5. When valid packet was received from the device that was more than one byte in length (indicating it was a data packet), then the debug port controller:
   a. Transmits an ACK handshake packet
   b. Resets the ERROR_GOOD#_STS bit
   c. Sets the DONE_STS bit
6. When no valid packet is received, then the debug port controller:
   a. Sets the EXCEPTION_STS field to 001b
   b. Sets the ERROR_GOOD#_STS bit
   c. Sets the DONE_STS bit.

5.18.11.2.3 Debug Software

Enabling the Debug Port

There are two mutually exclusive conditions that debug software must address as part of its startup processing:

1. The EHCI has been initialized by system software
2. The EHCI has not been initialized by system software

Debug software may determine the current 'initialized' state of the EHCI by examining the Configure Flag in the EHCI USB 2.0 Command Register. See Section 11.2.2.8, "Offset CAPLENGTH + 40 - 43h: CONFIGFLAG—Configure Flag Register" for information regarding offset 40h, bit 0. When this flag is set, then system software has initialized the EHCI. Otherwise, the EHCI should not be considered initialized. Debug software will initialize the debug port registers depending on the state the EHCI. However, before this may be accomplished, debug software must determine which root USB port is designated as the debug port.

Determining the Debug Port

Debug software may determine which USB root port has been designated as the debug port by examining bits 20:23 of the EHCI Host Controller Structural Parameters register. See Section 11.2.1.3, "Offset 04 - 07h: HCSPARAMS—Host Controller Structural Parameters" for information regarding offset 04h. This 4-bit field represents the numeric value assigned to the debug port (i.e., 0000=port 0).

Debug Software Startup with Non-Initialized EHCI

Debug software may attempt to use the debug port after setting the OWNER_CNT bit in the Control/Status Register, Section 11.2.3.1, “Offset 00h: Control/Status Register”, offset 00h, bit 30, and the Current Connect Status bit in the appropriate (See Determining the Debug Port) PORTSC register is set. See Section 11.2.2.9, “PORTSC-Port N Status and Control” for information regarding bit 0. When the Current Connect Status bit is not set, then debug software may choose to terminate or it may choose to wait until a device is connected.

When a device is connected to the port, then debug software must reset/enable the port. Debug software does this by setting and then clearing the Port Reset bit the PORTSC register. To ensure a successful reset, debug software should wait at least 50 ms before clearing the Port Reset bit. Due to possible delays, this bit may not change to zero immediately; reset is complete when this bit reads as zero. Software must not continue until this bit reads zero.

When a high-speed device is attached, the EHCI will automatically set the Port Enabled/Disabled bit in the PORTSC register and the debug software may proceed. Debug software should set the ENABLED_CNT bit in the Debug Port Control/Status register, and then reset (clear) the Port Enabled/Disabled bit in the PORTSC register (so that the system host controller driver doesn't see an enabled port when it is first loaded).

Debug Software Startup with Initialized EHCI

Debug software may attempt to use the debug port when the Current Connect Status bit in the appropriate (See Determining the Debug Port) PORTSC register is set. When the Current Connect Status bit is not set, then debug software may choose to terminate or it may choose to wait until a device is connected.

When a device is connected, then debug software must set the OWNER_CNT bit and then the ENABLED_CNT bit in the Debug Port Control/Status register. See Section 11.2.3.1, “Offset 00h: Control/Status Register” for information regarding offset 00h, bits 30 and 28.

Determining Debug Peripheral Presence

After enabling the debug port functionality, debug software may determine when a debug peripheral is attached by attempting to send data to the debug peripheral. When all attempts result in an error (Exception bits in the Debug Port Control/Status register indicates a Transaction Error), the attached device is not a debug peripheral. See Section 11.2.3.1, “Offset 00h: Control/Status Register” for information regarding offset 00h, bits [9:7]. When the debug port peripheral is not present, then debug software may choose to terminate or it may choose to wait until a debug peripheral is connected.

5.19 SMBus Controller Functional Description (D31:F3)

5.19.1 Overview

The Intel® 6300ESB ICH provides an SMBus 2.0 compliant Host Controller as well as an SMBus slave interface.

The host controller provides a mechanism for the processor to initiate communications with SMBus peripherals (slaves). The Intel® 6300ESB ICH is also capable of operating in a mode in which it may communicate with I^2C compatible devices.

The Intel® 6300ESB ICH may perform SMBus messages with either packet error checking (PEC) enabled or disabled. The actual PEC calculation and checking is performed in software.The SMBus Host Controller logic may automatically append the CRC byte when configured to do so.

The Slave Interface allows an external master to read from or write to the Intel® 6300ESB ICH. Write cycles may be used to cause certain events or pass messages, and the read cycles may be used to determine the state of various status bits. The Intel® 6300ESB ICH's internal Host Controller cannot access the Intel® 6300ESB ICH's internal Slave Interface.

The Intel® 6300ESB ICH SMBus logic exists in Device 31:Function 3 configuration space, and consists of a transmit data path, and host controller. The transmit data path provides the data flow logic needed to implement the seven different SMBus command protocols and is controlled by the host controller. The Intel® 6300ESB ICH SMBus controller logic is clocked by RTC clock.

The SMBus Address Resolution Protocol (ARP) is supported by using the existing host controller commands through software, except for the new Host Notify command (which is actually a received message).

The programming model of the host controller is combined into two portions: a PCI configuration portion, and a system I/O mapped portion. All static configuration, such as the I/O base address, is done through the PCI configuration space. Real-time programming of the Host interface is done in system I/O space.

5.19.2 Host Controller

The SMBus Host Controller is used to send commands to other SMBus slave devices. Software sets up the host controller with an address, command, and, for writes, data and optional PEC; and then tells the controller to start. When the controller has finished transmitting data on writes, or receiving data on reads, it will generate an SMI# or interrupt, when enabled.

The host controller supports seven command protocols of the SMBus interface (see the SMBus Specification): Quick Command, Send Byte, Receive Byte, Write Byte/Word, Read Byte/Word, ProcessCall, Block Read, Block Write and Block Write-Block Read process call.

The SMBus Host Controller requires that the various data and command fields be setup for the type of command to be sent. When software sets the START bit, the SMBus Host Controller will perform the requested transaction, and interrupt the processor (or generate an SMI#) when the transaction is completed. Once a START command has been issued, the values of the "active registers" (Host Control, Host Command, Transmit Slave Address, Data 0, Data 1) should not be changed or read until the interrupt status bit (INTR) has been set (indicating the completion of the command). Any register values needed for computation purposes should be saved prior to issuing of a new command, as the SMBus Host Controller will update all registers while completing the new command.

Using the SMBus Host Controller to send commands to the Intel® 6300ESB ICH's SMBus slave port is supported.

The Intel® 6300ESB ICH supports slave functionality, including the Host Notify protocol, on the SMLink pins when in TCO compatible mode. Therefore, in order to be fully compliant with the SMBus 2.0 specification (which requires the Host Notify cycle), the SMLink and SMBus signals must be tied together externally. However, this requirement to tie both SMLink and SMBus signals externally is not needed in advanced TCO mode as the slave functionality is available on the SMBus pins.

5.19.2.1 Command Protocols

In all of the following commands, the Host Status Register (offset 00h) is used to determine the progress of the command. While the command is in operation, the HOST_BUSY bit is set. When the command completes successfully, the INTR bit will be set in the Host Status Register. When the device does not respond with an acknowledge, and the transaction times out, the DEV_ERR bit is set. When software sets the KILL bit in the Host Control Register while the command is running, the transaction will stop and the FAILED bit will be set. When the KILL bit is set, the Intel® 6300ESB ICH will abort current transaction by asserting SMBCLK low for greater than the timeout period, assert a STOP condition and then releases SMBCLK and SMBDATA. However, setting the KILL bit does not affect SMLINK or TCO transactions or causes Intel® 6300ESB ICH to force a timeout when it is not performing a transaction.

Quick Command

When programmed for a Quick Command, the Transmit Slave Address Register is sent. The PEC byte is never appended to the Quick Protocol. Software should force the PEC_EN bit to '0' when performing the Quick Command. Software must force the I2C_EN bit to 0 when running this command. The format of the protocol is shown in Table 115.

Table 115. Quick Protocol

Bit	Description
1	Start Condition
2–8	Slave Address - 7 bits
9	Read / Write Direction
10	Acknowledge from slave
11	Stop

Send Byte / Receive Byte

For the Send Byte command, the Transmit Slave Address and Device Command Registers are sent
For the Receive Byte command, the Transmit Slave Address Register is sent. The data received is stored in the DATA0 register. Software must force the I2C_EN bit to 0 when running this command.

The Receive Byte is similar to a Send Byte, the only difference is the direction of data transfer. The format of the protocol is shown in Table 116 and Table 117.

Table 116. Send / Receive Byte Protocol without PEC

Send Byte Protocol		Receive Byte Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Read
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Data byte from slave
19	Acknowledge from slave	19	NOT Acknowledge
20	Stop	20	Stop

Table 117. Send/Receive Byte Protocol with PEC

Send Byte Protocol		Receive Byte Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Read
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Data byte from slave
19	Acknowledge from slave	19	Acknowledge
20–27	PEC	20–27	PEC from slave
28	Acknowledge from slave	28	Not Acknowledge
29	Stop	29	Stop

Write Byte/Word

The first byte of a Write Byte/Word access is the command code. The next 1 or 2 bytes are the data to be written. When programmed for a Write Byte/Word command, the Transmit Slave Address, Device Command, and Data0 Registers are sent. In addition, the Data1 Register is sent on a Write Word command. Software must force the I2C_EN bit to 0 when running this command. The format of the protocol is shown in Table 118 and Table 119.

Table 118. Write Byte/Word Protocol without PEC

Write Byte Protocol		Write Word Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
20–27	Data Byte - 8 bits	20–27	Data Byte Low - 8 bits
28	Acknowledge from Slave	28	Acknowledge from Slave
29	Stop	29–36	Data Byte High - 8 bits
		37	Acknowledge from slave
		38	Stop

Table 119. Write Byte/Word Protocol with PEC

Write Byte Protocol		Write Word Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Write

Table 119. Write Byte / Word Protocol with PEC

Write Byte Protocol		Write Word Protocol	
Bit	**Description**	**Bit**	**Description**
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
20–27	Data Byte - 8 bits	20–27	Data Byte Low - 8 bits
28	Acknowledge from Slave	28	Acknowledge from Slave
29–36	PEC	29–36	Data Byte High - 8 bits
37	Acknowledge from Slave	37	Acknowledge from slave
38	Stop	38–45	PEC
		46	Acknowledge from slave
		47	Stop

Read Byte / Word

Reading data is slightly more complicated than writing data. First the Intel® 6300ESB ICH must write a command to the slave device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The slave then returns 1 or 2 bytes of data. Software must force the I2C_EN bit to 0 when running this command.

When programmed for the read byte/word command, the Transmit Slave Address and Device Command Registers are sent. Data is received into the DATA0 on the read byte, and the DATA0 and DATA1 registers on the read word. The format of the protocol is shown in Table 120 and Table 121.

Table 120. Read Byte / Word Protocol without PEC

Read Byte Protocol		Read Word Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
20	Repeated Start	20	Repeated Start
21–27	Slave Address - 7 bits	21–27	Slave Address - 7 bits
28	Read	28	Read
29	Acknowledge from slave	29	Acknowledge from slave
30–37	Data from slave - 8 bits	30–37	Data Byte Low from slave - 8 bits
38	NOT acknowledge	38	Acknowledge
39	Stop	39–46	Data Byte High from slave - 8 bits
		47	NOT acknowledge
		48	Stop

Table 121. Read Byte / Word Protocol with PEC

Read Byte Protocol		Read Word Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
20	Repeated Start	20	Repeated Start
21–27	Slave Address - 7 bits	21–27	Slave Address - 7 bits
28	Read	28	Read
29	Acknowledge from slave	29	Acknowledge from slave
30–37	Data from slave - 8 bits	30–37	Data Byte Low from slave - 8 bits
38	Acknowledge	38	Acknowledge
39–46	PEC from slave	39–46	Data Byte High from slave - 8 bits
47	NOT Acknowledge	47	Acknowledge
48	Stop	48–55	PEC from slave
		56	NOT acknowledge
		57	Stop

Process Call

The process call is so named because a command sends data and waits for the slave to return a value dependent on that data. The protocol is simply a Write Word followed by a Read Word, but without a second command or stop condition.

When programmed for the Process Call command, the Intel® 6300ESB ICH transmits the Transmit Slave Address, Host Command, DATA0 and DATA1 registers. Data received from the device is stored in the DATA0 and DATA1 registers. The Process Call command with I2C_EN set and the PEC_EN bit set produces undefined results. Software must force either I2C_EN or PEC_EN to 0 when running this command. The format of the protocol is shown in Table 122 and Table 123.

Note: For process call command, the value written into bit 0 of the Transmit Slave Address Register (SMBus I/O register, offset 04h) needs to be 0.

Table 122. Process Call Protocol without PEC

Bit	Description
1	Start
2–8	Slave Address - 7 bits
9	Write
10	Acknowledge from Slave
11–18	Command code - 8 bits
19	Acknowledge from slave
20–27	Data byte Low - 8 bits
28	Acknowledge from slave
29–36	Data Byte High - 8 bits
37	Acknowledge from slave
38	Repeated Start
39–45	Slave Address - 7 bits
46	Read
47	Acknowledge from slave
48–55	Data Byte Low from slave - 8 bits
56	Acknowledge
57–64	Data Byte High from slave - 8 bits
65	NOT acknowledge
66	Stop

Table 123. Process Call Protocol with PEC (Sheet 1 of 2)

Bit	Description
1	Start
2–8	Slave Address - 7 bits
9	Write
10	Acknowledge from Slave
11–18	Command code - 8 bits

Table 123. Process Call Protocol with PEC (Sheet 2 of 2)

Bit	Description
19	Acknowledge from slave
20–27	Data byte Low - 8 bits
28	Acknowledge from slave
29–36	Data Byte High - 8 bits
37	Acknowledge from slave
38	Repeated Start
39–45	Slave Address - 7 bits
46	Read
47	Acknowledge from slave
48–55	Data Byte Low from slave - 8 bits
56	Acknowledge
57–64	Data Byte High from slave - 8 bits
65	Acknowledge
66–73	PEC from slave
74	NOT acknowledge
75	Stop

Block Read/Write

The Intel® 6300ESB ICH contains a 32-byte buffer for read and write data which may be enabled by setting bit '1' of the Auxiliary Control register at offset 0Dh in I/O space, as opposed to a single byte of buffering. This 32-byte buffer is filled with write data before transmission, and filled with read data on reception. In the Intel® 6300ESB ICH, the interrupt is generated only after a transmission or reception of 32 bytes, or when the entire byte count has been transmitted/received.

This requires the Intel® 6300ESB ICH to check the byte count field. Currently, the byte count field is transmitted but ignored by the hardware as software will end the transfer after all bytes it cares about have been sent or received.

For a Block Write software must either force the I2C_EN bit or both the PEC_EN and AAC bits to 0 when running this command.

SMBus mode: The block write begins with a slave address and a write condition. After the command code the Intel® 6300ESB ICH issues a byte count describing how many more bytes will follow in the message. When a slave had 20 bytes to send, the first byte would be the number 20 (14h), followed by 20 bytes of data. The byte count may not be 0. A Block Read or Write is allowed to transfer a maximum of 32 data bytes.

When programmed for a block write command, the Transmit Slave Address, Device Command, and Data0 (count) registers are sent. Data is then sent from the Block Data Byte register; the total data sent being the value stored in the Data0 Register. On block read commands, the first byte received is stored in the Data0 register, and the remaining bytes are stored in the Block Data Byte register.

The format of the Block Read/Write protocol is shown in Table 124 and Table 125.

I²C Mode: For Block Write, when the I²C_EN bit is set, the format of the command changes slightly. The Intel® 6300ESB ICH will still send the number of bytes (on writes) or receive the number of bytes (on reads) indicated in the DATA0 register. However, it will not send the contents of the DATA0 register as part of the message.

Table 124. Block Read / Write Protocol without PEC

Block Write Protocol		Block Read Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave Address - 7 bits	2–8	Slave Address - 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
20–27	Byte Count - 8 bits (Skip this step when I²C_En bit set)	20	Repeated Start
28	Acknowledge from Slave (Skip this step when I2C_EN bit set)	21–27	Slave Address - 7 bits
29–36	Data Byte 1 - 8 bits	28	Read
37	Acknowledge from Slave	29	Acknowledge from slave
38–45	Data Byte 2 - 8 bits	30–37	Byte Count from slave - 8 bits
46	Acknowledge from slave	38	Acknowledge
...	Data Bytes / Slave Acknowledges...	39–46	Data Byte 1 from slave - 8 bits
...	Data Byte N - 8 bits	47	Acknowledge
...	Acknowledge from Slave	48–55	Data Byte 2 from slave - 8 bits
...	Stop	56	Acknowledge
		...	Data Bytes from slave/ Acknowledge
		...	Data Byte N from slave - 8 bits
		...	NOT Acknowledge
		...	Stop

Table 125. Block Read / Write Protocol with PEC (Sheet 1 of 2)

Block Write Protocol		Block Read Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
2–8	Slave address - 7 bits	2–8	Slave address - 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
11–18	Command code - 8 bits	11–18	Command code - 8 bits

Table 125. Block Read/Write Protocol with PEC (Sheet 2 of 2)

Block Write Protocol		Block Read Protocol	
Bit	**Description**	**Bit**	**Description**
19	Acknowledge from slave	19	Acknowledge from slave
20–27	Byte count - 8 bits (Skip this step when I²C_En bit set)	20	Repeated start
28	Acknowledge from slave (skip this step when I²C_EN bit set)	21–27	Slave address - 7 bits
29–36	Data byte 1 - 8 bits	28	Read
37	Acknowledge from slave	29	Acknowledge from slave
38–45	Data byte 2 - 8 bits	30–37	Byte count from slave - 8 bits
46	Acknowledge from slave	38	Acknowledge
...	Data bytes/slave Acknowledges	39–46	Data byte 1 from slave - 8 bits
...	Data byte N - 8 bits	47	Acknowledge
...	Acknowledge from slave	48–55	Data byte 2 from slave - 8 bits
...	PEC - 8 bits	56	Acknowledge
...	Acknowledge from slave	...	Data bytes from slave/acknowledge
...	Stop	...	Data byte N from slave - 8 bits
		...	Acknowledge
		...	PEC from slave - 8 bits
		...	NOT acknowledge
		...	Stop

I²C Block Read

This command allows the Intel® 6300ESB ICH to perform block reads to certain I²C devices, such as serial E²PROMs. The SMBus Block Read supports the 7-bit addressing mode only. However, this does not allow access to devices using the I²C "Combined Format" that has data bytes after the address. Typically these data bytes correspond to an offset (address) within the serial memory chips.

Note: This command is supported independent of the setting of the I2C_EN bit. The I²C Read command with the PEC_EN bit set produces undefined results. Software must force both the PEC_EN and AAC bit to 0 when running this command.

For I²C Read command, the value written into bit 0 of the Transmit Slave Address Register (SMBus I/O register, offset 04h) needs to be 0.

To support these devices, the Intel® 6300ESB ICH implements an I²C Read command with the format: shown in Table 126.

Table 126. I²C Block Read Protocol (Sheet 1 of 2)

Bit	Description
1	Start
2-8	Slave Address - 7 bits
9	Write

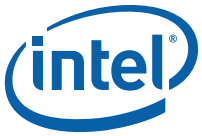


Table 126. I²C Block Read Protocol (Sheet 2 of 2)

Bit	Description
10	Acknowledge from slave
11-18	Send DATA1 register
19	Acknowledge from slave
20	Repeated Start
20-27	Slave Address - 7 bits
28	Read
29	Acknowledge from slave
30-38	Data Byte 1 from slave - 8 bits
39	Acknowledge
39-46	Data Byte 2 from slave - 8 bits
47	Acknowledge
...	Data Bytes from slave/ Acknowledge
...	Data Byte N from slave - 8 bits
...	NOT Acknowledge
...	Stop

The Intel® 6300ESB ICH will continue reading data from the peripheral until the NAK is received.

Block Write-Block Read Process Call

The Block Write-Block Read process call is a two-part message. The call begins with a slave address and a write condition. After the command code the host issues a write byte count (M) that describes how many more bytes will be written in the first part of the message. If a master has 6 bytes to send, the byte count field will have the value 6 (0000 0110b), followed by the 6 bytes of data. The write byte count (M) cannot be zero.

The second part of the message is a block of read data beginning with a repeated start condition followed by the slave address and a Read bit. The next byte is the read byte count (N), which may differ from the write byte count (M). The read byte count (N) cannot be zero.

The combined data payload must not exceed 32 bytes. The byte length restrictions of this process call are summarized as follows:

- M ≥ 1 byte
- N ≥ 1 byte
- M + N ≤ 32 bytes

The read byte count does not include the PEC byte. The PEC is computed on the total message beginning with the first slave address and using the normal PEC computational rules. It is highly recommended that a PEC byte be used with the Block Write-Block Read Process Call. Software must do a read to the command register (offset 2h) to reset the 32 byte buffer pointer prior to reading the block data register.

Note: There is no STOP condition before the repeated START condition, and a NACK signifies the end of the read transfer.

Note: E32B bit in the Auxiliary Control register must be set when using this protocol.

Table 127. Block Write-Block Read Process Call Protocol With / Without PEC

Bit	Description
1	Start
2-8	Slave Address - 7 bits
9	Write
10	Acknowledge from Slave
11-18	Command code - 8 bits
19	Acknowledge from slave
20-27	Data Byte Count (M) - 8 bits
28	Acknowledge from Slave
29-36	Data Byte (1) - 8 bits
37	Acknowledge from slave
38-45	Data Byte (2) - 8 bits
46	Acknowledge from slave
...	...
	Data Byte (M) - 8 bits
	Acknowledge from slave
	Repeated Start
	Slave Address - 7 bits
	Read
	Acknowledge from slave
	Data Byte Count (N) from slave – 8 bits
	Acknowledge from master
	Data Byte (1) from slave – 8 bits
	Acknowledge from master
	Data Byte (2) from slave – 8 bits
	Acknowledge from master
...	...
	Data Byte Count (N) from slave – 8 bits
	Acknowledge from master (Skip if no PEC)
	PEC from slave (Skip if no PEC)
	NOT acknowledge
	Stop

5.19.2.2 I²C Behavior

When the I²C_EN bit is set, the Intel® 6300ESB ICH SMBus logic will instead be set to communicate with I²C devices. This forces the following changes:

1. The Process Call command will skip the Command code (and its associated acknowledge).
2. The Block Write command will skip sending the Byte Count (DATA0).

In addition, the Intel® 6300ESB ICH will support the new I²C Read command. This is independent of the I²C_EN bit.

Note: When operating in I²C mode the Intel® 6300ESB ICH will not use the 32-byte buffer for block commands.

5.19.2.3 Heartbeat for Use with the External LAN Controller

The Heartbeat method allows the Intel® 6300ESB ICH to send messages to an external LAN Controller when the processor is otherwise unable to do so. It uses the SMLINK I/F between the Intel® 6300ESB ICH and the external LAN Controller. The actual Heartbeat message is a Block Write. Only eight bytes are sent.

5.19.3 Bus Arbitration

Several masters may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The Intel® 6300ESB ICH must continuously monitor the SMBDATA line. When the Intel® 6300ESB ICH is attempting to drive the bus to a '1' by letting go of the SMBDATA line, and it samples SMBDATA low, then some other master is driving the bus and the Intel® 6300ESB ICH must stop transferring data.

When the Intel® 6300ESB ICH sees that it has lost arbitration, the condition is called a collision. The Intel® 6300ESB ICH will set the BUS_ERR bit in the Host Status Register, and when enabled, generate an interrupt or SMI#. The processor is responsible for restarting the transaction.

When the Intel® 6300ESB ICH is a SMBus master, it will drive the clock. When the Intel® 6300ESB ICH is sending address or command as an SMBus master, or data bytes as a master on writes, it will drive data relative to the clock it is also driving. It will not start toggling the clock until the start or stop condition meets proper setup and hold time. The Intel® 6300ESB ICH will also ensure minimum time between SMBus transactions as a master.

Note: The Intel® 6300ESB ICH supports the same arbitration protocol for both the SMBus and the System Management (SMLINK) interfaces.

5.19.4 Bus Timing

5.19.4.1 Clock Stretching

Some devices may not be able to handle their clock toggling at the rate that the Intel® 6300ESB ICH as an SMBus master would like. They have the capability of stretching the low time of the clock. When the Intel® 6300ESB ICH attempts to release the clock (allowing the clock to go high), the clock will remain low for an extended period of time.

The Intel® 6300ESB ICH must monitor the SMBus clock line after it releases the bus to determine whether to enable the counter for the high time of the clock. While the bus is still low, the high time counter must not be enabled. Similarly, the low period of the clock may be stretched by an SMBus master when it is not ready to send or receive data.

5.19.4.2 Bus Time Out (Intel® 6300ESB ICH as SMBus Master)

When there is an error in the transaction, such that an SMBus device does not signal an acknowledge, or holds the clock lower than the allowed time-out time, the transaction will time out. The Intel® 6300ESB ICH will discard the cycle, and set the DEV_ERR bit. The time out minimum is 25 ms. The time-out counter inside the Intel® 6300ESB ICH will start after the last bit of data is transferred by the Intel® 6300ESB ICH and it is waiting for a response. The 25 ms will be a count of 800 RTC clocks.

5.19.5 Interrupts/SMI#

The Intel® 6300ESB ICH SMBus controller uses PIRQB# as its interrupt pin. However, the system may alternatively be set up to generate SMI# instead of an interrupt, by setting the SMBUS_SMI_EN bit.

Table 129 and Table 130 specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and Slave SMI, and Wake internal signals. The rows in the tables are additive, which means that when more than one row is true for a particular scenario then the results for all of the activated rows will occur.

Table 128. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	SMBALERT_DIS (Slave Command I/O Register, Offset 11h, Bit 2)	Result
SMBALERT# asserted low (always reported in Host Status Register, Bit 5)	X	X	X	Wake generated
	X	1	0	Slave SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 129. Enables for SMBus Slave Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit1)	Event
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS).
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

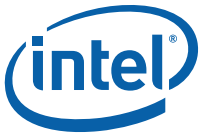


Table 130. Enables for the Host Notify Command

HOST_NOTIFY_INTREN (Slave Control I/O Register, Offset 11h, bit 0)	SMB_SMI_EN (Host Config Register, D31:F3:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Slave Control I/O Register, Offset 11h, bit 1)	Result
0	X	0	None
X	X	1	Wake generated
1	0	X	Interrupt generated
1	1	X	Slave SMI# generated (SMBUS_SMI_STS)

5.19.6 SMBALERT#

SMBALERT# is multiplexed with GPIO[11]. When enabled and the signal is asserted, the Intel® 6300ESB ICH may generate an interrupt, an SMI# or a wake event from S1–S4.

Note: Any event on SMBALERT# (regardless whether it is programmed as a GPIO or not), causes the event message to be sent in “heartbeat mode.”

5.19.7 SMBus CRC Generation and Checking

When the AAC bit is set in the Auxiliary Control register, the Intel® 6300ESB ICH will automatically calculate and drive CRC at the end of the transmitted packet for write cycles, and will check the CRC for read cycles. It will not transmit the contents of the PEC register for CRC. The PEC bit must not be set in the Host Control register when this bit is set, or unspecified behavior will result.

When the read cycle results in a CRC error, the DEV_ERR bit and the CRCE bit in the Auxiliary Status register at offset 0Ch will be set.

5.19.8 SMBus Slave Interface

The Intel® 6300ESB ICH’s SMBus Slave interface is accessed through the SMLINK[1:0] signals. The SMBus slave logic will not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol (Alert on LAN*) device. The slave interface allows the Intel® 6300ESB ICH to decode cycles, and allows an external microcontroller to perform specific actions. Key features and capabilities include:

- Supports decode of three types of messages: Byte Write, Byte Read, and Host Notify
- Receive Slave Address register: This is the address that the Intel® 6300ESB ICH decodes. A default value is provided so that the slave interface may be used without the processor having to program this register.
- Receive Slave Data register in the SMBus I/O space that includes the data written by the external microcontroller
- Registers that the external microcontroller may read to get the state of the Intel® 6300ESB ICH. See Table 135.
- Status bits to indicate that the SMLink/SMBus slave logic caused an interrupt or SMI# due to the reception of a message that matched the slave address.
  - — Bit 0 of the Slave Status Register for the Host Notify command.

— Bit 16 of the SMI Status Register (Section 8.8.3.10, "SMI_STS—SMI Status Register") for all others.

Note: The external microcontroller should not attempt to access the Intel® 6300ESB ICH's SMBus slave logic until 1 second after both: RTEST# is high and RSMRST# is high.

When a master leaves the clock and data bits of the SMLink interface at '1' for 50 µs or more in the middle of a cycle, the Intel® 6300ESB ICH slave logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the slave logic.

Note: When an external microcontroller accesses the SMBus Slave Interface over the SMLink a translation in the address is needed to accommodate the least significant bit used for read/write control. For example, when the Intel® 6300ESB ICH slave address (RCV_SLVA) is left at 44h (default), the external microcontroller would use an address of 88h/89h (write/read).

5.19.8.1 Format of Slave Write Cycle

The external master performs Byte Write commands to the Intel® 6300ESB ICH SMBus Slave I/F. The "Command" field (bits 11-18) indicate which register is being accessed. The Data field (bits 20-27) indicates the value that should be written to that register.

The Write Cycle format is shown in Table 131. Table 132 has the values associated with the registers.

Table 131. Slave Write Cycle Format

Bits	Description	Driven by	Comment
1	Start Condition	External Microcontroller	
2–8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register.
9	Write	External Microcontroller	Always 0
10	ACK	Intel® 6300ESB ICH	
11–18	Command	External Microcontroller	This field indicates which register will be accessed. See Table 132 below for the register definitions
19	ACK	Intel® 6300ESB ICH	
20–27	Register Data	External Microcontroller	See Table 132 below for the register definitions
28	ACK	Intel® 6300ESB ICH	
29	Stop	External Microcontroller	

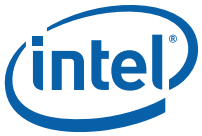


Table 132. Slave Write Registers

Register	Function
0	Command Register. See Table 133 for legal values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–7	Reserved
8	Frequency Straps will be written on bits 3:0. Bits 7:4 should be 0, but will be ignored.
9-FFh	Reserved

NOTE: The external microcontroller is responsible to make sure that it does not update the contents of the data byte registers until they have been read by the system processor. The Intel® 6300ESB ICH will overwrite the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. Intel® 6300ESB ICH will not attempt to cover this race condition (i.e. unpredictable results in this case).

Table 133. Command Types (Sheet 1 of 2)

Command Type	Description
0	Reserved
1	**Wake/SMI#:** Wake system if it is not already awake. If system is already awake, then an SMI# will be generated. **NOTE:** The SMB_WAK_STS bit will be set by this command, even when the system is already awake. The SMI handler should then clear this bit.
2	**Unconditional Powerdown:** This command sets the PWRBTNOR_STS bit, and has the same effect as the Powerbutton Override occurring.
3	**Hard Reset Without Cycling:** The will cause a hard reset of the system (does not include cycling of the power supply). This is equivalent to a write to the CF9h register with bits 2:1 set to 1, but bit 3 set to 0.
4	**Hard Reset System:** The will cause a hard reset of the system (including cycling of the power supply). This is equivalent to a write to the CF9h register with bits 3:1 set to 1.
5	**Disable the TCO Messages.** This command will disable the Intel® 6300ESB ICH from sending Heartbeat and Event messages (as described in Section 5.12.4, "Heartbeat and Event Reporting through SMLink/SMbus"). Once this command has been executed, Heartbeat and Event message reporting may only be re enabled by assertion and deassertion of the RSMRST# signal.
6	**WD Reload:** Reload watchdog timer.

Table 133. Command Types (Sheet 2 of 2)

Command Type	Description
7	Reserved
8	SMLINK_SLV_SMI. When Intel® 6300ESB ICH detects this command type while in the S0 state, it sets the SMLINK_SLV_SMI_STS bit (see Section 8.9.8, "TCO1_CNT—TCO1 Control Register"). This command should only be used when the system is in an S0 state. When the message is received during S1-S5 states, the Intel® 6300ESB ICH acknowledges it, but the SMLINK_SLV_SMI_STS bit does not get set. **NOTE:** It is possible that the system transitions out of the S0 state at the same time that the SMLINK_SLV_SMI command is received. In this case, the SMLINK_SLV_SMI_STS bit may get set but not serviced before the system goes to sleep. Once the system returns to S0, the SMI associated with this bit would then be generated. Software must be able to handle this scenario.
9-FFh	Reserved

5.19.8.2 Format of Read Command

The external master performs Byte Read commands to the Intel® 6300ESB ICH SMBus Slave I/F. The "Command" field (bits 11-18) indicate which register is being accessed. The Data field (bits 30-37) contains the value that should be read from that register.

Table 134 shows the Read Cycle format. Table 135 shows the register mapping for the data byte.

Table 134. Read Cycle Format (Sheet 1 of 2)

Bit	Description	Driven by:	Comment:
1	Start	External Microcontroller	
2-8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register.
9	Write	External Microcontroller	Always zero.
10	ACK	Intel® 6300ESB ICH	
11-18	Command code – 8 bits	External Microcontroller	Indicates which register is being accessed. See Table 135.
19	ACK	Intel® 6300ESB ICH	
20	Repeated Start	External Microcontroller	
21-27	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register.
28	Read	External Microcontroller	Always one.
29	ACK	Intel® 6300ESB ICH	

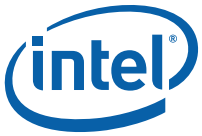


Table 134. Read Cycle Format (Sheet 2 of 2)

Bit	Description	Driven by:	Comment:
30-37	Data Byte	Intel® 6300ESB ICH	Value depends on register being accessed. See Table 135.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 135. Data Values for Slave Read Registers (Sheet 1 of 2)

Register	Bits	Description
0	7:0	Reserved.
1	2:0	System Power State 000 = S0 001 = S1 010 = Reserved 011 = S3 100 = S4 101 = S5 110 = Reserved 111 = Reserved
1	7:3	Reserved
2	3:0	Frequency Strap Register
2	7:4	Reserved
3	5:0	Watchdog Timer current value
3	7:6	Reserved
4	0	1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
4	1	1 = BTI Temperature Event occurred. This bit will be set when the Intel® 6300ESB ICH's THRM# input signal is active. Need to take after polarity control.
4	2	Boot-Status. This bit will be 1 when the processor does not fetch the first instruction.
4	3	This bit will be set after the TCO timer times out a second time (Both TIMEOUT and SECOND_TO_STS bits set).
4	6:4	Reserved.
4	7	The bit will reflect the state of the GPI[11]/SMBALERT# signal, and will depend on the GP_INV[11] bit. It does not matter if the pin is configured as GPI[11] or SMBALERT#. **NOTE:** When the GP_INV[11] bit is 1 then the value of register 4 bit 7 will equal the level of the GPI[11]/SMBALERT# pin (high = 1, low = 0). **NOTE:** When the GP_INV[11] bit is 0 then the value of register 4 bit 7 will equal the inverse of the level of the GPI[11]/SMBALERT# pin (high = 1, low = 0).
5	0	Unprogrammed FWH bit. This bit will be 1 to indicate that the first BIOS fetch returned FFh, which indicates that the FWH is probably blank.
5	1	**Battery Low Status.** '1' when the BATLOW# pin is a '0'.
5	2	**CPU Power Failure Status:** '1' when the CPUPWR_FLR bit in the GEN_PMCON_2 register is set.

Table 135. Data Values for Slave Read Registers (Sheet 2 of 2)

Register	Bits	Description
5	7:3	Reserved
6	7:0	Contents of the Message 1 register. See Section 8.9.10, "TCO_MESSAGE1 and TCO_MESSAGE2 Registers".
7	7:0	Contents of the Message 2 register. See Section 8.9.10, "TCO_MESSAGE1 and TCO_MESSAGE2 Registers".
8	7:0	Contents of the WDSTATUS register. See Section 8.9.11, "Offset TCOBASE + OEh: TCO_WDSTATUS—TCO2 Control Register".
9 – FFh	7:0	Reserved

Warning: The external microcontroller is responsible to make sure that it does not read the contents of the various message registers until they have been written by the system processor. The Intel® 6300ESB ICH will overwrite the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. The Intel® 6300ESB ICH will not attempt to cover this race condition (i.e., unpredictable results).

5.19.8.2.1 Behavioral Notes

According to SMBus protocol, Read and Write messages always begin with a Start bit - Address - Write bit sequence. When the Intel® 6300ESB ICH detects that the address matches the value in the Receive Slave Address register, it will assume that the protocol is always followed and ignore the Write bit (bit 9) and signal an Acknowledge during bit 10 (See Table 131 and Table 134). In other words, when a Start - Address - Read occurs (which is illegal for SMBus Read or Write protocol), and the address matches the Intel® 6300ESB ICH's Slave Address, the Intel® 6300ESB ICH will still grab the cycle.

Also according to SMBus protocol, a Read cycle contains a Repeated Start - Address - Read sequence beginning at bit 20 (See Table 134). Once again, when the Address matches the Intel® 6300ESB ICH's Receive Slave Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Slave Read cycle.

Note: An external microcontroller must not attempt to access the Intel® 6300ESB ICH's SMBus Slave logic until at least 1 second after both RTCRST# and RSMRST# are deasserted (high).

5.19.8.3 Format of Host Notify Command

The Intel® 6300ESB ICH tracks and responds to the standard Host Notify command as specified in the SMBus 2.0 specification. The host address for this command is fixed to 0001000b. When the Intel® 6300ESB ICH already has data for a previously-received host notify command which has not been serviced yet by the host software (as indicated by the HOST_NOTIFY_STS bit), it will NACK following the host address byte of the protocol. This allows the host to communicate non-acceptance to the master and retain the host notify address and data values for the previous cycle until host software completely services the interrupt.

Note: Host software must always clear the HOST_NOTIFY_STS bit after completing any necessary reads of the address and data registers.

Table 136 shows the Host Notify format.

Table 136. Host Notify Format

Bit	Description	Driven by:	Comment
1	Start	External Master	
2–8	SMBus Host Address - 7 bits	External Master	Always 0001_000
9	Write	External Master	Always zero.
10	ACK (or NACK)	Intel® 6300ESB ICH	Intel® 6300ESB ICH NACKs when HOST_NOTIFY_STS is 1.
11–17	Device Address – 7 bits	External Master	Indicates the address of the master; loaded into the Notify Device Address Register.
18	Unused - Always 0	External Master	7-bit-only address; this bit is inserted to complete the byte
19	ACK	Intel® 6300ESB ICH	
20–27	Data Byte Low - 8 bits	External Master	Loaded into the Notify Data Low Byte Register.
28	ACK	Intel® 6300ESB ICH	
29–36	Data Byte High - 8 bits	External Master	Loaded into the Notify Data High Byte Register.
37	ACK	Intel® 6300ESB ICH	
38	Stop	External Master	

5.20 AC'97 Controller Functional Description (Audio D31:F5, Modem D31:F6)

5.20.1 Overview

Note: All references to AC'97 in this document refer to the *Audio Codec'97 Component Specification, Version 2.2*. For further information on the operation of the AC-link protocol, please see the AC '97 specification.

The Intel® 6300ESB ICH AC'97 controller features include:

- Independent (FDX) channels for mono Line in and out.
- Supports 16 bit samples.
- Multiple sample rates up to 48 KHz
- Supports dual codec implementations for audio in dock
- Supports read/write access to all Primary and Secondary AC'97 registers
- Supports low latency access to 16 GPIO and wake up event status bits.

Note: The AC'97 Rev 2.0 spec. defines the following features which are NOT supported by the Intel® 6300ESB ICH:

- Support for optional double rate sampling (n+1 sample for PCM L, R and C)
- Support for 18 and 20 bit sample lengths
- Handset channels (In and Out)
- Dual Audio Codec support

Table 137 shows a detailed list of features supported by the Intel® 6300ESB ICH AC'97 digital controller.

Table 137. Features Supported by Intel® 6300ESB ICH (Sheet 1 of 2)

Feature	Description
System Interface	• Isochronous low latency bus master memory interface • Scatter/gather support for word-aligned buffers in memory (all mono or stereo 20-bit and 16-bit data types are supported, no 8-bit data types are supported) • Data buffer size in system memory from 3 to 65535 samples per input • Data buffer size in system memory from 0 to 65535 samples per output • Independent PCI audio and modem functions with configuration and IO spaces • AC'97 codec registers are shadowed in system memory through driver. • AC'97 codec register accesses are serialized through semaphore bit in PCI IO space (new accesses are not allowed while a prior access is still in progress).
Power Management	• Power management through PCI Power Management
PCI Audio Function	• Read/write access to audio codec registers 00h-3Ah and vendor registers 5Ah-7Eh • 20-bit stereo PCM output, up to 48 KHz (L,R, Center, Sub-woofer, L-rear and R-rear channels on slots 3,4,6,7,8,9,10,11) • 16-bit stereo PCM input, up to 48 KHz (L,R channels on slots 3,4) • 16-bit mono mic in w/ or w/o mono mix, up to 48 KHz (L,R channel, slots 3,4) (mono mix supports mono hardware AEC reference for speakerphone) • 16-bit mono PCM input, up to 48 KHz from dedicated mic ADC (slot 6) (supports speech recognition or stereo hardware AEC ref for speakerphone) • During cold reset AC_RST# is held low until after POST and software deassertion of AC_RST# (supports passive PC_BEEP to speaker connection during POST).

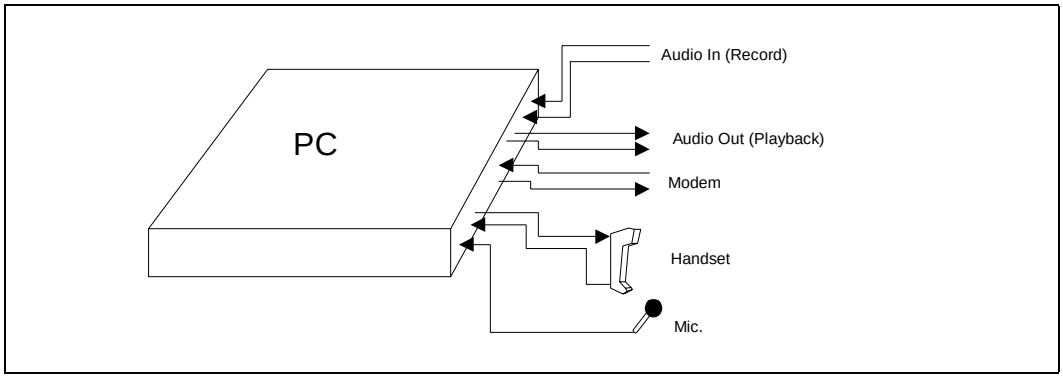
Table 137. Features Supported by Intel® 6300ESB ICH (Sheet 2 of 2)

Feature	Description
PCI Modem function	• Read/write access to modem codec registers 3Ch-58h and vendor registers 5Ah-7Eh • 16-bit mono modem line1 output and input, up to 48 KHz (slot 5) • Low latency GPIO[15:0] via hardwired update between slot 12 and PCI IO register. • Programmable PCI interrupt on modem GPIO input changes via slot 12 GPIO_INT • SCI event generation on AC_SDIN[2:0] wake-up signals
AC-link	• AC'97 2.0 compliant AC-link interface • Variable sample rate output support through AC'97 SLOTREQ protocol (slots 3,4,5,6,7,8,9,10,11) • Variable sample rate input support through monitoring of slot valid tag bits (slots 3,4,5,6) • 3.3 V digital operation meets AC'97 2.2 DC switching levels • AC-link IO driver capability meets AC'97 2.2 triple codec specifications • Codec register status reads must be returned with data in the next AC-link frame, per AC'97 2.2 spec.
Multiple Codec	• Triple codec addressing: All AC'97 Audio codec register accesses are addressable to codec ID 00 (primary), codec ID 01 (secondary), or codec ID 10 (tertiary). • Modem codec addressing: All AC'97 Modem codec register accesses are addressable to codec ID 00 (primary) or codec ID 01 (secondary). • Triple codec receive capability through AC_SDIN[2:0] pins (AC_SDIN[2:0] frames are internally validated, synch'd, and OR'd depending on the Steer Enable bit status in the SDM register) • AC_SDIN mapping to DMA engine mapping capability allows for simultaneous input from three different audio codecs. **NOTES:** 1. Audio Codec IDs are remappable and not limited to 00,01,10 2. Modem Codec IDs are remappable and limited to 00,01 3. When using multiple codecs, the Modem Codec must be ID 01.

Note: Throughout this document, references to D31:F5 indicate that the audio function exists in PCI Device 31, Function 5. References to D31:F6 indicate that the modem function exists in PCI Device 31, Function 6.

Note: Throughout this document, references to tertiary, third, or triple codecs refer to the third codec in the system connected to the AC_SDIN[2] pin. The AC'97 2.2 specification refers to non-primary codecs as multiple secondary codecs. To avoid confusion and excess verbiage this EDS refers to it as the third or tertiary codec.

Figure 23. Intel® 6300ESB ICH Based AC'97 Controller Connection to Companion Codec(s)



5.20.1.1 PCI Power Management

This Power Management section applies for all AC'97 controller functions. After a power management event is detected, the AC'97 controller will wake the host system. The sections below describe these events and the AC'97 controller power states.

Device Power States

The AC'97 controller supports D0 and D3 PCI Power Management states. Notes regarding the Intel® 6300ESB ICH AC'97 controller implementation of the Device States:

1. The AC'97 controller hardware does not inherently consume any more power when it is in the D0 state than it does in D3 state. However, software may halt the DMA engine prior to entering these low power states such that the maximum power consumption is reduced.
2. In the D0 state, all implemented AC'97 controller features are enabled.
3. In D3 state, accesses to the AC'97 controller memory-mapped or I/O range will result in master abort.
4. In D3 state, the AC'97 controller interrupt must never assert for any reason. The internal PME# signal is used to signal wake events, etc.
5. When the Device Power State field is written from $D3_{HOT}$ to D0, an internal reset is generated.
6. AC'97 STS bit will be set only when the audio or modem resume events were detected and their respective PME enable bits were set.
7. GPIO Status change interrupt no longer has a direct path to AC'97 STS bit. This will cause a wake up event only when the modem controller was in D3.
8. Resume events on AC_SDIN[2:0] will cause resume interrupt status bits to be set only if their respective controllers are not in D3.
9. Edge detect logic will prevent the interrupts from being asserted in case AC'97 controller is switched from D3 to D0 after a wake event.
10. Once the interrupt status bits are set, they will cause PIRQB# if their respective enable bits were set. One of the audio or the modem drivers will handle the interrupt.

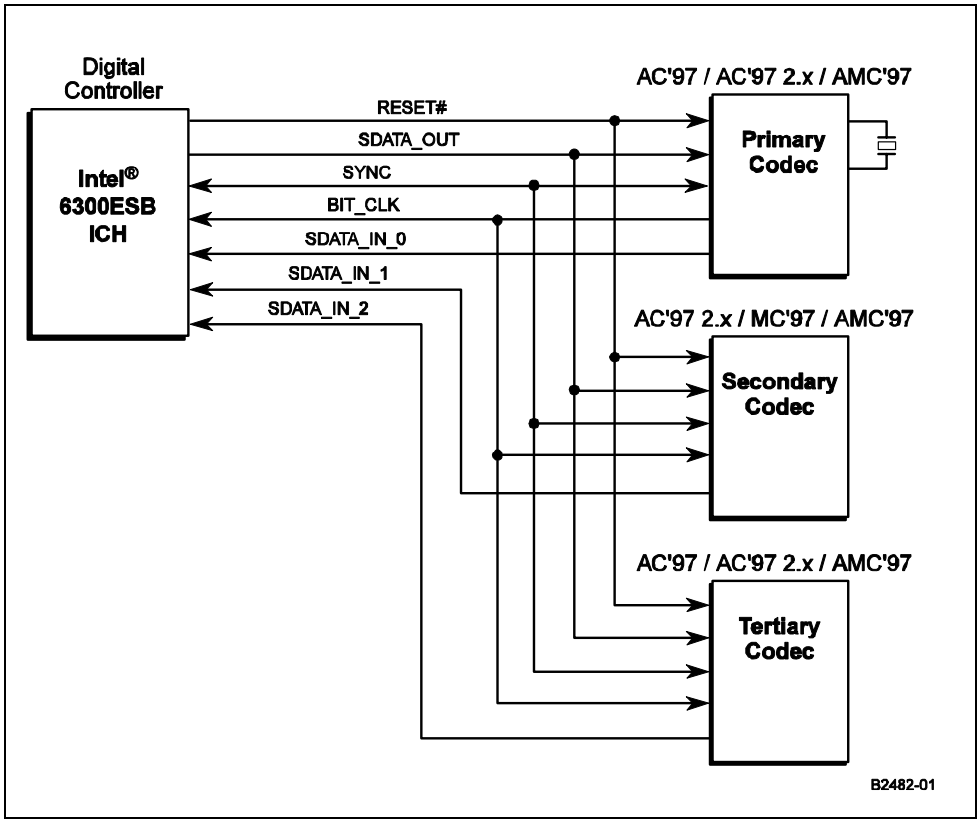
5.20.2 AC-Link Overview

The Intel® 6300ESB ICH is an AC'97 2.0 compliant controller that communicates with companion codecs through a digital serial link called the AC-link. All digital audio/ modem streams and command/status information is communicated over the AC-link.

The AC-link is a bi-directional, serial PCM digital stream. It handles multiple input and output data streams, as well as control register accesses, employing a time division multiplexed (TDM) scheme. The AC-link architecture provides for data transfer through individual frames transmitted in a serial fashion. Each frame is divided into 12 outgoing and 12 incoming data streams, or slots. The architecture of the Intel® 6300ESB ICH AC-link allows a maximum of three codecs to be connected. Figure 24 shows a three codec topology of the AC-link for the Intel® 6300ESB ICH.

The AC'97 modem controller is a separate PCI function. However, the AC'97 modem controller is implemented in the same logical unit as the AC'97 audio functions. There are registers declared in the audio function which contain modem information - these registers are also visible in the modem IO space, but are implemented as a single register.

Figure 24. AC'97 2.2 Controller-Codec Connection



The AC-link consists of a five signal interface between the controller and codec. Table 138 indicates the AC-link signal pins on the Intel® 6300ESB ICH and their associated power wells.

Table 138. AC'97 Signals

Signal Name	Type	Power Well	Description
AC_RESET#	Output	Resume	Master hardware reset
AC_SYNC	Output	Core	48 KHz fixed rate sample sync
AC_BIT_CLK	Input	Core	12.288 MHz Serial data clock
AC_SDOUT	Output	Core	Serial output data
AC_SDIN 0	Input	Resume	Serial input data
AC_SDIN 1	Input	Resume	Serial input data
AC_SDIN 2	Input	Resume	Serial input data

NOTE: Power well voltage levels are 3.3 V.

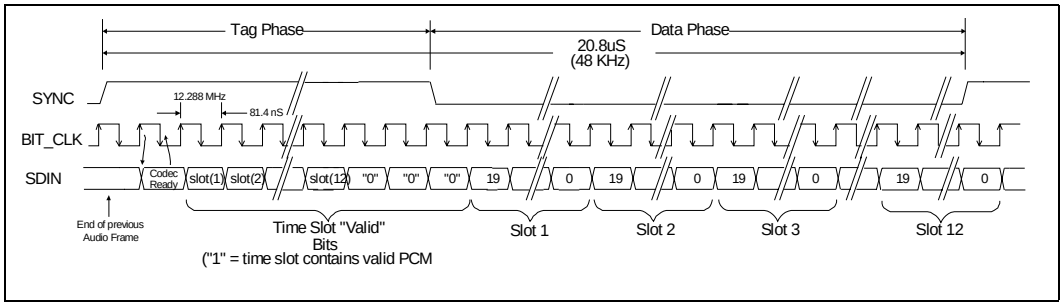
Intel® 6300ESB ICH core well outputs may be used as strapping options for the Intel® 6300ESB ICH, sampled during system reset. These signals may have weak pullups/ pulldowns on them, however this will not interfere with link operation. Intel® 6300ESB ICH inputs integrate weak pulldowns to prevent floating traces when a secondary and/ or tertiary codec is not attached. When the Shut Off bit in the control register is set, all buffers will be turned off and the pins will be held in a steady state, based on these pullups/pulldowns.

BIT_CLK is fixed at 12.288 MHz and is sourced by the primary codec. It provides the necessary clocking to support the twelve 20-bit time slots. AC-link serial data is transitioned on each rising edge of BIT_CLK. The receiver of AC-link data samples each serial bit on the falling edge of BIT_CLK.

When BIT_CLK makes no transitions for four consecutive PCI clocks, the Intel® 6300ESB ICH assumes the primary codec is not present or not working. It sets bit 28 of the Global Status Register (I/O offset 30h). All accesses to codec registers with this bit set will return data of FFh to prevent system hangs.

Synchronization of all AC-link data transactions is signaled by the AC'97 controller through the AC_SYNC signal, as shown in Figure 25. The primary codec drives the serial bit clock onto the AC-link, which the AC'97 controller then qualifies with the AC_SYNC signal to construct data frames. AC_SYNC, fixed at 48 KHz, is derived by dividing down BIT_CLK. AC_SYNC remains high for a total duration of 16 BIT_CLKs at the beginning of each frame. The portion of the frame where AC_SYNC is high is defined as the tag phase. The remainder of the frame where AC_SYNC is low is defined as the data phase. Each data bit is sampled on the falling edge of BIT_CLK.

Figure 25. AC-Link Protocol



The Intel® 6300ESB ICH has three AC_SDIN pins allowing a single, dual, or triple codec configuration. When multiple codecs are connected, the primary, secondary, and tertiary codecs may be connected to any AC_SDIN line. The Intel® 6300ESB ICH does

not distinguish between codecs on its AC_SDIN[2:0] pins, however the registers do distinguish between AC_SDIN[0], AC_SDIN[1], and AC_SDIN[2] for wake events, etc. When using a Modem Codec it is recommended to connect it to AC_SDIN[1].

See your Platform Design Guide for a matrix of valid codec configurations.

The Intel® 6300ESB ICH does not support optional test modes as outlined in the AC'97 specification.

5.20.2.1 AC-link Output Frame (SDOUT)

A new output frame begins with a low to high transition of AC_SYNC. AC_SYNC is synchronous to the rising edge of BIT_CLK. On the immediately following falling edge of BIT_CLK, the codec samples the assertion of AC_SYNC. This falling edge marks the time when both sides of AC-link are aware of the start of a new frame. On the next rising edge of BIT_CLK, the Intel® 6300ESB ICH transitions SDOUT into the first bit position of slot 0, or the valid frame bit. Each new bit position is presented to the AC-link on a rising edge of BIT_CLK, and subsequently sampled by the codec on the following falling edge of BIT_CLK. This sequence ensures that data transitions and subsequent sample points for both incoming and outgoing data streams are time aligned.

The output frame data phase corresponds to the multiplexed bundles of all digital output data targeting codec DAC inputs and control registers. Each output frame supports up to twelve outgoing data time slots. The Intel® 6300ESB ICH generates 16 or 20 bits and stuffs remaining bits with zeros.

The output data stream is sent with the most significant bit first, and all invalid slots are stuffed with zeros. When mono audio sample streams are output from the Intel® 6300ESB ICH, software must ensure both left and right sample stream time slots are filled with the same data.

5.20.2.2 Output Slot 0: Tag Phase

Slot 0 is considered the tag phase. The tag phase is a special 16 bit time slot wherein each bit conveys a valid tag for its corresponding time slot within the current frame. A one in a given bit position of slot 0 indicates that the corresponding time slot within the current frame has been assigned to a data stream and contains valid data. When a slot is tagged invalid with a zero in the corresponding bit position of slot 0, the Intel® 6300ESB ICH stuffs the corresponding slot with zeros during that slot's active time.

Within slot 0, the first bit is a valid frame bit (slot 0, bit 15) which flags the validity of the entire frame. When the valid frame bit is set to one, this indicates that the current frame contains at least one slot with valid data. When there is no transaction in progress, the Intel® 6300ESB ICH will deassert the frame valid bit. Note that after a write to slot 12, that slot will always stay valid, and therefore the frame valid bit will remain set.

The next 12 bit positions of slot 0 (bits [14:3]) indicate which of the corresponding twelve time slots contain valid data. Bits [1:0] of slot 0 are used as codec ID bits to distinguish between separate codecs on the link.

Using the valid bits in the tag phase allows data streams of differing sample rates to be transmitted across the link at its fixed 48 KHz frame rate. The codec may control the output sample rate of the Intel® 6300ESB ICH using the SLOTREQ bits as described in the AC'97 specification.

5.20.2.3 Output Slot 1: Command Address Port

The command port is used to control features and monitor status of AC'97 functions including, but not limited to, mixer settings and power management.

The control interface architecture supports up to 64 16-bit read/write registers, addressable on even byte boundaries. Only the even registers (00h, 02h, etc.) are valid.

Output frame slot 1 communicates control register address, and write/read command information.

In the case of the multiple codec implementation, accesses to the codecs are differentiated by the driver using address offsets 00h–7Fh for the primary codec, address offsets 80h–FEh for the secondary codec, and address offsets 100h–17Fh for the tertiary codec. The differentiation on the link, however, is done through the codec ID bits.

5.20.2.4 Output Slot 2: Command Data Port

The command data port is used to deliver 16-bit control register write data in the event that the current command port operation is a write cycle as indicated in slot 1, bit 19. When the current command port operation is a read, the entire slot time stuffed with zeros by the Intel® 6300ESB ICH. Bits [19:4] contain the write data. Bits [3:0] are reserved and are stuffed with zeros.

5.20.2.5 Output Slot 3: PCM Playback Left Channel

Output frame slot 3 is the composite digital audio left playback stream. Typically this slot is composed of standard PCM (.wav) output samples digitally mixed by the host processor. The Intel® 6300ESB ICH transmits sample streams of 16 bits or 20 bits and stuffs remaining bits with zeros.

Data in output slots 3 and 4 from the Intel® 6300ESB ICH should be duplicated by software when there is only a single channel out.

5.20.2.6 Output Slot 4: PCM Playback Right Channel

Output frame slot 4 is the composite digital audio right playback stream. Typically this slot is composed of standard PCM (.wav) output samples digitally mixed by the host processor. The Intel® 6300ESB ICH transmits sample streams of 16 or 20 bits and stuffs remaining bits with zeros.

Data in output slots 3 and 4 from the Intel® 6300ESB ICH should be duplicated by software when there is only a single channel out.

5.20.2.7 Output Slot 5: Modem Codec

Output frame slot 5 contains modem DAC data.

The modem DAC output supports 16-bit resolution. At boot time, when the modem codec is supported, the AC'97 controller driver determines the DAC resolution. During normal runtime operation the Intel® 6300ESB ICH stuffs trailing bit positions within this time slot with zeros.

5.20.2.8 Output Slot 6: PCM Playback Center Front Channel

When set up for 6-channel mode, this slot is used for the front center channel. The format is the same as Slots 3 and 4. When not set up for 6-channel mode, this channel will always be stuffed with zeros by Intel® 6300ESB ICH.

5.20.2.9 Output Slots 7-8: PCM Playback Left and Right Rear Channels

When set up for 4 or 6 channel modes, slots 7 and 8 are used for the rear Left and Right channels. The format for these two channels are the same as Slots 3 and 4.

5.20.2.10 Output Slot 9: Playback Sub Woofer Channel

When set for 6-channel mode, this slot is used for the Sub Woofer. The format is the same as Slot three. When not set up for 6-channel mode, this channel will always be stuffed with zeros by Intel® 6300ESB ICH.

5.20.2.11 Output Slots 10-11: Reserved

Output frame slots 10-11 are reserved and are always stuffed with 0s by the Intel® 6300ESB ICH AC'97 controller.

5.20.2.12 Output Slot 12: I/O Control

Sixteen bits of DAA and GPIO control (output) and status (input) have been directly assigned to bits on slot 12 in order to minimize latency of access to changing conditions.

The value of the bits in this slot are the values written to the GPIO control register at offset 54h and D4h (in the case of a secondary codec) in the modem codec I/O space. The following rules govern the usage of slot 12.

1. Slot 12 is marked invalid by default on coming out of AC-link reset, and will remain invalid until a register write to 54h/D4h.
2. A write to offset 54h/D4h in codec I/O space will cause the write data to be transmitted on slot 12 in the next frame, with slot 12 marked valid, and the address/data information to also be transmitted on slots 1 and 2.
3. After the first write to offset 54h/D4h, slot 12 remains valid for all following frames. The data transmitted on slot 12 is the data last written to offset 54h/D4h. Any subsequent write to the register will cause the new data to be sent out on the next frame.
4. Slot 12 will get invalidated after the following events:
   — PCI reset, AC'97 cold reset, warm reset, and hence a wake from S3, S4, or S5.
   — Slot 12 will remain invalid until the next write to offset 54h/D4h.

5.20.2.13 AC-Link Input Frame (SDIN)

There are three AC_SDIN lines on the Intel® 6300ESB ICH for use with up to three codecs. Each AC_SDIN pin may have a codec attached. The input frame data streams correspond to the multiplexed bundles of all digital input data targeting the AC'97 controller. As in the case for the output frame, each AC-link input frame consists of twelve time slots.

A new audio input frame begins with a low to high transition of AC_SYNC. AC_SYNC is synchronous to the rising edge of BIT_CLK. On the immediately following falling edge of BIT_CLK, the receiver samples the assertion of AC_SYNC. This falling edge marks the time when both sides of AC-link are aware of the start of a new audio frame. On the next rising edge of BIT_CLK, the codec transitions AC_SDIN into the first bit position of slot 0 (codec ready bit). Each new bit position is presented to AC-link on a rising edge of BIT_CLK, and subsequently sampled by the Intel® 6300ESB ICH on the following falling edge of BIT_CLK. This sequence ensures that data transitions and subsequent sample points for both incoming and outgoing data streams are time aligned.

SDIN data stream must follow the AC'97 specification and be MSB justified with all non-valid bit positions (for assigned and/or unassigned time slots) stuffed with zeros. AC_SDIN data is sampled by the Intel® 6300ESB ICH on the falling edge of BIT_CLK.

5.20.2.14 Input Slot 0: Tag Phase

Input slot 0 consists of a codec ready bit (bit 15), and slot valid bits for each subsequent slot in the frame (bits [14:3]).

The codec ready bit within slot 0 (bit 15) indicates whether the codec on the AC-link is ready for register access (digital domain). When the codec ready bit in slot 0 is a zero, the codec is not ready for register access. When the AC-link codec ready bit is a 1, it indicates that the AC-link and codec control and status registers are in a fully operational state. The codec ready bits are visible through the Global Status register of the Intel® 6300ESB ICH. Software must further probe the Powerdown Control/Status register in the codec to determine exactly which subsections, when any, are ready.

Bits [14:3] in slot 0 indicate which slots of the input stream to the Intel® 6300ESB ICH contain valid data, just as in the output frame. The remaining bits in this slot are stuffed with zeros.

5.20.2.15 Input Slot 1: Status Address Port/Slot Request Bits

The status port is used to monitor status of codec functions including, but not limited to, mixer settings and power management.

Slot 1 must echo the control register index, for historical reference, for the data to be returned in slot 2, assuming that slots 1 and 2 had been tagged valid by the codec in slot 0.

For variable sample rate output, the codec examines its sample rate control registers, the state of its FIFOs, and the incoming SDOUT tag bits at the beginning of each audio output frame to determine which SLOTREQ bits to set active (low). SLOTREQ bits asserted during the current audio input frame signal which output slots require data from the controller in the next audio output frame. For fixed 48 KHz operation the SLOTREQ bits are always set active (low) and a sample is transferred each frame.

For variable sample rate input, the tag bit for each input slot indicates whether valid data is present or not.

Table 139. Input Slot 1 Bit Definitions

Bit	Description
19	Reserved (Set to zero)
18:12	Control Register Index (Stuffed with zeros when tagged as invalid)
11	Slot 3 Request: PCM Left Channel(1)
10	Slot 4 Request: PCM Right Channel(1)
9	Slot 5 Request: Modem Line 1
[8:2}	Slot 6-12 Request: Not Implemented
1:0	Reserved (Stuffed with zeros)

NOTE: Slot 3 Request and Slot 4 Request bits must be the same value, i.e. set or cleared in tandem. This is also true for the Slot 7 and Slot 8 Request bits, as well as the Slot 6 and Slot 9 Request bits.

As shown in Table 139, slot 1 delivers codec control register read address and multiple sample rate slot request flags for all output slots of the controller. When a slot request bit is set by the codec, the controller will return data in that slot in the next output frame. Slot request bits for slots 3 and 4 are always set or cleared in tandem, i.e. both are set or cleared.

When set, the input slot 1 tag bit only pertains to Status Address Port data from a previous read. SLOTREQ bits are always valid independent of the slot 1 tag bit.

5.20.2.16 Input Slot 2: Status Data Port

The status data port receives 16-bit control register read data.

Bit [19:4]: Control Register Read Data

Bit [3:0]: Reserved.

5.20.2.17 Input Slot 3: PCM Record Left Channel

Input slot 3 is the left channel input of the codec. The Intel® 6300ESB ICH supports 16-bit sample resolution. Samples transmitted to the Intel® 6300ESB ICH must be in left/right channel order.

5.20.2.18 Input Slot 4: PCM Record Right Channel

Input slot 4 is the right channel input of the codec. The Intel® 6300ESB ICH supports 16-bit sample resolution. Samples transmitted to the Intel® 6300ESB ICH must be in left/right channel order.

5.20.2.19 Input Slot 5: Modem Line

Input slot 5 contains MSB justified modem data. The Intel® 6300ESB ICH supports 16-bit sample resolution.

5.20.2.20 Input Slot 6: Optional Dedicated Microphone Record Data

Input slot 6 is a third PCM system input channel available for dedicated use by a microphone. This input channel supplements a true stereo output which enables more precise echo cancellation algorithm for speakerphone applications. The Intel® 6300ESB ICH supports 16-bit resolution for slot 6 input.

5.20.2.21 Input Slots 7-11: Reserved

Input frame slots 7-11 are reserved for future use and should be stuffed with zeros by the codec, per the AC'97 specification.

5.20.2.22 Input Slot 12: I/O Status

The status of the GPIOs configured as inputs are to be returned on this slot in every frame. The data returned on the latest frame is accessible to software by reading the register at offset 54h/D4h in the codec I/O space. Only the 16 MSBs are used to return GPI status. In order for GPI events to cause an interrupt, both the 'sticky' and 'interrupt' bits must be set for that particular GPIO pin in regs 50h and 52h. Therefore, the interrupt will be signalled until it has been cleared by the controller, which may be much longer than one frame.

Reads from 54h/D4h will not be transmitted across the link in slot 1 and 2. The data from the most recent slot 12 is returned on reads from offset 54h/D4h.

5.20.2.23 Register Access

In the Intel® 6300ESB ICH implementation of the AC-link, up to three codecs may be connected to the SDOUT pin. The following mechanism is used to address the primary, secondary, and tertiary codecs individually.

The primary device uses bit 19 of slot 1 as the direction bit to specify read or write. Bits [18:12] of slot 1 are used for the register index. For I/O writes to the primary codec, the valid bits [14:13] for slots 1 and 2 must be set in slot 0, as shown in Table 140. Slot 1 is used to transmit the register address, and slot 2 is used to transmit data. For I/O reads to the primary codec, only slot 1 should be valid since only an address is transmitted. For I/O reads only slot 1 valid bit is set, while for I/O writes both slots 1 and 2 valid bits are set.

The secondary and tertiary codec registers are accessed using slots 1 and 2 as described above, however the slot valid bits for slots 1 and 2 are marked invalid in slot 0 and the codec ID bits [1:0] (bit 0 and bit 1 of slot 0) is set to a non zero value. This allows the secondary or tertiary codec to monitor the slot valid bits of slots 1 and 2, and bits [1:0] of slot 0 to determine when the access is directed to the secondary or tertiary codec. When the register access is targeted to the secondary or tertiary codec, slot 1 and 2 will contain the address and data for the register access. Since slots 1 and 2 are marked invalid, the primary codec will ignore these accesses.

Table 140. Output Tag Slot 0

Bit	Primary Access Example	Secondary Access Example	Description
15	1	1	Frame Valid
14	1	0	Slot 1 Valid, Command Address bit (Primary codec only)
13	1	0	Slot 2 Valid, Command Data bit (Primary codec only)
12:3	X	X	Slot 3-12 Valid
2	0	0	Reserved
1:0	00	01	Codec ID (00 reserved for primary; 01 indicate secondary; 10 indicate tertiary).

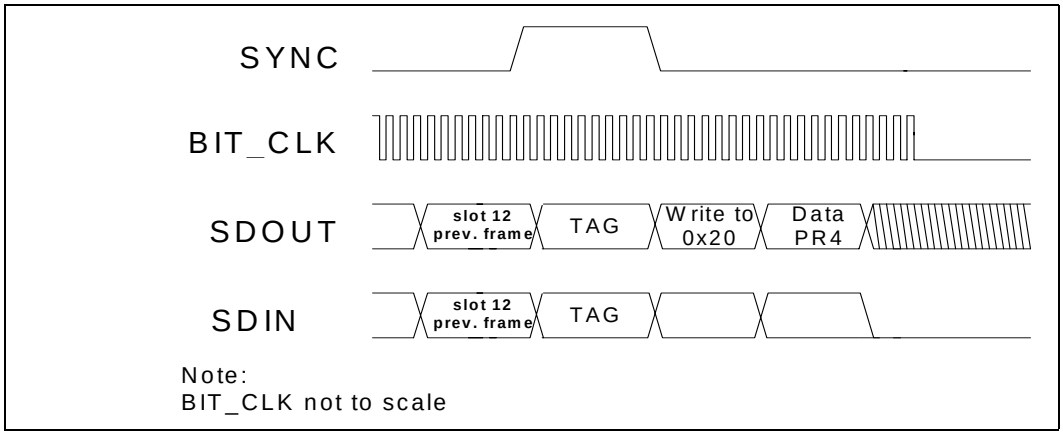
When accessing the codec registers, only one I/O cycle may be pending across the AC-link at any time. The Intel® 6300ESB ICH implements write posting on I/O writes across the AC-link (i.e., writes across the link are indicated as complete before they are actually sent across the link). In order to prevent a second I/O write from occurring before the first one is complete, software must monitor the CAS bit in the Codec Access Semaphore register which indicates that a codec access is pending. Once the CAS bit is cleared, then another codec access (read or write) may go through. The exception to this being reads to offset 54h/D4h/154h (slot 12) which are returned immediately with the most recently received slot 12 data. Writes to offset 54h, D4h, and 154h (primary, secondary and tertiary codecs), get transmitted across the AC-link in slots 1 and 2 as a normal register access. Slot 12 is also updated immediately to reflect the data being written.

The controller will 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.

5.20.3 AC-Link Low Power Mode

The AC-link signals may be placed in a low-power mode. When the AC'97 Powerdown Register (26h), is programmed to the appropriate value, both BIT_CLK and AC_SDIN will be brought to, and held at a logic low voltage level.

Figure 26. AC-Link Powerdown Timing



BIT_CLK and AC_SDIN transition low immediately, within the maximum specified time, after a write to the Powerdown Register (26h) with PR4 enabled. When the AC'97 controller driver is at the point where it is ready to program the AC-link into its low-power mode, slots 1 and 2 are assumed to be the only valid stream in the audio output frame. At this point in time it is assumed that all sources of audio input have been neutralized.

The AC'97 controller also drives AC_SYNC, and SDOUT low after programming AC'97 to this low power, halted mode.

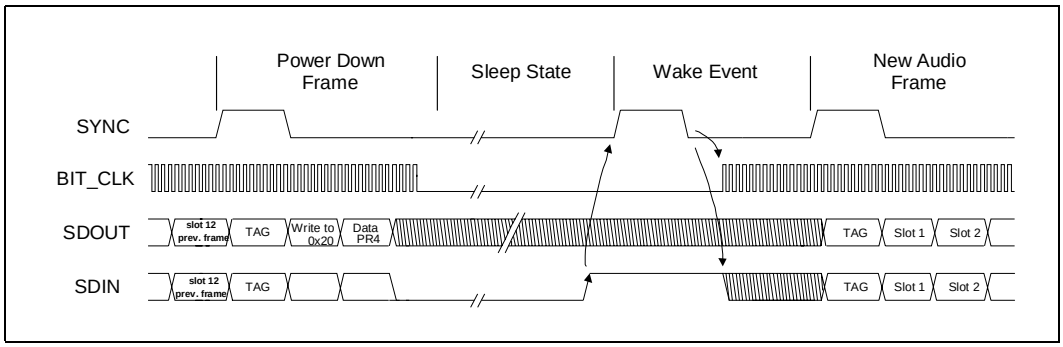
Once the codec has been instructed to halt BIT_CLK, a special wake up protocol must be used to bring the AC-link to the active mode since normal output and input frames cannot be communicated in the absence of BIT_CLK. Once in a low-power mode, the Intel® 6300ESB ICH provides three methods for waking up the AC-link; external wake event, cold reset and warm reset.

Note: Before entering any low-power mode where the link interface to the codec is expected to be powered down while the rest of the system is awake, the software must set the "Shut Off" bit in the control register to ensure that the Intel® 6300ESB ICH controller does not drive the output pins of the link.

5.20.3.1 External Wake Event

Codecs may signal the controller to wake the AC-link, and wake the system using AC_SDIN.

Figure 27. SDIN Wake Signaling



The minimum AC_SDIN wake up pulse width is 1 us. The rising edge of SDATA_IN (SDATA_IN(0) or SDATA)IN(1) for split partitioned implementation) causes the audio controller to sequence through its AC-link “warm reset” and signal PME# to the system’s ACPI controller. The primary codec must wait to sample AC_SYNC high and low before restarting BIT_CLK as diagrammed in Figure 6-24. The codec that signaled the wake event must keep its AC_SDIN high until it has sampled AC_SYNC having gone high, and then low.

The AC-link protocol provides for a cold reset and a warm reset. The type of reset used depends on the system’s current power down state. Unless a cold or register reset (a write to the Reset register in the codec) is performed, wherein the AC‘97 codec registers are initialized to their default values, registers are required to keep state during all power down modes.

Once powered down, activation of the AC-link through re-assertion of the AC_SYNC signal must not occur for a minimum of four audio frame times following the frame in which the power down was triggered. When AC-link powers up, it indicates readiness via the codec ready bit.

5.20.4 AC‘97 Cold Reset

A cold reset is achieved by asserting AC_RST# for 1 us. By driving AC_RST# low, BIT_CLK, and SDOUT will be activated and all codec registers will be initialized to their default power on reset values.

AC_RST# is an asynchronous AC‘97 input to the codec.

5.20.5 AC‘97 Warm Reset

A warm reset will re-activate the AC-link without altering the current codec register values. A warm reset is signaled by driving AC_SYNC high for a minimum of 1us in the absence of BIT_CLK.

Within normal frames, AC_SYNC is a synchronous AC‘97 input to the codec. However, in the absence of BIT_CLK, AC_SYNC is treated as an asynchronous input to the codec used in the generation of a warm reset.

The codec must not respond with the activation of BIT_CLK until AC_SYNC has been sampled low again by the codec. This will prevent the false detection of a new frame.

Note: On receipt of wake up signalling from the codec, the digital controller will issue an interrupt when enabled. Software will then have to issue a warm or cold reset to the codec by setting the appropriate bit in the Global Control Register.

5.20.6 System Reset

Table 141 indicates the states of the link during various system reset and sleep conditions.

Table 141. AC-link State during PXPCIRST#

Signal	Power Plane	I/O	During PXPCIRST#/	After PXPCIRST#/	S1	S3	S4/S5
AC_RST#	Resume[3]	Output	Low	Low	Cold Reset bit (Hub Interface)	Low	Low
AC_SDOUT	Core[1]	Output	Low	Running	Low	Low	Low
AC_SYNC	Core	Output	Low	Running	Low	Low	Low
BIT_CLK	Core	Input	Driven by codec	Running	Low[2,4]	Low[2,4]	Low[2,4]
AC_SDIN[2:0]	Resume	Input	Driven by codec	Running	Low[2,4]	Low[2,4]	Low[2,4]

NOTES:
1. Intel® 6300ESB ICH core well outputs are used as strapping options for the Intel® 6300ESB ICH, sampled during system reset. These signals may have weak pullups/pulldowns on them. The Intel® 6300ESB ICH outputs will be driven to the appropriate level prior to AC_RST# being deasserted, preventing a codec from entering test mode. Straps are tied to the core well to prevent leakage during a suspend state.
2. The pull-down resistors on these signals are only enabled when the AC-Link Shut Off bit in the AC'97 Global Control Register is set to 1. All other times, the pull-down resistor is disabled.
3. AC_RST# will be held low during S3-S5. It cannot be programmed high during a suspend state.
4. BIT_CLK and AC_SDIN[2:0] are driven low by the codecs during normal states. When the codec is powered during suspend states, it will hold these signals low. However, when the codec is not present, or not powered in suspend, external pull-down resistors are required.

The transition of AC_RST# to the deasserted state will only occur under driver control. In the S1 sleep state, the state of the AC_RST# signal is controlled by the AC'97 Cold Reset# bit (bit 1) in the Global Control register. AC_RST# will be asserted (low) by the Intel® 6300ESB ICH under the following conditions:

- RSMRST# (system reset, including the reset of the resume well and PXPCIRST#)
- Mechanical power up (causes PXPCIRST#)
- Write to CF9h hard reset (causes PXPCIRST#)
- Transition to S3/S4/S5 sleep states (causes PXPCIRST#)
- Write to AC'97 Cold Reset# bit in the Global Control Register.

Hardware will never deassert AC_RST# (i.e., never deasserts the Cold Reset# bit) automatically. Only software may deassert the Cold Reset# bit, and hence the AC_RST# signal. This bit, while it resides in the core well, will remain cleared upon return from S3/S4/S5 sleep states. The AC_RST# pin will remain actively driven from the resume well as indicated.

5.20.7 Hardware Assist to Determine AC_SDIN Used Per Codec

Software first performs a read to one of the audio codecs. The read request goes out on AC_SDOUT. Since under our micro-architecture only one read may be performed at a time on the link, eventually the read data will come back on one of the AC_SDIN[2:0] lines.

The codec will do this by indicating that status data is valid in its TAG, then echo the read address in slot 1 followed by the read data in slot 2.

The new function of the Intel® 6300ESB ICH hardware is to notice which AC_SDIN line contains the read return data, and to set new bits in the new register indicating which AC_SDIN line the register read data returned on. When it returned on AC_SDIN0, bits [1:0] contain the value '00'. When it returned on AC_SDIN1, the bits contain the value '01', etc.

Intel® 6300ESB ICH hardware may set these bits every time register read data is returned from a function 5 read. No special command is necessary to cause the bits to be set. The new driver/BIOS software will read the bits from this register when it cares to, and may ignore it otherwise. When software is attempting to establish the codec-to-AC_SDIN mapping, it will single feed the read request and not pipeline to ensure it gets the right mapping, hardware cannot ensure the serialization of the access.

5.20.8 Software Mapping of AC_SDIN to DMA Engine

Once software has performed the register read to determine codec-to-AC_SDIN mapping, it will then either set bits [5:4] or [7:6] in the SDATA_IN MAP register to map this codec to the DMA engine. After it maps the audio codecs, it will set the "SE" (steer enable) bit, which now lets the hardware know to no longer OR the AC_SDIN lines, and to use the mappings in the register to steer the appropriate AC_SDIN line to the correct DMA engines.

Register and Memory Mapping 6

The Intel® 6300ESB ICH contains registers that are located in the processor's I/O space, memory space and sets of PCI configuration registers that are located in PCI configuration space. This chapter details the Intel® 6300ESB ICH I/O and memory maps. Register access is also described.
Register-level address maps and Individual register bit descriptions are provided in the following chapters. The following notations are used in the chapters that follow.

RO	**Read Only:** Writes to this register location generally have no effect. However, in some cases, two separate registers are located at the same location where a read will access one register and a write will access the other register. See the I/O and memory map tables for details.
WO	**Write Only:** Reads to this register location generally have no effect. However, in some cases, two separate registers are located at the same location where a read will access one register and a write will access the other register. See the I/O and memory map tables for details.
R / W	**Read / Write:** A register with this attribute may be read and written.
R / WC	**Read / Write Clear:** A register bit with this attribute may be read and written. However, writing a 1 will clear (sets to zero) the corresponding bit, and writing a 0 will have no effect.
Default	When coming out of reset, the registers are set to predetermined default states. It is the responsibility of the system initialization software to determine configuration, operating parameters, and optional system features that are applicable, and to program the Intel® 6300ESB ICH registers accordingly.

6.1 PCI Devices and Functions

The Intel® 6300ESB ICH incorporates multiple PCI functions as shown in Table 142. These functions are divided into four PCI devices. The first is the Hub Interface Link-To-PCI bridge, D: 30 F:0. The second device, D31:F1, contains most of the standard PCI functions present in most ICHs, as well as some new related Intel® 6300ESB ICH features; SATA and SMBus Controller. The third device, D29 Fx, is the USB host controller device which includes new features specific to the Intel® 6300ESB ICH; Watchdog Timer and an additional IOxAPIC. The fourth PCI device, D28:F0, is also a new Intel® 6300ESB ICH feature; a Hub Interface-to-PCI-X bridge.

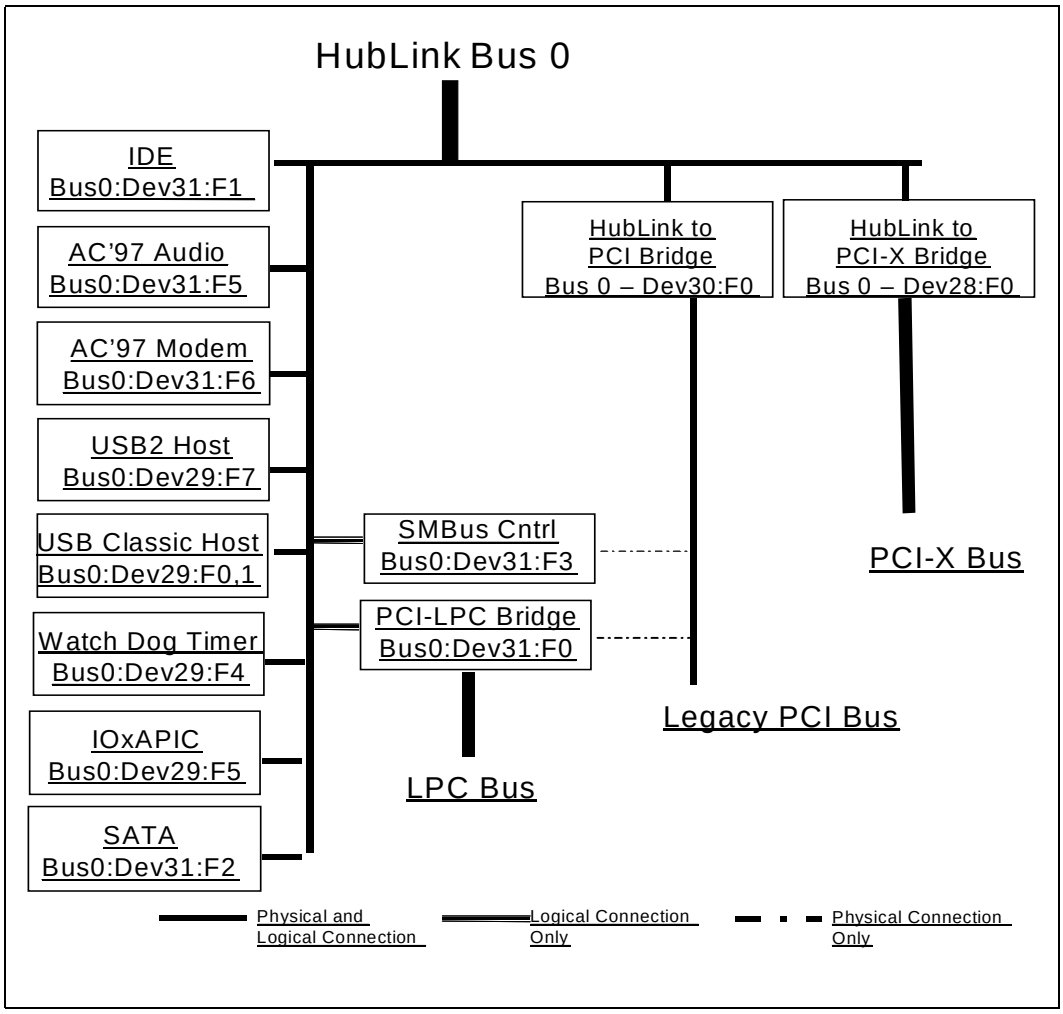
When a particular system does not want to support any one of Device 31's, 29's or 28's functions, they may individually be disabled. When a function is disabled, it does not appear at all to the software. A disabled function will not respond to any register reads or writes.

Table 142. PCI Devices and Functions

Bus:Device:Function	Function Description
Bus 0: Device 30: Function 0	Hub Interface to PCI Bridge
Bus 0: Device 31: Function 0	PCI to LPC Bridge
Bus 0: Device 31: Function 1	IDE Controller
Bus 0: Device 31: Function 2	SATA Controller
Bus 0: Device 31: Function 3	SMBus Controller
Bus 0: Device 31: Function 5	AC'97 Audio Controller
Bus 0: Device 31: Function 6	AC'97 Modem Controller
Bus 0: Device 29: Function 0	USB Controller #1
Bus 0: Device 29: Function 1	USB Controller #2
Bus 0: Device 29: Function 4	New: Watchdog Timer
Bus 0: Device 29: Function 5	New: IOxAPIC
Bus 0: Device 29: Function 7	USB 2.0 Controller
Bus 0: Device 28: Function 0	New: Hub Interface to PCI-X Bridge

NOTE: The PCI to LPC bridge contains registers that control LPC, Power Management, System Management, GPIO, CPU Interface, RTC, Interrupts, Timers, DMA.

Figure 28. Intel® 6300ESB ICH Device Diagram



6.2 PCI Configuration Map

Each PCI function on the Intel® 6300ESB ICH has a set of PCI configuration registers. The register map tables for each function are included at the beginning of each respective chapter.

Configuration Space registers are accessed through configuration cycles on the PCI bus by the Host bridge using configuration mechanism #1 detailed in the *PCI Local Bus Specification,* Revision 2.2.

Some PCI registers contain "Reserved" bits. Software must deal correctly with fields that are reserved. On reads, software must use appropriate masks to extract the defined bits and not rely on reserved bits being any particular value. On writes, software must ensure that the values of reserved bit positions are preserved. That is, the values of reserved bit positions must first be read, merged with the new values for other bit positions and then written back. Note the software does not need to perform read, merge, write operation for the Configuration Address Register (0xCF8h).

In addition to reserved bits within a register, the configuration space contains reserved locations. Software should not write to reserved PCI configuration locations in the device-specific region (above address offset 3Fh).

6.3 I/O Map

The I/O map is divided into Fixed and Variable address ranges. Fixed ranges cannot be moved, but in some cases may be disabled. Variable ranges may be moved and may also be disabled.

6.3.1 Fixed I/O Address Ranges

Table 143 shows the Fixed I/O decode ranges from the CPU perspective. Note that for each I/O range, there may be separate behavior for reads and writes. The Hub Interface cycles that go to target ranges that are marked as “Reserved” will not be decoded by the Intel® 6300ESB ICH, and will be passed to PCI. When a PCI master targets one of the fixed I/O target ranges, it will be positively decoded by the Intel® 6300ESB ICH in medium speed.

Note: Unclaimed PCI cycles will be subtractively decoded and forwarded to the LPC.

Address ranges that are not listed or marked “Reserved” are NOT decoded by the Intel® 6300ESB ICH (unless assigned to one of the variable ranges).

Table 143. Fixed I/O Ranges Decoded by Intel® 6300ESB I/O Controller Hub (Sheet 1 of 3)

I/O Address	Read Target	Write Target	Internal Unit
00h - 08h	DMA Controller	DMA Controller	DMA
09h - 0Eh	Reserved	DMA Controller	DMA
0Fh	DMA Controller	DMA Controller	DMA
10h - 18h	DMA Controller	DMA Controller	DMA
19h - 1Eh	Reserved	DMA Controller	DMA
1Fh	DMA Controller	DMA Controller	DMA
20h - 21h	Interrupt Controller	Interrupt Controller	Interrupt
24h - 25h	Interrupt Controller	Interrupt Controller	Interrupt
28h - 29h	Interrupt Controller	Interrupt Controller	Interrupt
2Ch - 2Dh	Interrupt Controller	Interrupt Controller	Interrupt
2E-2F	LPC SIO	LPC SIO	Forwarded to LPC
30h - 31h	Interrupt Controller	Interrupt Controller	Interrupt
34h - 35h	Interrupt Controller	Interrupt Controller	Interrupt

NOTES:
1. Only when the Port 61 Alias Enable bit (Device 31:Function 0, Offset D0, Bit 4) bit is set. Otherwise, the target is PCI.
2. Only when IDE Standard I/O space is enabled for Secondary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
3. Only when IDE Standard I/O space is enabled for Primary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
4. Should forward read cycles to this address to LPC Variable I/O Decode Ranges.

Table 143. Fixed I/O Ranges Decoded by Intel® 6300ESB I/O Controller Hub (Sheet 2 of 3)

I/O Address	Read Target	Write Target	Internal Unit
38h - 39h	Interrupt Controller	Interrupt Controller	Interrupt
3Ch - 3Dh	Interrupt Controller	Interrupt Controller	Interrupt
40h - 42h	Timer/Counter	Timer/Counter	PIT (8254)
43h	Reserved	Timer/Counter	PIT
4E-4F	LPC SIU	LPC SIU	Forwarded to LPC
50h - 52h	Timer/Counter	Timer/Counter	PIT
53h	Reserved	Timer/Counter	PIT
60h	Microcontroller/Emulation	Microcontroller/Emulation	Forwarded to LPC
61h	NMI Controller	NMI Controller	CPU I/F
62h	Microcontroller	Microcontroller	Forwarded to LPC
63h	NMI Controller	NMI Controller	CPU I/F
64h	Microcontroller/Emulation	Microcontroller/Emulation	Forwarded to LPC
65h	NMI Controller	NMI Controller	CPU I/F
66h	Microcontroller	Microcontroller	Forwarded to LPC
67h	NMI Controller	NMI Controller	CPU I/F
70h	Reserved	NMI and RTC Controller	RTC
71h	RTC Controller	RTC Controller	RTC
72h	RTC Controller	NMI and RTC Controller	RTC
73h	RTC Controller	RTC Controller	RTC
74h	RTC Controller	NMI and RTC Controller	RTC
75h	RTC Controller	RTC Controller	RTC
76h	RTC Controller	NMI and RTC Controller	RTC
77h	RTC Controller	RTC Controller	RTC
80h	DMA Controller	DMA Controller and LPC or PCI	DMA
81h - 83h	DMA Controller	DMA Controller	DMA
84h - 86h	DMA Controller	DMA Controller and LPC or PCI	DMA
87h	DMA Controller	DMA Controller	DMA
88h	DMA Controller	DMA Controller and LPC or PCI	DMA
89h - 8Bh	DMA Controller	DMA Controller	DMA

NOTES:
1. Only when the Port 61 Alias Enable bit (Device 31:Function 0, Offset D0, Bit 4) bit is set. Otherwise, the target is PCI.
2. Only when IDE Standard I/O space is enabled for Secondary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
3. Only when IDE Standard I/O space is enabled for Primary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
4. Should forward read cycles to this address to LPC Variable I/O Decode Ranges.

Table 143. Fixed I/O Ranges Decoded by Intel® 6300ESB I/O Controller Hub (Sheet 3 of 3)

I/O Address	Read Target	Write Target	Internal Unit
8Ch - 8Eh	DMA Controller	DMA Controller and LPC or PCI	DMA
08Fh	DMA Controller	DMA Controller	DMA
90h - 91h	DMA Controller	DMA Controller	DMA
92h	Reset Generator	Reset Generator	CPU I/F
93h - 9Fh	DMA Controller	DMA Controller	DMA
A0h - A1h	Interrupt Controller	Interrupt Controller	Interrupt
A4h - A5h	Interrupt Controller	Interrupt Controller	Interrupt
A8h - A9h	Interrupt Controller	Interrupt Controller	Interrupt
ACh - ADh	Interrupt Controller	Interrupt Controller	Interrupt
B0h - B1h	Interrupt Controller	Interrupt Controller	Interrupt
B2h - B3h	Power Management	Power Management	Power Management
B4h - B5h	Interrupt Controller	Interrupt Controller	Interrupt
B8h - B9h	Interrupt Controller	Interrupt Controller	Interrupt
BCh - BDh	Interrupt Controller	Interrupt Controller	Interrupt
C0h - D1h	DMA Controller	DMA Controller	DMA
D2h - DDh	Reserved	DMA Controller	DMA
DEh - DFh	DMA Controller	DMA Controller	DMA
F0h	See Note 2	FERR#/IGNNE# / Interrupt Controller	CPU I/F
170h - 177h	IDE Controller[2]	IDE Controller[2]	Forwarded to IDE
1F0h - 1F7h	IDE Controller[1]	IDE Controller[1]	Forwarded to IDE
376h	IDE Controller[2]	IDE Controller[2]	Forwarded to IDE
200-207h	Gameport Low	Gameport Low	Forwarded to LPC
208-20Fh	Gameport High	Gameport High	Forwarded to LPC
376h	IDE Controller[2]	IDE Controller[2]	Forwarded to IDE
388-38Bh	AdLib	AdLib	Forwarded to LPC
3F6h	IDE Controller[1]	IDE Controller[1]	Forwarded IDE
4D0h - 4D1h	Interrupt Controller	Interrupt Controller	Interrupt
CF9h	Reset Generator	Reset Generator	CPU I/F

NOTES:
1. Only when the Port 61 Alias Enable bit (Device 31:Function 0, Offset D0, Bit 4) bit is set. Otherwise, the target is PCI.
2. Only when IDE Standard I/O space is enabled for Secondary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
3. Only when IDE Standard I/O space is enabled for Primary Channel and the IDE Controller is in legacy mode. Otherwise, the target is PCI.
4. Should forward read cycles to this address to LPC Variable I/O Decode Ranges.

6.3.2 Variable I/O Decode Ranges

Table 144 shows the Variable I/O Decode Ranges. They are set using Base Address Registers (BARs) or other configuration bits in the various PCI configuration spaces. The PNP software (PCI or ACPI) may use their configuration mechanisms to set and adjust these values.

When a cycle is detected on the Hub Interface, the Intel® 6300ESB ICH will positively decode the cycle. When the response is on the behalf of an LPC device, the Intel® 6300ESB ICH will forward the cycle to the LPC.

Refer to Table 144 for a complete list of all variable I/O registers.

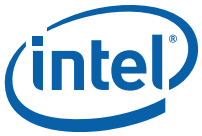
Warning: The Variable I/O Ranges should not be set to conflict with the Fixed I/O Ranges. There may be unpredictable results when the configuration software allows conflicts to occur. The Intel® 6300ESB ICH does not perform any checks for conflicts.

Table 144. Variable I/O Decode Ranges

Range Name	Mappable	Size (Bytes)	Target
ACPI	Anywhere in 64K I/O Space	64	Power Management
IDE Bus Master	Anywhere in 64K I/O Space	16	IDE Unit
USB #1	Anywhere in 64K I/O Space	32	USB 1.0 Host Controller 1
SMBus	Anywhere in 64K I/O Space	32	SMB Unit
AC'97 Audio Mixer	Anywhere in 64K I/O Space	256	AC'97 Unit
AC'97 Bus Master	Anywhere in 64K I/O Space	64	AC'97 Unit
AC'97 Modem Mixer	Anywhere in 64K I/O Space	256	AC'97 Unit
TCO	96 Bytes above ACPI Base	32	TCO Unit
GPIO	Anywhere in 64K I/O Space	64	GPIO Unit
Parallel Port	3 ranges in 64K I/O Space	8	LPC Peripheral
Serial Port 1	8 Ranges in 64K I/O Space	8	LPC Peripheral
Serial Port 2	8 Ranges in 64K I/O Space	8	LPC Peripheral
Floppy Disk Controller	2 Ranges in 64K I/O Space	8	LPC Peripheral
MIDI	4 Ranges in 64K I/O Space	2	LPC Peripheral
MSS	4 Ranges in 64K I/O Space	8	LPC Peripheral
SoundBlaster	2 Ranges in 64K I/O Space	32	LPC Peripheral

Table 144. Variable I/O Decode Ranges

USB #2	Anywhere in 64K I/O Space	32	USB 1.0 Host Controller 2
LPC Generic 1	Anywhere in 64K I/O Space	128	LPC Peripheral
LPC Generic 2	Anywhere in 64K I/O Space	16	LPC Peripheral
Monitors 4:7	Anywhere in 64K I/O Space	16	LPC Peripheral or Trap on PCI
Native IDE Primary Command	Anywhere in 64K I/O Space	8	IDE Unit
Native IDE Primary Control	Anywhere in 64K I/O Space	4	IDE Unit
Native IDE Secondary Command	Anywhere in 64K I/O Space	8	IDE Unit
Native IDE Secondary Control	Anywhere in 64K I/O Space	4	IDE Unit



6.4 Memory Map

Table 145 shows, from the CPU perspective, the memory ranges that the Intel® 6300ESB ICH will decode. Cycles that arrive from the Hub Interface that are not directed to any of the internal memory targets that decode directly from Hub Interface will be driven out on PCI. The Intel® 6300ESB ICH may then claim the cycle for it to be forwarded to LPC or claimed by the internal
I/O APIC or subtractive decode the cycle. When subtractive decode is enabled, the subtractive decoded cycle may be forwarded to the LPC I/F or to the FWH.

PCI cycles generated by an external PCI master will be positively decoded unless it falls in the PCI-PCI bridge forwarding range (those addresses are reserved for PCI peer-to-peer traffic). When the cycle is not in the I/O APIC or FWH/LPC ranges, it will be forwarded up the Hub Interface to the Host Controller. PCI masters cannot access the memory ranges for functions that decode directly from Hub Interface.

Table 145. Memory Decode Ranges from CPU Perspective (Sheet 1 of 2)

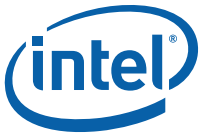
Memory Range	Target	Dependency/Comments
0000 0000 - 000D FFFF 0010 0000 - TOM (Top of Memory)	Main Memory	TOM registers in Host Controller
000E 0000 - 000F FFFF	FWH	Bit 7 in FWH Decode Enable Register is set
FEC0 0000 - FEC0 0043	I/O APIC inside the Intel® 6300ESB ICH	Downstream memory writes to FEC0 0020 are also decoded by D29:F5 APIC to support EOI.
FEC1 0000 - FEC1 0043	I/O APIC (D29:F5)	D29:F5 APIC also supports FEC00000-FEC00043 range of message signaled interrupts from the PCI-X interface (See Note 1)
FFC0 0000 - FFC7 FFFF FF80 0000 - FF87 FFFF	FWH	Bit 0 in FWH Decode Enable Register
FFC8 0000 - FFCF FFFF FF88 0000 - FF8F FFFF	FWH	Bit 1 in FWH Decode Enable Register
FFD0 0000 - FFD7 FFFF FF90 0000 - FF97 FFFF	FWH	Bit 2 in FWH Decode Enable Register is set
FFD8 0000 - FFDF FFFF FF98 0000 - FF9F FFFF	FWH	Bit 3 in FWH Decode Enable Register is set
FFE0 000 - FFE7 FFFF FFA0 0000 - FFA7 FFFF	FWH	Bit 4 in FWH Decode Enable Register is set
FFE8 0000 - FFEF FFFF FFA8 0000 - FFAF FFFF	FWH	Bit 5 in FWH Decode Enable Register is set
FFF0 0000 - FFF7 FFFF FFB0 0000 - FFB7 FFFF	FWH	Bit 6 in FWH Decode Enable Register is set.
FFF8 0000 - FFFF FFFF FFB8 0000 - FFBF FFFF	FWH	Always enabled. The top two 64K-byte blocks of this range may be swapped, as described in Section 6.4.1, "Boot-Block Update Scheme".

NOTES:
1. These ranges are decoded directly from Hub Interface. The memory cycles will not be seen on PCI.
2. Software must not attempt locks to memory mapped I/O ranges for USB EHCI, High Performance Event Timers, and IDE Expansion. When attempted, the lock is not honored, which means potential deadlock conditions may occur.

Table 145. Memory Decode Ranges from CPU Perspective (Sheet 2 of 2)

Memory Range	Target	Dependency/Comments
FF70 0000 - FF7F FFFF FF30 0000 - FF3F FFFF	FWH	Bit 3 in FWH Decode Enable 2 Register is set
FF60 0000 - FF6F FFFF FF20 0000 - FF2F FFFF	FWH	Bit 2 in FWH Decode Enable 2 Register is set
FF50 0000 - FF5F FFFF FF10 0000 - FF1F FFFF	FWH	Bit 1 in FWH Decode Enable 2 Register is set
FF40 0000 - FF4F FFFF FF00 0000 - FF0F FFFF	FWH	Bit 0 in FWH Decode Enable 2 Register is set
1 Kbyte anywhere in 4 Gbyte range	IDE Expansion[2]	Enable through standard PCI mechanism and bits in IDE I/O Configuration Register (Device 31, Function 1)
512B anywhere in 4 Gbyte range	AC'97 Host Controller (Mixer)[1]	Enable via standard PCI mechanism (Device 31, Function 5)
256B anywhere in 4 Gbyte range	AC'97 Host Controller (Bus Master)[1]	Enable via standard PCI mechanism (Device 31, Function 5)
1 Kbyte anywhere in 4 Gbyte range	USB EHCI Controller[1, 2]	Enable through standard PCI mechanism (Device 29, Function 7).
FED0 X000 - FED0 X3FF	Multimedia Timers[1, 2]	BIOS determines the "fixed" location which is one of four, 1-Kbyte ranges where X (in the first column) is 0h, 1h, 2h, or 3h.
1 Kbyte anywhere in 4 Gbyte range	SATA[1]	Enable via standard PCI mechanism (Device 31, Function 2)
1 Kbyte anywhere in 4 Gbyte range	WDT	Enable via standard PCI mechanism (Device 29, Function 4)
1 Mbyte to 4 Gbyte anywhere in 4 Gbyte range	PCI-X[1]	Enable via standard PCI mechanism (Device 28, Function 0)
All other	PCI	None/ If the address is below 16M, is not in one of the above BIOS Ranges, and positive decode is disabled; then the cycle will be forwarded to LPC as a standard LPC memory cycle. If the address is above 16M, if the cycle is not claimed by a device on PCI and neither by the Intel® 6300ESB ICH, then the cycle will Master-Abort on PCI

NOTES:
1. These ranges are decoded directly from Hub Interface. The memory cycles will not be seen on PCI.
2. Software must not attempt locks to memory mapped I/O ranges for USB EHCI, High Performance Event Timers, and IDE Expansion. When attempted, the lock is not honored, which means potential deadlock conditions may occur.



6.4.1 Boot-Block Update Scheme

The Intel® 6300ESB ICH supports a “top-block swap” mode that has the Intel® 6300ESB ICH swap the top block in the FWH (the boot block) with another location. This allows for safe update of the Boot Block (even if a power failure occurs). When the “TOP_SWAP” Enable bit is set, the Intel® 6300ESB ICH will invert A16 for cycles targeting FWH BIOS space. When this bit is zero, the Intel® 6300ESB ICH will not invert A16. This bit is automatically set to zero by RTCRST#, but not by PXPCIRST#.

The scheme is based on the concept that the top block is reserved as the “boot” block, and the block immediately below the top block is reserved for doing boot-block updates.

The algorithm is:

1. Software copies the top block to the block immediately below the top.
2. Software checks that the copied block is correct. This could be done by performing a checksum calculation.
3. Software sets the TOP_SWAP bit. This will invert A16 for cycles going to the FWH. Processor access to FFFF_0000 through FFFF_FFFF will be directed to FFFE_0000 through FFFE_FFFF in the FWH, and processor accesses to FFFE_0000 through FFFE_FFFF will be directed to FFFF_0000 through FFFF_FFFF.
4. Software erases the top block.
5. Software writes the new top block.
6. Software checks the new top block.
7. Software clears the TOP_SWAP bit.

If a power failure occurs at any point after step 3, the system will be able to boot from the copy of the boot block that is stored in the block below the top. This is because the TOP_SWAP bit is backed in the RTC well.

Note: The top-block swap mode may be forced by an external strapping option (See Section 3.21.1, “Functional Straps”). When top-block swap mode is forced in this manner, the TOP_SWAP bit cannot be cleared by software. A re-boot with the strap removed will be required to exit a forced top-block swap mode.

Note: Top-block swap mode only affects accesses to the FWH BIOS space, not feature space.

Note: The top-block swap mode has no effect on accesses below FFFE_0000.

November 2007
Order Number: 300641-004US

Hub Interface to PCI Bridge Registers (D30:F0) 7

The Hub Interface to PCI Bridge resides in PCI Device 30, Function 0 on Bus #0. This portion of the Intel® 6300ESB ICH implements the buffering and control logic between PCI and the Hub Interface. The arbitration for the PCI bus is handled by this PCI device. The PCI decoder in this device must decode the ranges for the Hub Interface. All register contents will be lost when core well power is removed.

7.1 PCI Configuration Registers (D30:F0)

Note: Registers that are not shown should be treated as Reserved (see Section 6.2, "PCI Configuration Map" for details).

Table 146. PCI Configuration Registers (D30:F0) (Sheet 1 of 2)

Offset	Mnemonic	Register Name/Function	Default	Type
00-01h	VID	Vendor ID	8086h	RO
02-03h	DID	Device ID	244Eh	RO
04-05h	CMD	PCI Device Command Register	0001h	R/W
06-07h	PD_STS	PCI Device Status Register	0080h	R/W
08h	RID	Revision ID	See NOTE:	RO
0Ah	SCC	Sub Class Code	04h	RO
0Bh	BCC	Base Class Code	06h	RO
0Dh	PMLT	Primary Master Latency Timer	00h	RO
0Eh	HEADTYP	Header Type	01h	RO
18h	PBUS_NUM	Primary Bus Number	00h	RO
19h	SBUS_NUM	Secondary Bus Number	00h	R/W
1Ah	SUB_BUS_NUM	Subordinate Bus Number	00h	R/W
1Bh	SMLT	Secondary Master Latency Timer	00h	R/W
1Ch	IOBASE	IO Base Register	F0h	R/W
1Dh	IOLIM	IO Limit Register	00h	R/W
1E-1Fh	SECSTS	Secondary Status Register	0280h	R/W
20-21h	MEMBASE	Memory Base	FFF0h	R/W
22-23h	MEMLIM	Memory Limit	0000h	R/W
24-25h	PREF_MEM_BASE	Prefetchable Memory Base	0000h	RO
26-27h	PREF_MEM_MLT	Prefetchable Memory Limit	0000h	RO
30-31h	IOBASE_HI	I/O Base Upper 16 Bits	0000h	RO
32-33h	IOLIMIT_HI	I/O Limit Upper 16 Bits	0000h	RO

NOTE: Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID register.

Table 146. PCI Configuration Registers (D30:F0) (Sheet 2 of 2)

Offset	Mnemonic	Register Name/Function	Default	Type
3Ch	INT_LINE	Interrupt Line	00h	RO
3E-3Fh	BRIDGE_CNT	Bridge Control	0000h	R/W
40-43h	HI1_CMD	Hub Interface 1 Command Control	20202802h	R/W
44-45h	DEVICE_HIDE	Secondary PCI Device Hiding Register	00	R/W
50-51h	CNF	Intel® 6300ESB ICH Configuration Register	1400h	R/W
70h	MTT	Multi-Transaction Timer	20h	R/W
82h	PCI_MAST_STS	PCI Master Status	00h	R/W
90h	ERR_CMD	Error Command Register	00h	R/W
92h	ERR_STS	Error Status Register	00h	R/W
F8h	MANID	Manufacturer's ID	0F66h	RO

NOTE: Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID register.

7.1.1 Offset 00 - 01h: VID—Vendor ID Register (HUB-PCI—D30:F0)

Table 147. Offset 00 - 01h: VID—Vendor ID Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	00-01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit

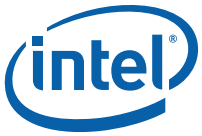
Bits	Name	Description	Access
15:0	Vendor Identification Number	This is a 16-bit value assigned to Intel. Intel VID = 8086h.	RO

7.1.2 Offset 02 - 03h: DID—Device ID Register (HUB-PCI—D30:F0)

Table 148. Offset 02 - 03h: DID—Device ID Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	02-03h	***Attribute:***	Read-Only
Default Value:	244Eh	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Device Identification Number	This is a 16-bit value assigned to the Intel® 6300ESB ICH Hub Interface to PCI bridge.	RO



7.1.3 Offset 04 - 05h: CMD—Command Register (HUB-PCI—D30:F0)

Table 149. Offset 04 - 05h: CMD—Command Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	04-05h	***Attribute:***	Read/Write
Default Value:	0001h	***Size:***	16-bit

Bits	Name	Description	Access
15:10	Reserved	Reserved.	
9	Fast Back to Back Enable (FBE)	Hardwired to'0'. The Intel® 6300ESB ICH does not support this capability.	RO
8	SERR# Enable (SERR_EN)	0 = Disable. 1 = Enable the Intel® 6300ESB ICH to generate an NMI (or SMI# if NMI routed to SMI#) when the D30:F0 SSE bit (offset 06h, bit 14) is set.	R/W
7	Wait Cycle Control	Hardwired to '0'	RO
6	Parity Error Response	0 = The Intel® 6300ESB ICH will ignore parity errors on the Hub Interface. 1 = The Intel® 6300ESB ICH is allowed to report parity errors detected on the Hub Interface. **NOTE:** The Hub Interface Parity Unsupported bit (D30:F0:40h:bit20) must be cleared for the PER bit to have any effect.	R/W
5	VGA Palette Snoop	Hardwired to '0'.	RO
4	Memory Write and Invalidate Enable (MWE)	Hardwired to '0'.	RO
3	Special Cycle Enable (SCE)	Hardwired to '0' by P2P Bridge spec.	RO
2	Bus Master Enable (BME)	0 = Disable 1 = Allows the Hub Interface-to-PCI bridge to accept cycles from PCI to run on the Hub Interface. **NOTE:** This bit does not affect the CF8h and CFCh I/O accesses. **NOTE:** Cycles that generated from the Intel® 6300ESB ICH's Device 31 functionality are not blocked by clearing this bit.	R/W
1	Memory Space Enable (MSE)	The Intel® 6300ESB ICH provides this bit as read/writable for software only. However, the Intel® 6300ESB ICH ignores the programming of this bit, and runs Hub Interface memory cycles to PCI.	R/W
0	I/O Space Enable (IOE)	The Intel® 6300ESB ICH provides this bit as read/writable for software only. However, the Intel® 6300ESB ICH ignores the programming of this bit and runs Hub Interface I/O cycles to PCI that are not intended for USB, IDE, or AC'97.	R/W

7.1.4 Offset 06 - 07h: PD_STS—Primary Device Status Register (HUB-PCI—D30:F0)

Note: For the writable bits in this register, writing a 1 will clear the bit. Writing a 0 to the bit will have no effect.

Table 150. Offset 06 - 07h: PD_STS—Primary Device Status Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	06-07h	***Attribute:***	Read/Write Clear
Default Value:	0080h	***Size:***	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	0 = Software clears this bit by writing a '1' to the bit location. 1 = Indicates that the Intel® 6300ESB ICH detected a parity error on the Hub Interface and the Hub Interface Parity Unsupported bit is cleared (D30:F0:40h:bit20). This bit gets set even when the Parity Error Response bit (offset 04, bit 6) is not set.	R/WC
14	Signaled System Error (SSE)	0 = Software clears this bit by writing a '1' to the bit location. 1 = An address, or command parity error, or special cycles data parity error has been detected on the PCI bus, and the Parity Error Response bit (D30:F0, Offset 04h, bit 6) is set. When this bit is set because of parity error and the D30:F0 SERR_EN bit (Offset 04h, bit 8) is also set, the Intel® 6300ESB ICH will generate an NMI (or SMI# if NMI routed to SMI#).	R/WC
13	Received Master Abort (RMA)	0 = Software clears this bit by writing a '1' to the bit location. 1 = The Intel® 6300ESB ICH received a master abort from the Hub Interface device.	R/WC
12	Received Target Abort (RTA)	0 = Software clears this bit by writing a '1' to the bit location. 1 = The Intel® 6300ESB ICH received a target abort from the Hub Interface device. The setting of this bit can be enabled to cause an internal SERR#.	R/WC
11	Signaled Target Abort (STA)	0 = Software clears this bit by writing a '1' to the bit location. 1 = The Intel® 6300ESB ICH signals a target abort condition on the Hub Interface.	R/WC
10:9	DEVSEL# Timing Status	00h = Fast timing. This register applies to the Hub Interface.	RO
8	Master Data Parity Error Detected (MDPD)	Since this register applies to the Hub Interface, the Intel® 6300ESB ICH must interpret this bit differently than it is in the PCI spec. 0 = Software clears this bit by writing a '1' to the bit location. 1 = The Intel® 6300ESB ICH detects a parity error on the Hub Interface and the Parity Error Response bit in the Command Register (offset 04h, bit 6) is set.	R/WC
7	Fast Back to Back	Hardwired to '1'.	RO
6	User Definable Features (UDF)	Hardwired to '0'.	RO
5	66 MHz Capable	Hardwired to '0'.	RO
4:0	Reserved	Reserved.	

7.1.5 Offset 08h: RID—Revision Identification Register (HUB-PCI—D30:F0)

Table 151. Offset 08h: RID—Revision Identification Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	08h	***Attribute:***	Read-Only
Default Value:	08h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	8-bit value that indicates the revision number for the Intel® 6300ESB ICH ICH Hub Interface-to-PCI bridge.	RO

7.1.6 Offset 0Ah: SCC—Sub-Class Code Register (HUB-PCI—D30:F0)

Table 152. Offset 0Ah: SCC—Sub-Class Code Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	04h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub-Class Code	8-bit value that indicates the category of bridge for the Intel® 6300ESB ICH Hub Interface to PCI bridge. The code is 04h indicating a PCI-to-PCI bridge.	RO

7.1.7 Offset 0Bh: BCC—Base-Class Code Register (HUB-PCI—D30:F0)

Table 153. Offset 0Bh: BCC—Base-Class Code Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	06h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	8-bit value that indicates the type of device for the Intel® 6300ESB ICH Hub Interface to PCI bridge. The code is 06h indicating a bridge device.	RO

7.1.8 Offset 0Dh: PMLT—Primary Master Latency Timer Register (HUB-PCI—D30:F0)

Note: This register does not apply to Hub Interface.

Table 154. Offset 0Dh: PMLT—Primary Master Latency Timer Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	0Dh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Master Latency Count	Not implemented.	
2:0	Reserved	Reserved.	

7.1.9 Offset 0Eh: HEADTYP—Header Type Register (HUB-PCI—D30:F0)

Table 155. Offset 0Eh: HEADTYP—Header Type Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	0Eh	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7	Multi-Function Device	This bit is ‘0’ to indicate a single function device.	RO
6:0	Header Type	8-bit field identifies the header layout of the configuration space, which is a PCI-to-PCI bridge in this case.	RO

7.1.10 Offset 18h: PBUS_NUM—Primary Bus Number Register (HUB-PCI—D30:F0)

Table 156. Offset 18h: PBUS_NUM—Primary Bus Number Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	18h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Primary Bus Number	This field indicates the bus number of the Hub Interface and is hardwired to 00h.	RO

7.1.11 Offset 19h: SBUS_NUM—Secondary Bus Number Register (HUB-PCI—D30:F0)

Table 157. Offset 19h: SBUS_NUM—Secondary Bus Number Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	19h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Secondary Bus Number	This field indicates the bus number of PCI. When this number is equal to the primary bus number (i.e., bus #0), the Intel® 6300ESB ICH will run Hub Interface configuration cycles to this bus number as Type 1 configuration cycles on PCI.	R/W

7.1.12 Offset 1A: SUB_BUS_NUM—Subordinate Bus Number Register (HUB-PCI—D30:F0)

Table 158. Offset 1A: SUB_BUS_NUM—Subordinate Bus Number Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	1A	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Subordinate Bus Number	This field specifies the highest PCI bus number below the Hub Interface to PCI bridge. When a Type 1 configuration cycle from the Hub Interface does not fall in the Secondary-to-Subordinate Bus ranges of Device 30, the Intel® 6300ESB ICH will indicate a master abort back to the Hub Interface.	R/W

7.1.13 Offset 1Bh: SMLT—Secondary Master Latency Timer Register (HUB-PCI—D30:F0)

This Master Latency Timer (MLT) controls the amount of time that the Intel® 6300ESB ICH will continue to burst data as a master on the PCI bus. When the Intel® 6300ESB ICH starts the cycle after being granted the bus, the counter is loaded and starts counting down from the assertion of FRAME#. When the internal grant to this device is removed, the expiration of the MLT counter will result in the deassertion of FRAME#. When the internal grant has not been removed, the Intel® 6300ESB ICH may continue to own the bus.

When the Secondary Master Latency Timer in Device 30 (offset 1Bh) is programmed to 00h (the default value), the North PCI initiator logic operates as though the timer never expires. Therefore, with this programming, constant consecutive writes from the Hub Interface to PCI are capable of occupying the PCI bus indefinitely without releasing FRAME#.

A value of 00h disables the timer such that the North PCI initiator logic is never forced to end a burst prematurely due to a timeout.

Table 159. Offset 1Bh: SMLT—Secondary Master Latency Timer Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	1Bh	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:3	Master Latency Count	5-bit value that indicates the number of PCI clocks, in 8-clock increments, that the Intel® 6300ESB ICH will remain as master of the bus.	R/W
2:0	Reserved	Reserved.	

7.1.14 Offset 1Ch: IOBASE—I/O Base Register (HUB-PCI—D30:F0)

Table 160. Offset 1Ch: IOBASE—I/O Base Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	1Ch	**Attribute:**	Read/Write
Default Value:	F0h	**Size:**	8-bit

Bits	Name	Description	Access
7:4	I/O Address Base bits [15:12]	I/O Base bits corresponding to address lines 15:12 for 4-Kbyte alignment. Bits 11:0 are assumed to be padded to 000h.	R/W
3:0	I/O Addressing Capability	This is hardwired to 0h, indicating that the Hub Interface to PCI bridge does not support 32-bit I/O addressing. This means that the I/O base and limit upper address registers must be read only.	RO

7.1.15 Offset 1Dh: IOLIM—I/O Limit Register (HUB-PCI—D30:F0)

Table 161. Offset 1Dh: IOLIM—I/O Limit Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	1Dh	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:4	I/O Address Limit bits [15:12]	I/O Base bits corresponding to address lines 15:12 for 4 Kbyte alignment. Bits 11:0 are assumed to be padded to FFFh.	R/W
3:0	I/O Addressing Capability	This is hardwired to 0h, indicating that the Hub Interface-to-PCI bridge does not support 32-bit I/O addressing. This means that the I/O base and limit upper address registers must be read only.	RO

7.1.16 Offset 1E - 1Fh: SECSTS—Secondary Status Register (HUB-PCI—D30:F0)

Note: For the writable bits in this register, writing a 1 will clear the bit. Writing a 0 to the bit will have no effect.

Table 162. Offset 1E - 1Fh: SECSTS—Secondary Status Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	1E-1Fh	***Attribute:***	Read/Write
Default Value:	0280h	***Size:***	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	0 = This bit is cleared by software writing a 1. 1 = The Intel® 6300ESB ICH detected a parity error on the PCI bus.	R/WC
14	Received System Error (SSE)	0 = Software clears this bit by writing a'1' to the bit position. 1 = SERR# assertion is received on PCI.	R/WC
13	Received Master Abort (RMA)	0 = Software clears this bit by writing a'1' to the bit position. 1 = Hub Interface to PCI cycle was master-aborted on PCI.	R/WC
12	Received Target Abort (RTA)	0 = Software clears this bit by writing a'1' to the bit position. 1 = Hub Interface to PCI cycle was target-aborted on PCI. For "completion required" cycles from the Hub Interface, this event should also set the Signaled Target Abort in the Primary Status Register in this device, and the Intel® 6300ESB ICH must send the "target abort" status back to the Hub Interface.	R/WC
11	Signaled Target Abort (STA)	Intel® 6300ESB ICH does not generate target aborts.	RO
10:9	DEVSEL# Timing Status	01h = Medium timing.	RO

Table 162. Offset 1E - 1Fh: SECSTS—Secondary Status Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	1E-1Fh	**Attribute:**	Read/Write
Default Value:	0280h	**Size:**	16-bit

Bits	Name	Description	Access
8	Master Data Parity Error Detected (MDPD)	0 = Software clears this bit by writing a'1' to the bit position. 1 = The Intel® 6300ESB ICH sets this bit when all of the following three conditions are met: - The Parity Error Response Enable bit in the Bridge Control Register (bit 0, offset 3Eh) is set - USB, AC'97 or IDE is a Master - PERR# asserts during a write cycle OR a parity error is detected internally during a read cycle	R/WC
7	Fast Back to Back	Hardwired to '1' to indicate that the PCI to Hub Interface target logic is capable of receiving fast back-to-back cycles.	RO
6	User Definable Features (UDF)	Hardwired to '0'.	RO
5	66 MHz Capable	Hardwired to '0'.	RO
4	PERR# Assertion Detect	This bit is set by hardware whenever the PERR# pin is asserted on the rising edge of PCI clock. This includes cases in which the chipset is the agent driving PERR#. It remains asserted until cleared by software writing a '1' to this location. When enabled by the PERR#-to-SERR# Enable bit (in the Bridge Control register), a '1' in this bit can generate an internal SERR# and be a source for the NMI logic.	R/WC
3:0	Reserved	Reserved.	

7.1.17 Offset 20 - 21h: MEMBASE—Memory Base Register (HUB-PCI—D30:F0)

This register defines the base of the Hub Interface to PCI non-prefetchable memory range. Since the Intel® 6300ESB ICH will forward all Hub Interface memory accesses to PCI, the Intel® 6300ESB ICH will only use this information for determining when not to accept cycles as a target.

This register must be initialized by the configuration software. For the purpose of address decode, address bits AD[19:0] are assumed to be zero. Thus, the bottom of the defined memory address range will be aligned to a 1 Mbyte boundary.

Table 163. Offset 20 - 21h: MEMBASE—Memory Base Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	20-21h	**Attribute:**	Read/Write
Default Value:	FFF0h	**Size:**	16-bit

Bits	Name	Description	Access
15:4	Memory Address Base	Defines the base of the memory range for PCI. These 12 bits correspond to address bits 31:20.	R/W
3:0	Reserved	Reserved.	

7.1.18 Offset 22 - 23h: MEMLIM—Memory Limit Register (HUB-PCI—D30:F0)

This register defines the upper limit of the Hub Interface to PCI non-prefetchable memory range. Since the Intel® 6300ESB ICH will forward all Hub Interface memory accesses to PCI, the Intel® 6300ESB ICH will only use this information for determining when not to accept cycles as a target.

This register must be initialized by the configuration software. For the purpose of address decode, address bits AD[19:0] are assumed to be FFFFFh. Thus, the top of the defined memory address range will be aligned to a 1 Mbyte boundary.

Table 164. Offset 22 - 23h: MEMLIM—Memory Limit Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	22-23h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15:4	Memory Address Limit	Defines the top of the memory range for PCI. These 12 bits correspond to address bits 31:20.	R/W
3:0	Reserved	Reserved.	

7.1.19 Offset 24h - 25h: PREF_MEM_BASE—Prefetchable Memory Base Register (HUB-PCI—D30:F0)

Offset Address:	24h-25h	Attribute:	R/W
Default Value:	0000FFF0h	Size:	16-bit

This register defines the Base Address of the Hub Interface-to-PCI prefetchable memory range. Since the Intel® 6300ESB ICH will forward all Hub Interface memory accesses to PCI, the Intel® 6300ESB ICH will only use this information for determining when *not* to accept cycles as a target.

Note: When the Hub Interface is acting as the initiator, it will not respond as a target.

This register must be initialized by the config software. For the purpose of address decode, address bits AD[19:0] are assumed to be FFFFFh. Thus, the top of the defined memory address range will be aligned to a 1 Mbyte boundary.

Table 165. Offset 24h - 25h: PREF_MEM_BASE—Prefetchable Memory Base Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	24h-25h	***Attribute:***	Read/Write
Default Value:	0000FFF0h	***Size:***	16-bit

Bits	Name	Description	Access
15:4	Prefetchable Memory Address Base	Defines the base address of the prefetchable memory address range for PCI. These 12 bits correspond to address bits 31:20.	R/W
3:0	Reserved	Reserved.	RO

7.1.20 Offset 26h-27h: PREF_MEM_MLT—Prefetchable Memory Limit Register (HUB-PCI—D30:F0)

This register defines the upper limit of the Hub Interface-to-PCI non-prefetchable memory range. Since the Intel® 6300ESB ICH will forward all Hub Interface memory accesses to PCI, the Intel® 6300ESB ICH will only use this information for determining when *not* to accept cycles as a target.

Note: When the Hub Interface is acting as the initiator, it will not respond as a target.

Note: This register must be initialized by the config software. For the purpose of address decode, address bits A[19:0] are assumed to be FFFFFh. Thus, the top of the defined memory address range will be aligned to a 1 Mbyte boundary.

Table 166. Offset 26h-27h: PREF_MEM_MLT—Prefetchable Memory Limit Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	26h-27h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	16-bit

Bits	Name	Description	Access
15:4	Prefetchable Memory Address Limit	Defines the limit address of the prefetchable memory address range for PCI. These 12 bits correspond to address bits 31:20.	RW
3:0	Reserved	Reserved.	RO

7.1.21 Offset 30 - 31h: IOBASE_HI—I/O Base Upper 16 Bits Register (HUB-PCI—D30:F0)

Table 167. Offset 30 - 31h: IOBASE_HI—I/O Base Upper 16 Bits Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	30-31h	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	I/O Address Base Upper 16 bits [31:16]	Not supported; hardwired to 0.	RO

7.1.22 Offset 32 - 33h: IOLIM_HI—I/O Limit Upper 16 Bits Register (HUB-PCI—D30:F0)

Table 168. Offset 32 - 33h: IOLIM_HI—I/O Limit Upper 16 Bits Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	32-33h	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit

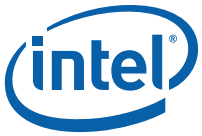
Bits	Name	Description	Access
15:0	I/O Address Limit Upper 16 bits [31:16]	Not supported; hardwired to 0.	RO

7.1.23 Offset 3Ch: INT_LINE—Interrupt Line Register (HUB-PCI—D30:F0)

Table 169. Offset 3Ch: INT_LINE—Interrupt Line Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	3Ch	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Interrupt Line Routing	Hardwired to 00h. The bridge does not generate interrupts, and interrupts from downstream devices are routed around the bridge.	RO



7.1.24 Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0)

Table 170. Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0) (Sheet 1 of 3)

Device:	30	***Function:***	0
Offset:	3E-3Fh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
12	PERR# to SERR# Enable	When this bit is set to '1' PCI PERR NMI reporting is enabled. In addition to setting this bit, you also must set bit 1 of D30_F0 PNE Register. Section 7.1.28 When this bit is set to a '1' and PERR# is asserted on PCI, the PERR# Assertion detect status bit (in the Secondary Status Register) will indicate a PERR# internal SERR# assertion. The SERR# can be a s source on NMI.	RW
11	Discard Timer SERR# Enable (DTSE)	Controls the generation of SERR# on the primary interface in response to a timer discard on the secondary interface: • When '0': Do not generate SERR# on a secondary timer discard • When '1': Generate SERR# in response to a secondary timer discard. *This bit replaces bit 1 of offset 90h, which held this function in ICH3.*	R/W
10	Discard Timer Status (DTS)	This bit is set to a '1' when the secondary discard timer expires (there is no discard timer for the primary interface). *This bit replaces bit 1 of offset 92h, which held this function in ICH3.*	R/W
9	Secondary Discard Timer (SDT)	Sets the maximum number of PCI clock cycles that the Intel® 6300ESB ICH waits for an initiator on PCI to repeat a delayed transaction request. The counter starts once the delayed transaction completion is at the head of the queue. When the master has not repeated the transaction at least once before the counter expires, the Intel® 6300ESB ICH discards the transaction from its queue. • When '0': The PCI master timeout value is between 2^{15} and 2^{16} PCI clocks • When '1': The PCI master timeout value is between 2^{10} and 2^{11} PCI clocks	R/W
8	**Primary Discard Timer (PDT)**	This bit is RW for software compatibility only.	R/W
7	Fast Back to Back Enable	Hardwired to '0'. The PCI logic will not generate fast back-to-back cycles on the PCI bus.	

Table 170. Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0) (Sheet 2 of 3)

Device:	30	***Function:***	0
Offset:	3E-3Fh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
6	Secondary Bus Reset	Controls PXPCIRST# assertion on PCI(X). 1 = The Intel® 6300ESB ICH asserts PCIXSBRST#. When PCIXSBRST# is asserted, the data buffers between the Hub Interface and PCI(X) and the PCI(X) bus are initialized back to reset conditions. The Hub Interface and the configuration registers are not affected. 0 = The Intel® 6300ESB ICH deasserts PCIXSBRST#	R/W
5	Master Abort Mode	This bit controls the behavior of the Intel® 6300ESB ICH when a master abort occurs on a transaction that crosses the Hub Interface-PCI bridge in either direction. The default is 0. When set to 0, the Intel® 6300ESB ICH behaves in the following manner: • Hub Interface Completion-Required requests to PCI: When these master abort on PCI, the Intel® 6300ESB ICH returns a master abort status. For reads, FFFFh is returned for each DWORD. • Hub Interface Posted Writes to PCI: When these master abort on PCI, the Intel® 6300ESB ICH discards the data. • PCI Reads to Hub Interface: When these master abort on Hub Interface, the Intel® 6300ESB ICH returns the data provided with the Hub Interface master abort packet to the PCI requestor. • PCI writes to Hub Interface: Intel® 6300ESB ICH has no idea when these “master-abort.” When set to 1, the Intel® 6300ESB ICH treats the master abort as an error: • Hub Interface Completion-Required requests to PCI: When these master abort on PCI, the Intel® 6300ESB ICH returns a target abort status. For reads, FFFFh is returned for each DWORD. • Hub Interface Posted Writes to PCI: When these master abort on PCI, the Intel® 6300ESB ICH discards the data and sets the Primary Signaled SERR# bit (when the corresponding SERR_EN bit is set). • PCI Reads to Hub Interface: When these master abort on Hub Interface, the Intel® 6300ESB ICH terminates the cycle with a target abort and flushes the remainder of the prefetched data. • PCI writes to Hub Interface: The Intel® 6300ESB ICH has no idea when these “master-abort.”	R/W
	VGA 16-Bit Decode	This bit does not have any functionality relative to address decodes because the Intel® 6300ESB ICH will forward the cycles to PCI, independent of the decode. Writes of one have no impact other than to force the bit to one. Writes of zero have no impact other than to force the bit to zero. Reads to this bit will return the previously written value (or zero when no writes since reset).	

Table 170. Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0) (Sheet 3 of 3)

Device:	30	***Function:***	0
Offset:	3E-3Fh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

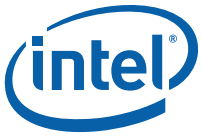
Bits	Name	Description	Access
3	VGA Enable	0 = No VGA device on PCI. 1 = Indicates that the VGA device is on PCI. Therefore, the PCI to Hub Interface decoder will not accept memory cycles in the range A0000h-BFFFFh. Note that the Intel® 6300ESB ICH will never take I/O cycles in the VGA range from PCI. If VGA is enabled on PCI-X Bridge Device 28 Function 0, offset 3Eh, bit 3. PCI-X will claim memory cycles in the VGA range before the legacy PCI Bridge.	R/W
2	ISA Enable	The Intel® 6300ESB ICH ignores this bit. However, this bit is read/write for software compatibility. Since the Intel® 6300ESB ICH forwards all I/O cycles that are not in the USB, AC'97, or IDE ranges to PCI, this bit would have no effect. ISA should be enabled on the legacy PCI or the PCI/PCI-X bridge, but NOT both. If both are enabled, unpredictable results will occur.	R/W
1	SERR# Enable	0 = Disable 1 = When this bit is set AND bit 8 in CMD register (D30:F0 Offset 04h) is also set, the Intel® 6300ESB ICH will set the SSE bit in PD_STS register (D30:F0, offset 06h, bit 14) **and** also generate an NMI (or SMI# if NMI routed to SMI) when the SERR# signal is asserted. The internal SERR# will be generated only if the SERR_EN bit is also set in offset 04h. **NOTE:** See Section 5.1.4, "SERR# Functionality" for more details on this bit.	R/W
0	Parity Error Response Enable	0 = Disable 1 = Enable the Hub Interface to PCI bridge for parity error detection and reporting on the PCI bus.	R/W

7.1.25 Offset 40 - 43h: HI_CMD—Hub Interface Command Control Register (HUB-PCI—D30:F0)

Table 171. Offset 40 - 43h: HI_CMD—Hub Interface Command Control Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	40-43h	**Attribute:**	Read/Write
Default Value:	76202802h	**Size:**	32-bit

Bits	Name	Description	Access
31:20	Reserved	Reserved.	
30:28	SATA Hub ID	This field identifies the Hub Interface ID number for the Serial ATA requests on the Hub Interface. Default=111b	RO
27	Reserved	Reserved.	RO
26:24	Second Hub ID	This field identifies the Hub Interface ID number for integrated requesters. Default=110b	RO
23:21	First Hub ID	This field identifies the Hub Interface ID number for integrated requesters. Hardwired to 001b	RO
20	Hub Interface Parity Unsupported	When set to 1, the Intel® 6300ESB ICH will not check parity on the Hub Interface even if enabled to do so according to the Parity Error Response bit in D30.F0.04h bit 6 or if the Parity Error Response bit of D28.F0.04h bit 6 is set.	R/W
19:16	Hub Interface Timeslice	This field sets the Hub Interface arbiter time-slice value with four base-clock granularity. A value of 0h means that the time-slice is immediately expired and that the Intel® 6300ESB ICH will allow the other master's request to be serviced after every message.	R/W
15:14	HI Width	This field is hardwired to 00b, indicating that the Hub Interface is eight bits wide.	RO
13	HI Rate_Valid	Hardwired to '1'.	RO
12:10	HI Rate	Encoded value representing the clock-to-transfer rate of the Hub Interface: 1:4 = 010b The value is loaded at reset by sampling the capability of the device connected to the Hub Interface port. The value for this field is fixed for 4x mode only.	RO
9:4	Reserved	Reserved.	
3:1	Max Data (MAXD)	Hardwired to 001b. This field specifies the maximum amount of data that the Intel® 6300ESB ICH is allowed to burst in one packet on the Hub Interface. The Intel® 6300ESB ICH will always do 64 byte bursts.	RO
0	Reserved	Reserved.	



7.1.26 Offset 44 - 45h: DEVICE_HIDE—Secondary PCI Device Hiding Register (HUB-PCI—D30:F0)

This register allows software to “hide” PCI devices. Specifically, when PCI devices are hidden, the configuration space is not accessible because the PCI IDSEL pin does not assert. The Intel® 6300ESB ICH supports the ability to hide four external devices (0 through 3).

Hiding a PCI device may be useful for debugging, bug work-arounds, and system management support. Devices should only be hidden during initialization before any configuration cycles are run. This ensures that the device is not in a semi-enable state.

Table 172. Offset 44 - 45h: DEVICE_HIDE—Secondary PCI Device Hiding Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	44-45h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	16-bit
		Power Well:	00h

Bits	Name	Description	Access
15:9	Reserved	Reserved.	
8	Reserved	Reserved.	
7:6	Reserved	Reserved.	
5	Reserved	Reserved.	
4	Reserved	Reserved.	
3	HIDE_DEV3	Same as bit 0 of this register, except for device 3 (AD{19]).	
2	HIDE_DEV2	Same as bit 0 of this register, except for device 2 (AD{18]).	
1	HIDE_DEV1	Same as bit 0 of this register, except for device 1 (AD[17])	
0	HIDE_DEV0	When this bit is set, it hides device 0 on the PCI bus. This is done by masking the IDSEL (keeping it low) for configuration cycles to that device. Since the device will not see its IDSEL go active, it will not respond to PCI configuration cycles and the processor will think the device is not present. AD[16] is used as IDSEL for device 0. When this bit is a 0, the PCI configuration cycles for this slot are not affected.	

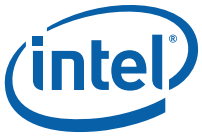
7.1.27 Offset 50 - 51h: CNF—Intel® 6300ESB ICH Configuration Register (HUB-PCI—D30:F0)

Table 173. Offset 50 - 51h: CNF—Intel® 6300ESB ICH Configuration Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	50-51h	***Attribute:***	Read/Write
Default Value:	1400h	***Size:***	16-bit

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13	Prefetch Flush Enable	**Prefetch Flush Enable**. 0 = Prefetch Flush Disable 1 = Causes CPU to PCI logic to only deliver “Demand” data for a delayed transaction if a processor-to-PCI write has occurred since the delayed transaction was initiated. (Default) **NOTE:** This bit must be set by system BIOS.	R/W
12:10	Reserved	Reserved.	
9	HP_PCI_EN	High Priority PCI Enable 0 = All PCI REQ#/GNT pairs have the same arbitration priority. 1 = Enables a mode where the REQ[0]#/GNT[0]# signal pair has a higher arbitration priority.	R/W
8	Hole Enable (15 MB-16 MB)	0 = Disable 1 = Enables the 15 Mbyte to 16 Mbyte hole in the DRAM.	R/W
7	Reserved	Reserved.	
6	HI-PCI Write Combining Enable	HI-PCI Write Combining Enable 1 = Disables Hub Interface to PCI Write combining	R/W
5:3	Reserved	Reserved.	
2	**Delayed Transaction Discard Timer**	**Delayed Transaction Discard Timer** When set to 1 this bit shortens all delayed transaction discard timers to 128 PCI clocks.	R/W
1:0	Reserved	Reserved.	

NOTE: Refer to the latest revision of the Intel® 6300ESB ICH BIOS Specification for the recommended configuration of this register.



7.1.28 Offset 58 - 5Bh: D30_PNE — PERR#_NMI_ENABLE Register (HUB-PCI—D30:F0)

Table 174. Offset 58 - 5Bh: D30_PNE — PERR#_NMI_ENABLE Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	58-5Bh	***Attribute:***	Read/Write
Default Value:	01341150h	***Size:***	32-bit

Bits	Name	Description	Access
31:22	Reserved	Reserved.	
21:20	Outstanding SATA Mem Read Acceptance	Indicate the number of outstanding (not completed) memory read requests from SATA that are accepted. The value of the bits is decoded as follows: 00 = Disable this feature by allowing Stunit to accept any number of outstanding memory reads from SATA until its buffer resources are full. 11 Accepts four outstanding reads from SATA. **Default** **NOTE:** Default value is 11b, BIOS must write 00 to this register	R/W
19:8	Reserved	Reserved.	
7	SATA Request Enable	0 = The arbiter will block S-ATA requests when there is a lock transaction targeting PCIX. 1 = the arbiter will allow S-ATA requests even when there is a lock transaction targeting PCIX. **NOTE:** Default value should be 0b, BIOS must write 1 to this register	
6:3	Reserved	Reserved.	
2	PCI-X PERR NMI Reporting	This bit enables PCI-X PERR NMI reporting. When this bit is set to '1' PCI-X PERR NMI reporting is enabled. Once enabled, this bit should not change. In addition to setting this bit, you also must set bit 12 of D30_F0 Bridge Control Register. See section Section 18.6.1.21. Default = 0	R/W
1	PCI PERR NMI Reporting	This bit enables PCI PERR NMI reporting. When this bit is set to '1' PCI PERR NMI reporting is enabled. Once enabled, this bit should not change. In addition to setting this bit, you also must set bit 12 of D30_F0 Bridge Control Register. See section Section 18.6.1.21. Default = 0	R/W
0	Reserved	Reserved.	

7.1.29 Offset 70h: MTT—Multi-Transaction Timer Register (HUB-PCI—D30:F0)

MTT is an 8-bit register that controls the amount of time that the Intel® 6300ESB ICH's arbiter allows a PCI initiator to perform multiple back-to-back transactions on the PCI bus. The Intel® 6300ESB ICH's MTT mechanism is used to ensure a fair share of the Primary PCI bandwidth to an initiator that performs multiple back-to-back transactions to fragmented memory ranges (and as a consequence it may not use long burst transfers).

The number of clocks programmed in the MTT represents the ensured time slice (measured in PCI clocks) allotted to the current agent, after which the arbiter will grant another agent that is requesting the bus. The MTT value must be programmed with 8-clock granularity in the same manner as MLT. For example, if the MTT is programmed to 18h, then the selected value corresponds to the time period of 24 PCI clocks.The default value of MTT is 20h (32 PCI clocks).

Note: Programming the MTT to a value of 00h disables this function, which could cause starvation problems for some PCI master devices. Programming of the MTT to anything less than 16 clocks will not allow the Grant-to-FRAME# latency to be 16 clocks. The MTT timer will timeout before the Grant-to-FRAME# trigger causing a rearbitration.

Table 175. Offset 70h: MTT—Multi-Transaction Timer Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	70h	***Attribute:***	Read/Write
Default Value:	20h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Multi-Transaction Timer Count Value	This field specifies the amount of time that grant will remain asserted to a master continuously asserting its request for multiple transfers. This field specifies the count in an 8-clock (PCI clock) granularity.	R/W
2:0	Reserved	Reserved.	

7.1.30 Offset 82h: PCI_MAST_STS—PCI Master Status Register (HUB-PCI—D30:F0)

Table 176. Offset 82h: PCI_MAST_STS—PCI Master Status Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	82h	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7	Internal South PCI Master Request Status (INT_MREQ_STS)	Allows software to see if the internal DMA controller or LPC has requested use of the PCI bus. 0 = Software clears this bit by writing a'1'to the bit position. 1 = The Intel® 6300ESB ICH's internal DMA controller or LPC has requested use of the PCI bus.	R/WC
6:4	Reserved	Reserved.	
3:0	PCI Master Request Status (PCI_MREQ_STS)	Allows software to see if a particular bus master has requested use of the PCI bus. 0 = Software clears these bits by writing a 1 to the bit position. 1 = The associated PCI master has requested use of the PCI bus.	R/WC

7.1.31 Offset 90h: ERR_CMD—Error Command Register (HUB-PCI—D30:F0)

Note: This register configures the Intel® 6300ESB ICH's Device 30 responses to various system errors. The actual assertion of the internal SERR# (routed to cause NMI# or SMI#) is enabled through the PCI Command register.

Table 177. Offset 90h: ERR_CMD—Error Command Register (HUB-PCI—D30:F0)

Device:	30	**Function:**	0
Offset:	90h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SERR# Enable on Receiving Target Abort (SERR_RTA_EN)	0 = Disable. 1 = Enable. When SERR_EN is set, the Intel® 6300ESB ICH will report SERR# when SERR_RTA is set.	R/W
1:0	Reserved	Reserved. Bit 1 was the SERR# Enabled for Delayed Transaction Timeout, see Section 7.1.24, "Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0)".	

7.1.32 Offset 92h: ERR_STS—Error Status Register (HUB-PCI—D30:F0)

Note: This register records the cause of system errors in Device 30. The actual assertion of SERR# is enabled through the PCI Command register.

Table 178. Offset 92h: ERR_STS—Error Status Register (HUB-PCI—D30:F0)

Device:	30	***Function:***	0
Offset:	92h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SERR# Due to Received Target Abort (SERR_RTA)	0 = This bit is cleared by writing a 1. 1 = Intel® 6300ESB ICH sets this bit when the Intel® 6300ESB ICH receives a target abort. When SERR_EN, the Intel® 6300ESB ICH will also generate an SERR# when SERR_RTA is set.	R/W
1:0	Reserved	Reserved. Bit 1 was the SERR# Enabled for Delayed Transaction Timeout, see Section 7.1.24, "Offset 3E - 3Fh: BRIDGE_CNT—Bridge Control Register (HUB-PCI—D30:F0)".	

7.1.33 Offset F8h - FBh: MANID— Manufacturer's ID

Table 179. Offset F8h - FBh: MANID— Manufacturer's ID

Device:	30	***Function:***	0
Offset:	F8h-FBh	***Attribute:***	Read-Only
Default Value:	0000 0F66h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:8	Manufacturer	0Fh = Intel	RO
7:0	Process/Dot	66h = Process 859.6	RO

LPC I/F Bridge Registers (D31:F0) 8

The LPC Bridge function of the Intel® 6300ESB ICH resides in PCI Device 31:Function 0. This function contains many other functional units, such as DMA and Interrupt Controllers, Timers, Power Management, System Management, GPIO RTC and LPC.

Registers and functions associated with other functional units (USB UHCI, USB EHCI, IDE, etc.) are described in their respective sections.

8.1 PCI Configuration Registers (D31:F0)

Note: Registers that are not shown should be treated as reserved. (See Section 6.2, "PCI Configuration Map" for details).

Table 180. PCI Configuration Registers (D31:F0) (Sheet 1 of 2)

Offset	Mnemonic	Register Name	Default	Type
00-01h	VID	Vendor ID	8086h	RO
02-03h	DID	Device ID	25A1h	RO
04-05h	PCICMD	PCI Command Register	000Fh	R/W
06-07h	PCISTA	PCI Device Status	0280h	R/W
08h	RID	Revision ID	See NOTE:	RO
09h	PI	Programming Interface	00h	RO
0Ah	SCC	Sub Class Code	01h	RO
0Bh	BCC	Base Class Code	06h	RO
0Eh	HEADTYP	Header Type	80h	RO
40-43h	PMBASE	ACPI Base Address	00000001h	R/W
44h	ACPI_CNTL	ACPI Control	00h	R/W
4E-4Fh	BIOS_CNTL	BIOS Control	0000h	R/W
54h	TCO_CNTL	TCO Control	00h	R/W
58-5Bh	GPIO_BASE	GPIO Base Address	00000001h	R/W
5Ch	GPIO_CNTL	GPIO Control	00h	R/W
60-63h	PIRQ[*n*]_ROUT	PIRQ[A-D] Routing Control	80808080h	R/W
64h	SERIRQ	Serial IRQ Control	50h	R/W
68-6Bh	PIRQ[*n*]_ROUT	PIRQ[E-H] Routing Control	80808080h	R/W
88h	D31_ERR_CFG	Device 31 Error Config	00h	R/W
8Ah	D31_ERR_STS	Device 31 Error Status	00h	R/W
90-91h	PCI_DMA_CFG	PCI DMA Configuration	0000h	R/W

NOTE: Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID register.

Table 180. PCI Configuration Registers (D31:F0) (Sheet 2 of 2)

Offset	Mnemonic	Register Name	Default	Type
A0-CFh		Power Management		
D0-D3h	GEN_CNTL	General Control	00000080h	R/W
D4h	GEN_STA	General Status	0Xh	R/W
D5h	BACK_CNTL	Backed Up Control	00f	R/W
D8h	RTC_CONF	Real Time Clock Configuration	00h	R/W
E0h	COM_DEC	LPC I/F COM Port Decode Ranges	00h	R/W
E1h	FDD/LPT_DEC	LPC I/F FDD and LPT Decode Ranges	00h	R/W
E2h	SND_DEC	LPC I/F Sound Decode Ranges	00h	R/W
E3h	FWH_DEC_EN1	FWH Decode Enable 1	FFh	R/W
E4-E5h	GEN1_DEC	LPC I/F General 1 Decode Range	0000h	R/W
E6-E7h	LPC_EN	LPC I/F Enables	00h	R/W
E8-EBh	FWH_SEL1	FWH Select 1	00112233h	R/W
EC-EDh	GEN2_DEC	LPC I/F General 2 Decode Range	0000h	R/W
EE-EFh	FWH_SEL2	FWH Select 2	5678h	R/W
F0h	FWH_DEC_EN2	FWH Decode Enable 2	0Fh	R/W
F2h	FUNC_DIS	Function Disable Register	0080h	R/W
F4-F7h	ETR1	PCI-X Extended Features	00000000h	R/W
F8h-FBh		Manufacturer's ID	000S0F66	RO

NOTE: Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID register.

8.1.1 Offset 00 - 01h: VID—Vendor ID Register (LPC I/F—D31:F0)

Table 181. Offset 00 - 01h: VID—Vendor ID Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	00-01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel. Intel VID = 8086h	RO

8.1.2 Offset 02 - 03h: DID—Device ID Register (LPC I/F—D31:F0)

Table 182. Offset 02 - 03h: DID—Device ID Register (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	02-03h	**Attribute:**	Read-Only
Default Value:	25A1h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Device ID Value	This is a 16-bit value assigned to the Intel® 6300ESB ICH LPC Bridge.	

8.1.3 Offset 04 - 05h: PCICMD—PCI COMMAND Register (LPC I/F—D31:F0)

Table 183. Offset 04 - 05h: PCICMD—PCI COMMAND Register (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	04-05h	**Attribute:**	Read/Write
Default Value:	000Fh	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

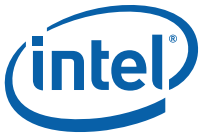
Bits	Name	Description	Access
15:10	Reserved	Reserved.	
9	FBE: Fast Back to Back Enable	Hardwired to 0.	RO
8	SERR_EN: SERR# Enable	0 = Disable. 1 = Enable. Allow SERR# to be generated.	
7	WCC: Wait Cycle Control	Hardwired to 0.	RO
6	PER: Parity Error Response	0 = No action is taken when detecting a parity error. 1 = The processor will take normal action when a parity error is detected.	
5	VPS: VGA Palette Snoop	Hardwired to 0	RO
4	PMWE: Postable Memory Write Enable	Hardwired to 0	RO
3	SCE: Special Cycle Enable	Hardwired to 1.	
2	BME: Bus Master Enable	Hardwired to 1 to indicate that bus mastering cannot be disabled for function 0 (DMA/ISA Master)	RO
1	MSE: Memory Space Enable	Hardwired to 1 to indicate that memory space cannot be disabled for Function 0 (LPC I/F)	RO
0	IOE: I/O Space Enable	Hardwired to 1 to indicate that the I/O space cannot be disabled for function 0 (LPC I/F)	RO

8.1.4 Offset 06 - 07h: PCISTA—PCI Device Status (LPC I/F—D31:F0)

Table 184. Offset 06 - 07h: PCISTA—PCI Device Status (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	06-07h	**Attribute:**	Read/Write Clear
Default Value:	0280h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15	DPE: Detected Parity Error	0 = This bit is cleared by software writing a 1 to the bit position. 1 = PERR# signal goes active. Set even when the PER bit is 0.	R/W
14	SSE: Signaled System Error	0 = This bit is cleared by software writing a 1 to the bit position. 1 = Set by the Intel® 6300ESB ICH n the SERR_EN bit is set and the Intel® 6300ESB ICH generates an SERR# on function 0. The ERR_STS register may be read to determine the cause of the SERR#. The SERR# may be routed to cause SMI#, NMI, or interrupt.	R/W
13	RMA: Received Master Abort	0 = This bit is cleared by software writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH generated a master abort on PCI due to LPC I/F master or DMA cycles.	R/W
12	RTA: Received Target Abort	0 = This bit is cleared by software writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH received a target abort during LPC I/F master or DMA cycles to PCI.	R/W
11	STA: Signaled Target Abort	0 = This bit is cleared by software writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH generated a target abort condition on PCI cycles claimed by the Intel® 6300ESB ICH for Intel® 6300ESB ICH internal registers or for going to LPC I/F.	R/W
10:9	DEV_STS: DEVSEL# Timing Status	01 = Medium Timing.	RO
8	DPED: Data Parity Error Detected	0 = This bit is cleared by software writing a 1 to the bit position. 1 = Set when all three of the following conditions are true: - The Intel® 6300ESB ICH is the initiator of the cycle, - The Intel® 6300ESB ICH asserted PERR# (for reads) or observed PERR# (for writes), and - The PER bit is set.	R/WC
7	FB2B: Fast Back to Back	Always 1. Indicates the Intel® 6300ESB ICH as a target may accept fast back-to-back transactions.	RO
6	UDF: User Definable Features	Hardwired to 0	
5	66MHZ_CAP: 66 MHz Capable	Hardwired to 0	RO
4:0	Reserved	Reserved.	



8.1.5 Offset 08h: RID—Revision ID Register (LPC I/F—D31:F0)

Table 185. Offset 08h: RID—Revision ID Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

8.1.6 Offset 09h: PI—Programming Interface (LPC I/F—D31:F0)

Table 186. Offset 09h: PI—Programming Interface (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	09h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Programming Interface Value	Programming Interface Value.	RO

8.1.7 Offset 0Ah: SCC—Sub-Class Code Register (LPC I/F—D31:F0)

Table 187. Offset 0Ah: SCC—Sub-Class Code Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub-Class Code	8-bit value that indicates the category of bridge for the LPC PCI bridge.	RO

8.1.8 Offset 0Bh: BCC—Base-Class Code Register (LPC I/F—D31:F0)

Table 188. Offset 0Bh: BCC—Base-Class Code Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	06h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	8-bit value that indicates the type of device for the LPC bridge. The code is 06h indicating a bridge device.	RO

8.1.9 Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—D31:F0)

Table 189. Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	0Eh	***Attribute:***	Read-Only
Default Value:	0Eh	***Size:***	8-bit

Bits	Name	Description	Access
7	Multi-Function Device	This bit is 1 to indicate a multi-function device.	RO
6:0	Header Type	7-bit field identifies the header layout of the configuration space.	RO

8.1.10 Offset 40 - 43h: PMBASE—ACPI Base Address (LPC I/F—D31:F0)

Note: Usage: ACPI or Legacy.

Note: Sets base address for ACPI I/O registers and TCO I/O registers. May be mapped anywhere in the 64K I/O space on 128-byte boundaries.

Table 190. Offset 40 - 43h: PMBASE—ACPI Base Address (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	40-43h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:7	Base Address	Provides 128 bytes of I/O space for ACPI, GPIO, and TCO logic. This is placed on a 128-byte boundary.	R/W
6:1	Reserved	Reserved.	
0	Resource Indicator	Tied to 1 to indicate I/O space.	RO

8.1.11 Offset 44h: ACPI_CNTL—ACPI Control (LPC I/F—D31:F0)

Note: Usage: ACPI or Legacy.

Table 191. Offset 44h: ACPI_CNTL—ACPI Control (LPC I/F—D31:F0) (Sheet 1 of 2)

Device:	31	***Function:***	0
Offset:	44h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved.	

Table 191. Offset 44h: ACPI_CNTL—ACPI Control (LPC I/F—D31:F0) (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	44h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
4	ACPI_EN: ACPI Enable	0 = Disable. 1 = Decode of the I/O range pointed to by the ACPI base register is enabled, and the ACPI power management function is enabled. Note that the APM power management ranges (B2/B3h) are always enabled and are not affected by this bit.	R/W
3	Reserved	Reserved.	
2:0	SCI_IRQ_SEL: SCI IRQ Select	Specifies on which IRQ the SCI will internally appear. When not using the APIC, the SCI must be routed to IRQ9-11, and that interrupt is not shared with the SERIRQ stream, but may be shared with other PCI interrupts. When using the APIC, the SCI may also be mapped to IRQ20-23, and may be shared with other interrupts. **Bits SCI Map** 000 IRQ9 001 IRQ10 010 IRQ11 011 Reserved 100 IRQ20 (Only available when APIC enabled) 101 IRQ21 (Only available when APIC enabled) 110 IRQ22 (Only available when APIC enabled) 111 IRQ23 (Only available when APIC enabled) **NOTE:** When the interrupt is mapped to APIC interrupts 9, 10, or 11, the APIC should be programmed for active-high reception. When the interrupt is mapped to APIC interrupts 20 through 23, the APIC should be programmed for active-low reception.	R/W

8.1.12 Offset 4E - 4Fh: BIOS_CNTL (LPC I/F—D31:F0)

Table 192. Offset 4E - 4Fh: BIOS_CNTL (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	4E-4Fh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:2	Reserved	Reserved.	
1	BLE: BIOS Lock Enable	0 = Setting the BIOSWE will not cause SMIs. Once set, this bit may only be cleared by a PXPCIRST#. 1 = Enables setting the BIOSWE bit to cause SMIs.	R/W
0	BIOSWE: BIOS Write Enable	0 = Only read cycles result in FWH I/F cycles. 1 = Access to the BIOS space is enabled for both read and write cycles. When this bit is written from a 0 to a 1 and BIOS lock Enable (BLE) is also set, an SMI# is generated. This ensures that only SMI code may update BIOS.	R/W

8.1.13 Offset 54h: TCO_CNTL—TCO Control (LPC I/F—D31:F0)

Table 193. Offset 54h: TCO_CNTL—TCO Control (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	54h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3	TCO_INT_EN: TCO Interrupt Enable	This bit enables/disables the TCO interrupt. 0 = Disables TCO interrupt. 1 = Enables TCO Interrupt, as selected by the TCO_INT_SEL field.	R/W
2:0	TCO_INT_SEL: TCO Interrupt Select	Specifies on which IRQ the TCO will internally appear. When not using the APIC, the TCO interrupt must be routed to IRQ9-11, and that interrupt is not shared with the SERIRQ stream, but may be shared with other PCI interrupts. When using the APIC, the TCO interrupt may also be mapped to IRQ20-23, and may be shared with other interrupt. Note that when the TCOSCI_EN bit is set (bit 6 of the GPEO_EN register), the TCO interrupt will be sent to the same interrupt as the SCI, and the TCO_INT_SEL bits will have no meaning. When the TCO interrupt is mapped to APIC interrupts 9, 10 or 11, the signal is in fact active high. When the TCO interrupt is mapped to IRQ 20, 21, 22, or 23 the signal is active low and may be shared with PCI interrupts that may be mapped to those same signals (IRQs). **Bits — SCI Map** 000 — IRQ9 001 — IRQ10 010 — IRQ11 011 — Reserved 100 — IRQ20 (Only available when APIC enabled) 101 — IRQ21 (Only available when APIC enabled) 110 — IRQ22 (Only available when APIC enabled) 111 — IRQ23 (Only available when APIC enabled)	R/W

8.1.14 Offset 58h - 5Bh: GPIO_BASE—GPIO Base Address (LPC I/F—D31:F0)

Note: Sets base address for GPIO registers. May be mapped anywhere in the 64K I/O space on 128-byte boundaries.

Table 194. Offset 58h - 5Bh: GPIO_BASE—GPIO Base Address (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	58h-5Bh	**Attribute:**	Read/Write
Default Value:	00000001h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:6	Base Address	Provides the 64 bytes of I/O space for GPIO.	R/W
5:1	Reserved	Reserved.	
0	Resource Indicator	Tied to 1 to indicate I/O space.	RO

8.1.15 Offset 5Ch: GPIO_CNTL—GPIO Control (LPC I/F—D31:F0)

Table 195. Offset 5Ch: GPIO_CNTL—GPIO Control (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	5Ch	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	GPIO_EN: GPIO Enable	This bit enables/disables decode of the I/O range pointed to by the GPIO base register and enables/disables the GPIO function. 0 = Disable. 1 = Enable.	R/W
3:0	Reserved	Reserved.	

8.1.16 Offset PIRQA - 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control (LPC I/F—D31:F0)

Table 196. Offset PIRQA - 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	PIRQA - 60h, PIRQB - 61h, PIRQC - 62h, PIRQD - 63h	***Attribute:***	Read/Write
Default Value:	80h	***Size:***	8-bit

Bits	Name	Description	Access
7	IRQEN: Interrupt Routing Enable	0 = The corresponding PIRQ is routed to one of the ISA-compatible interrupts specified in bits[3:0]. 1 = The PIRQ is not routed to the 8259. **NOTE:** BIOS must program this bit to "0" during POST for any of the PIRQs that are being used. The value of this bit may subsequently be changed by the OS when setting up for I/O APIC interrupt delivery mode.	R/W
6:4	Reserved	Reserved.	
3:0	IRQ Routing	(ISA compatible) **Bits Mapping Bits Mapping** 0000 = Reserved 1000 = Reserved 0001 = Reserved 1001 = IRQ9 0010 = Reserved 1010 = IRQ10 0011 = IRQ3 1011 = IRQ11 0100 = IRQ4 1100 = IRQ12 0101 = IRQ5 1101 = Reserved 0110 = IRQ6 1110 = IRQ14 0111 = IRQ7 1111 = IRQ15	R/W

8.1.17 Offset 64h: SERIRQ_CNTL—Serial IRQ Control (LPC I/F—D31:F0)

Table 197. Offset 64h: SERIRQ_CNTL—Serial IRQ Control (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	64h	***Attribute:***	Read/Write
Default Value:	10h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7	SIRQEN: Serial IRQ Enable	0 = The buffer is input only and internally SERIRQ will be a 1. 1 = Serial IRQs will be recognized. The SERIRQ pin will be configured as SERIRQ.	R/W
6	SIRQMD: Serial IRQ Mode Select	0 = The serial IRQ machine will be in quiet mode. 1 = The serial IRQ machine will be in continuous mode. **NOTE:** For systems using Quiet Mode, this bit should be set to 1 (Continuous Mode) for at least one frame after coming out of reset before switching back to Quiet Mode. Failure to do so will result in the Intel® 6300ESB ICH not recognizing SERIRQ interrupts. In order to enable UARTS, the LPC Master should be programmed to be in continuous mode. UART by default will be programmed to be in continuous mode.	R/W
5:2	SIRQSZ: Serial IRQ Frame Size	Fixed field that indicates the size of the SERIRQ frame. In the Intel® 6300ESB ICH, this field needs to be programmed to 21 frames (0100). This is an offset from a base of 17 which is the smallest data frame size.	R/W
1:0	SFPW: Start Frame Pulse Width	This is the number of PCI clocks that the SERIRQ pin will be driven low by the serial IRQ machine to signal a start frame. In continuous mode, the Intel® 6300ESB ICH will drive the start frame for the number of clocks specified. In quiet mode, the Intel® 6300ESB ICH will drive the start frame for the number of clocks specified minus one, as the first clock was driven by the peripheral. 00 = 4 clocks 01 = 6 clocks 10 = 8 clocks 11 = Reserved	R/W

8.1.18 Offset PIRQE - 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control (LPC I/F—D31:F0)

Table 198. Offset PIRQE - 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	PIRQE - 68h, PIRQF - 69h, PIRQG - 6Ah, PIRQH - 6Bh	***Attribute:***	Read/Write
Default Value:	80h	***Size:***	8-bit

Bits	Name	Description	Access
7	IRQEN	Interrupt Routing Enable 0 = The corresponding PIRQ is routed to one of the ISA-compatible interrupts specified in bits[3:0]. 1 = The PIRQ is not routed to the 8259. **NOTE:** BIOS must program this bit to '0' during POST for any of the PIRQs that are being used. The value of this bit may subsequently be changed by the OS when setting up for I/O APIC interrupt delivery mode.	R/W
6:4	Reserved	Reserved.	
3:0	IRQ Routing	(ISA compatible) 0000 = Reserved1000 = Reserved 0001 = Reserved1001 = IRQ9 0010 = Reserved1010 = IRQ10 0011 = IRQ31011 = IRQ11 0100 = IRQ41100 = IRQ12 0101 = IRQ51101 = Reserved 0110 = IRQ61110 = IRQ14 0111 = IRQ71111 = IRQ15	R/W

8.1.19 Offset 88h: D31_ERR_CFG—Device 31 Error Config Register (LPC I/F—D31:F0)

Note: This register configures the Intel® 6300ESB ICH's Device 31 responses to various system errors. The actual assertion of SERR# is enabled through the PCI Command register.

Table 199. Offset 88h: D31_ERR_CFG—Device 31 Error Config Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	88h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SERR_RTA_EN: SERR# on Received Target Abort Enable	0 = Disable. No SERR# assertion on Received Target Abort. 1 = The Intel® 6300ESB ICH will generate SERR# if the SERR_RTA is set and if SERR_EN is set.	R/W
1	SERR_DTT_EN: SERR# on Delayed Transaction Timeout Enable	0 = Disable. No SERR# assertion on Delayed Transaction Timeout. 1 = The Intel® 6300ESB ICH will generate SERR# if the SERR_DTT bit is set and if SERR_EN is set.	R/W
0	Reserved	Reserved.	

8.1.20 Offset 8Ah: D31_ERR_STS—Device 31 Error Status Register (LPC I/F—D31:F0)

Note: This register configures the Intel® 6300ESB ICH's Device 31 responses to various system errors. The actual assertion of SERR# is enabled through the PCI Command register.

Table 200. Offset 8Ah: D31_ERR_STS—Device 31 Error Status Register (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	8Ah	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SERR_RTA: SERR# Due to Received Target Abort	0 = Software clears this bit by writing a 1 to the bit location. 1 = The Intel® 6300ESB ICH sets this bit when it receives a target abort. When SERR_EN, the Intel® 6300ESB ICH will also generate an SERR# when SERR_RTA is set.	R/WC
1	SERR_DTT: SERR# Due to Delayed Transaction Timeout	0 = Software clears this bit by writing a 1 to the bit location. 1 = When a PCI master does not return for the data within 1 ms of the cycle's completion, the Intel® 6300ESB ICH clears the delayed transaction and sets this bit. When both SERR_DTT_EN and SERR_EN are set, then Intel® 6300ESB ICH will also generate an SERR# when SERR_DTT is set.	R/WC
0	Reserved	Reserved.	

8.1.21 Offset 90h - 91h: PCI_DMA_CFG—PCI DMA Configuration (LPC I/F—D31:F0)

Note: Since there is no ISA bus, the default is for DMA to be disabled. BIOS must set particular channels for LPC DMA. When ISA is present (through MoonISA), channels not assigned to LPC should be assigned as Disabled.

Table 201. Offset 90h - 91h: PCI_DMA_CFG—PCI DMA Configuration (LPC I/F—D31:F0) (Sheet 1 of 2)

Device:	31	**Function:**	0
Offset:	90h-91h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:14	Channel 7 Select	00 = Reserved 01 = Reserved 10 = Reserved 11 = LPC I/F DMA	R/W
13:12	Channel 6 Select	Same bit decode as for Channel 7.	R/W
11:10	Channel 5 Select	Same bit decode as for Channel 7.	R/W
9:8	Reserved	Reserved.	
7:6	Channel 3 Select	Same bit decode as for Channel 7.	R/W

Table 201. Offset 90h - 91h: PCI_DMA_CFG—PCI DMA Configuration (LPC I/F—D31:F0) (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	90h-91h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
5:4	Channel 2 Select	Same bit decode as for Channel 7.	R/W
3:2	Channel 1 Select	Same bit decode as for Channel 7.	R/W
1:0	Channel 0 Select	Same bit decode as for Channel 7.	R/W

8.1.22 Offset D0h - D3h: GEN_CNTL—General Control Register (LPC I/F—D31:F0)

Table 202. Offset D0h - D3h: GEN_CNTL—General Control Register (LPC I/F—D31:F0) (Sheet 1 of 3)

Device:	31	**Function:**	0
Offset:	D0h - D3h	**Attribute:**	Read/Write
Default Value:	00000080h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:25	Reserved	Reserved.	
24	HIDE_ISA: Hide ISA Bridge	0 = The Intel® 6300ESB ICH will not prevent AD22 from asserting during config cycles to the PCI-to-ISA bridge. 1 = Software sets this bit to 1 to disable config cycle from being claimed by a PCI-to-ISA bridge. This will prevent the OS PCI PnP from getting confused by seeing two ISA bridges. It is required for the Intel® 6300ESB ICH PCI address line AD22 to connect to the PCI-to-ISA bridge's IDSEL input. When this bit is set, the Intel® 6300ESB ICH will not assert AD22 during config cycles to the PCI-to-ISA bridge.	R/W
23:22	Reserved	Reserved.	
21	FERR#-MUX-EN: CPU Break Event Indication Enable	0 = (Default) The Intel® 6300ESB ICH will not examine the FERR# signal during C2, C3 or C4. 1 = Software sets this bit to 1 to enable the Intel® 6300ESB ICH to examine the FERR# signal during a C2, C3 or C4 state as a break event. (See Section 5.11.6, "Dynamic Processor Clock Control" for details.)	R/W
20	Reserved	Reserved.	

Table 202. Offset D0h - D3h: GEN_CNTL—General Control Register (LPC I/F—D31:F0) (Sheet 2 of 3)

Device:	31	**Function:**	0
Offset:	D0h - D3h	**Attribute:**	Read/Write
Default Value:	00000080h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
19:18	SCRATCHPAD	These bits are provided for possible future use.	
17	MMT_ADDR_EN	When set to 1, this bit enables the Intel® 6300ESB ICH to decode the High Performance Event Timer Memory Address Range selected by bits 16:15 below.	
16:15	MMT_ADDR_SEL	This 2-bit field selects 1 of 4 possible memory address ranges for the High Performance Event Timer functionality. The encodings are: **Bits [16:15] — Memory Address Range** 00 — FED0_0000h - FED0_03FFh 01 — FED0_1000h - FED0_13FFh 10 — FED0_2000h - FED0_23FFh 11 — FED0_3000h - FED0_33FFh	
14	Reserved	Reserved.	
13	COPR_ERR_EN: Coprocessor Error Enable	0 = FERR# will not generate IRQ13 nor IGNNE#. 1 = When FERR# is low, the Intel® 6300ESB ICH generates IRQ13 internally and holds it until an I/O write to port F0h. It will also drive IGNNE# active.	R/W
12	IRQ1LEN: Keyboard IRQ1 Latch Enable	0 = IRQ1 will bypass the latch. 1 = The active edge of IRQ1 will be latched and held until a port 60h read.	R/W
11	IRQ12LEN: Mouse IRQ12 Latch Enable	0 = IRQ12 will bypass the latch. 1 = The active edge of IRQ12 will be latched and held until a port 60h read.	R/W
10:9	Reserved	Reserved.	
8	APIC_EN: APIC Enable	0 = Disables internal I/O (x) APIC. 1 = Enables the internal I/O (x) APIC and its address decode.	R/W
7	Reserved	Reserved.	
6	ALTACC_EN: Alternate Access Mode Enable	0 = Alt Access Mode Disabled (default). ALT access mode allows reads to otherwise unreadable registers and writes otherwise unwritable registers. 1 = Alt Access Mode Enable.	R/W
5:3	Reserved	Reserved.	

Table 202. Offset D0h - D3h: GEN_CNTL—General Control Register (LPC I/F—D31:F0) (Sheet 3 of 3)

Device:	31	**Function:**	0
Offset:	D0h - D3h	**Attribute:**	Read/Write
Default Value:	00000080h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	DCB_EN: DMA Collection Buffer Enable	0 = DCB disabled. 1 = Enables DMA Collection Buffer (DCB) for LPC I/F and PC/PCI DMA.	R/W
1	DTE: Delayed Transaction Enable	0 = Delayed transactions disabled. 1 = The Intel® 6300ESB ICH enables delayed transactions for internal register, FWH and LPC I/F accesses.	R/W
0	POS_DEC_EN	0 = The Intel® 6300ESB ICH will perform subtractive decode on the PCI bus and forward the cycles to LPC if not to an internal register or other known target on LPC. Accesses to internal registers and to known LPC devices will still be positively decoded. 1 = Enables Intel® 6300ESB ICH to only perform positive decode on the PCI bus. This must be selected when the PCI to ISA (subtractive docking bridge) is used.	

8.1.23 Offset D4h: GEN_STA—General Status (LPC I/F—D31:F0)

Table 203. Offset D4h: GEN_STA—General Status (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	D4h	**Attribute:**	Read/Write
Default Value:	0Xh	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SAFE_MODE	0 = The Intel® 6300ESB ICH sampled AC_SDOUT low on the rising edge of PWROK. 1 = The Intel® 6300ESB ICH sampled AC_SDOUT high on the rising edge of PWROK.	RO
1	NO_REBOOT	0 = Normal TCO Timer reboot functionality (reboot after 2nd TCO timeout). This bit cannot be set to 0 by software when the strap is set to No Reboot. 1 = The Intel® 6300ESB ICH will disable the TCO Timer system reboot feature. This bit is set either by hardware when SPKR is sampled high on the rising edge of PWROK, or by software writing a 1 to the bit.	R/W (special)
0	Reserved	Reserved.	

8.1.24 Offset D5h: BACK_CNTL—Backed Up Control (LPC I/F—D31:F0)

Table 204. Offset D5h: BACK_CNTL—Backed Up Control (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	D5h	**Attribute:**	Read/Write
Default Value:	0Fh (upon RTCRST# assertion low) 2Fh (TOP_SWAP strap active)	**Size:**	8-bit
Lockable:	No	**Power Well:**	RTC

Bits	Name	Description	Access
7	Reserved	Reserved.	RO
6	Reserved	Reserved.	RO
5	TOP_SWAP: Top-Block Swap Mode	0 = The Intel® 6300ESB ICH will not invert A16. This bit is automatically set to 0 by RTCRST#, but not by any other type of reset. 1 = The Intel® 6300ESB ICH will invert A16 for cycles going to the BIOS space in the FWH (but not for cycles to the Feature Space in the FWH). **NOTE:** If the Intel® 6300ESB ICH is strapped for Top-Swap (SIU0_DTR# is low at rising edge of PWROK), then this bit CANNOT be cleared by software. The strap jumper should be removed and the system rebooted.	R/W
4	CPU_BIST_EN: Enables CPU BIST	0 = Disable. 1 = The INIT# signal will be driven active when CPURST# is active. INIT# will go inactive with the same timings as the other CPU I/F signals (Hold Time after CPURST# inactive). Note that CPURST# is generated by the memory controller hub, but the Intel® 6300ESB ICH has a Hub Interface special cycle that allows the Intel® 6300ESB ICH to control the assertion/deassertion of CPURST#. **NOTE:** This bit is in the Resume well and is reset by RSMRST#, but not by PXPCIRST# nor CF9h writes.	R/W
3:0	FREQ_STRAP[3:0]: CPU Frequency Strap	These bits determine the internal frequency multiplier of the processor. These bits may be reset to 1111 based on an external pin strap or through the RTCRST# input signal. Software must program this field based on the processor's specified frequency. Note that this field is only writable when the SAFE_MODE bit is cleared to zero, and SAFE_MODE is only cleared by PWROK rising edge. These bits are in the RTC well.	R/W

8.1.25 Offset D8h: RTC_CONF—RTC Configuration Register (LPC I/F—D31:F0)

Table 205. Offset D8h: RTC_CONF—RTC Configuration Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	D8h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	Yes	***Power Well:***	Core

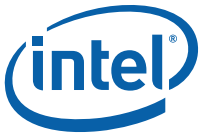
Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	U128LOCK: Upper 128-byte Lock	0 = Access to these bytes in the upper CMOS RAM range have not been locked. 1 = Locks reads and writes to bytes 38h-3Fh in the upper 128-byte bank of the RTC CMOS RAM. Write cycles to this range will have no effect and read cycles will not return any particular ensured value. This is a write once register that may only be reset by a hardware reset.	R/W (special)
3	L128LOCK: Lower 128-byte Lock	0 = Access to these bytes in the lower CMOS RAM range have not been locked. 1 = Locks reads and writes to bytes 38h-3Fh in the lower 128-byte bank of the RTC CMOS RAM. Write cycles to this range will have no effect and read cycles will not return any particular ensured value. This is a write once register that may only be reset by a hardware reset.	R/W (special)
2	U128E: Upper 128-byte Enable	0 = Disable. 1 = Enables access to the upper 128-byte bank of RTC CMOS RAM.	R/W
1:0	Reserved	Reserved.	

8.1.26 Offset E0h: COM_DEC—LPC I/F Communication Port Decode Ranges (LPC I/F—D31:F0)

Table 206. Offset E0h: COM_DEC—LPC I/F Communication Port Decode Ranges (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	E0h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7	Reserved	Reserved.	
6:4	COMB Decode Range	This field determines which range to decode for the COMB Port. **Bits** — **Decode Range** 000 — 3F8h - 3FFh (COM1) 001 — 2F8h - 2FFh (COM2) 010 — 220h - 227h 011 — 228h - 22Fh 100 — 238h - 23Fh 101 — 2E8h - 2EFh (COM4) 110 — 338h - 33Fh 111 — 3E8h - 3EFh (COM3)	R/W
3	Reserved	Reserved.	
2:0	COMA Decode Range	This field determines which range to decode for the COMA Port. **Bits** — **Decode Range** 000 — 3F8h - 3FFh (COM1) 001 — 2F8h - 2FFh (COM2) 010 — 220h - 227h 011 — 228h - 22Fh 100 — 238h - 23Fh 101 — 2E8h - 2EFh (COM4) 110 — 338h - 33Fh 111 — 3E8h - 3EFh (COM3)	R/W



8.1.27 Offset E1h: FDD/LPT_DEC—LPC I/F FDD and LPT Decode Ranges (LPC I/F—D31:F0)

Table 207. Offset E1h: FDD/LPT_DEC—LPC I/F FDD and LPT Decode Ranges (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	E1h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	FDD Decode Range	Determines which range to decode for the FDD Port 0 = 3F0h - 3F5h, 3F7h (Primary) 1 = 370h - 2FFh (Secondary)	R/W
3:2	Reserved	Reserved.	
1:0	LPT Decode Range	This field determines which range to decode for the LPTPort. 00 = 378h - 37Fh and 778h - 77Fh 01 = 278h - 27Fh (port 279h is read only) and 678h - 67Fh 10 = 3BCh -3BEh and 7BCh - 7BEh 11 = Reserved	R/W

8.1.28 Offset E2h: SND_DEC—LPC I/F Sound Decode Ranges (LPC I/F—D31:F0)

Note: This register is no longer supported and will not be validated.

Table 208. Offset E2h: SND_DEC—LPC I/F Sound Decode Ranges (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	E2h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:6	Reserved	Reserved.	
5:4	MSS Decode Range	This field determines which range to decode for the Microsoft* Sound System (MSS) 00 = 530h - 537h 01 = 604h - 60Bh 10 = E80h - E87h 11 = F40h - F47h	R/W
3	MIDI Decode Range	This bit determines which range to decode for the Midi Port. 0 = 330h - 331h 1 = 300h - 301h	R/W
2	Reserved	Reserved.	
1:0	SB16 Decode Range	This field determines which range to decode for the Sound Blaster 16 (SB16) Port. 00 = 220h - 233h 01 = 240h - 253h 10 = 260h - 273h 11 = 280h - 293h	R/W

8.1.29 Offset E3h: FWH_DEC_EN1—FWH Decode Enable 1 Register (LPC I/F—D31:F0)

Note: This register determines which memory ranges will be decoded on the PCI bus and forwarded to the FWH. The Intel® 6300ESB ICH will subtractively decode cycles on PCI unless POS_DEC_EN is set to 1.

Table 209. Offset E3h: FWH_DEC_EN1—FWH Decode Enable 1 Register (LPC I/F—D31:F0) (Sheet 1 of 2)

Device:	31	**Function:**	0
Offset:	E3h	**Attribute:**	Read/Write
Default Value:	FFh	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7	FWH_F8_EN	Enables decoding two 512 Kbyte FWH memory ranges, and one 128 Kbyte memory range. 0 = Disable 1 = Enable the following ranges for the FWH FFF80000h - FFFFFFFFh FFB80000h - FFBFFFFFh 000E0000h - 000FFFFFh	RO
6	FWH_F0_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH: FFF00000h - FFF7FFFFh FFB00000h - FFB7FFFFh	R/W
5	FWH_E8_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH: FFE80000h - FFEFFFFFh FFA80000h - FFAFFFFFh	R/W
4	FWH_E0_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH: FFE00000h - FFE7FFFFh FFA00000h - FFA7FFFFh	R/W
3	FWH_D8_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FFD80000h - FFDFFFFFh FF980000h - FF9FFFFFh	R/W

Table 209. Offset E3h: FWH_DEC_EN1—FWH Decode Enable 1 Register (LPC I/F—D31:F0) (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	E3h	**Attribute:**	Read/Write
Default Value:	FFh	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	FWH_D0_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FFD00000h - FFD7FFFFh FF900000h - FF97FFFFh	R/W
1	FWH_C8_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FFC80000h - FFCFFFFFh FF880000h - FF8FFFFFh	R/W
0	FWH_C0_EN	Enables decoding two 512 Kbyte FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FFC00000h - FFC7FFFFh FF800000h - FF87FFFFh	R/W

8.1.30 Offset E4h - E5h: GEN1_DEC—LPC I/F Generic Decode Range 1 (LPC I/F—D31:F0)

Table 210. Offset E4h - E5h: GEN1_DEC—LPC I/F Generic Decode Range 1 (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	E4h - E5h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	16-bit
Lockable:	Yes	**Power Well:**	Core

Bits	Name	Description	Access
15:7	GEN1_BASE: Generic I/O Decode Range 1 Base Address	This address is aligned on a 128-byte boundary, and must have address lines 31:16 as 0. Note that this generic decode is for I/O addresses only, not memory addresses. The size of this range is 128 bytes.	R/W
6:1	Reserved	Reserved.	
0	GEN1_EN: Generic Decode Range 1 Enable	0 = Disable. 1 = Enable the GEN1 I/O range to be forwarded to the LPC I/F.	R/W

8.1.31 Offset E6h - E7h: LPC_EN—LPC I/F Enables (LPC I/F—D31:F0)

Table 211. Offset E6h - E7h: LPC_EN—LPC I/F Enables (LPC I/F—D31:F0) (Sheet 1 of 2)

Device:	31	***Function:***	0
Offset:	E6h - E7h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	Yes	***Power Well:***	Core

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13	CNF2_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 4Eh and 4Fh to the LPC interface. This is used for the internal SIU.	R/W
12	CNF1_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 2Eh and 2Fh to the LPC interface. This range is used for Super I/O devices.	R/W
11	MC_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 62h and 66h to the LPC interface. This range is used for a microcontroller.	R/W
10	KBC_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 60h and 64h to the LPC interface. This range is used for a microcontroller.	R/W
9	GAMEH_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 208h to 20Fh to the LPC interface. This range is used for a gameport.	R/W
8	GAMEL_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 200h to 207h to the LPC interface. This range is used for a gameport.	R/W
7	ADLIB_LPC_EN	0 = Disable. 1 = Enables the decoding of the I/O locations 388h - 38Bh to the LPC interface. **NOTE:** This bit is no longer supported and will not be validated.	R/W
6	MSS_LPC_EN	0 = Disable. 1 = Enables the decoding of the MSS range to the LPC interface. This range is selected in the LPC_Sound Decode Range Register. **NOTE:** This bit is no longer supported and will not be validated.	R/W
5	MIDI_LPC_EN	0 = Disable. 1 = Enables the decoding of the MIDI range to the LPC interface. This range is selected in the LPC_Sound Decode Range Register. **NOTE:** This bit is no longer supported and will not be validated.	R/W

Table 211. Offset E6h - E7h: LPC_EN—LPC I/F Enables (LPC I/F—D31:F0) (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	E6h - E7h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	Yes	**Power Well:**	Core

Bits	Name	Description	Access
4	SB16_LPC_EN	0 = Disable. 1 = Enables the decoding of the SB16 range to the LPC interface. This range is selected in the LPC_Sound Decode Range Register. **NOTE:** This bit is no longer supported and will not be validated.	R/W
3	FDD_LPC_EN	0 = Disable. 1 = Enables the decoding of the FDD range to the LPC interface. This range is selected in the LPC_FDD/LPT Decode Range Register.	R/W
2	LPT_LPC_EN	0 = Disable. 1 = Enables the decoding of the LPTrange to the LPC interface. This range is selected in the LPC_FDD/LPT Decode Range Register.	R/W
1	COMB_LPC_EN	0 = Disable. 1 = Enables the decoding of the COMB range to the LPC interface. This range is selected in the LPC_COM Decode Range Register.	R/W
0	COMA_LPC_EN	0 = Disable. 1 = Enables the decoding of the COMA range to the LPC interface. This range is selected in the LPC_COM Decode Range Register.	R/W

8.1.32 Offset E8h: FWH_SEL1—FWH Select 1 Register (LPC I/F—D31:F0)

Table 212. Offset E8h: FWH_SEL1—FWH Select 1 Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	E8h	***Attribute:***	Read/Write
Default Value:	00112233h	***Size:***	32-bit

Bits	Name	Description	Access
31:28	FWH_F8_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges and one 128 Kbyte memory range. This field is fixed at 0000. The IDSEL programmed in this field addresses the following memory ranges: FFF8 0000h - FFFF FFFFh FFB8 0000h - FFBF FFFFh 000E 0000h - 000F FFFFh	RO
27:24	FWH_F0_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFF0 0000h - FFF7 FFFFh FFB0 0000h - FFB7 FFFFh	R/W
23:20	FWH_E8_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFE8 0000h - FFEF FFFFh FFA8 0000h - FFAF FFFFh	R/W
19:16	FWH_E0_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFE0 0000h - FFE7 FFFFh FFA0 0000h - FFA7 FFFFh	R/W
15:12	FWH_D8_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFD8 0000h - FFDF FFFFh FF98 0000h - FF9F FFFFh	R/W
11:8	FWH_D0_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFD0 0000h - FFD7 FFFFh FF90 0000h - FF97 FFFFh	R/W
7:4	FWH_C8_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFC8 0000h - FFCF FFFFh FF88 0000h - FF8F FFFFh	R/W
3:0	FWH_C0_IDSEL	IDSEL for two 512 Kbyte FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FFC0 0000h - FFC7 FFFFh FF80 0000h - FF87 FFFFh	R/W

8.1.33 Offset ECh - EDh: GEN2_DEC—LPC I/F Generic Decode Range 2 (LPC I/F—D31:F0)

Table 213. Offset ECh - EDh: GEN2_DEC—LPC I/F Generic Decode Range 2 (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	ECh - EDh	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	16-bit
Lockable:	Yes	**Power Well:**	Core

Bits	Name	Description	Access
15:4	GEN2_BASE	Generic I/O Decode Range 2 Base Address. This address is aligned on a 64-byte boundary, and must have address lines 31:16 as 0. Note that this generic decode is for I/O addresses only, not memory addresses. The size of this range is 16 bytes.	R/W
3:1	Reserved	Reserved. Read as 0.	
0	GEN2_EN	Generic I/O Decode Range 2 Enable 0 = Disable. 1 = Accesses to the GEN2 I/O range will be forwarded to the LPC I/F.	R/W

8.1.34 Offset EEh - EFh: FWH_SEL2—FWH Select 2 Register (LPC I/F—D31:F0)

Table 214. Offset EEh - EFh: FWH_SEL2—FWH Select 2 Register (LPC I/F—D31:F0)

Device:	31	**Function:**	0
Offset:	EEh-EFh	**Attribute:**	Read/Write
Default Value:	4567h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:12	FWH_70_IDSEL	IDSEL for two 1M FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FF70 0000h - FF7F FFFFh FF30 0000h - FF3F FFFFh	R/W
11:8	FWH_60_IDSEL	IDSEL for two 1M FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FF60 0000h - FF6F FFFFh FF20 0000h - FF2F FFFFh	R/W
7:4	FWH_50_IDSEL	IDSEL for two 1M FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FF50 0000h - FF5F FFFFh FF10 0000h - FF1F FFFFh	R/W
3:0	FWH_40_IDSEL	IDSEL for two 1M FWH memory ranges. The IDSEL programmed in this field addresses the following memory ranges: FF40 0000h - FF4F FFFFh FF00 0000h - FF0F FFFFh	R/W

8.1.35 Offset F0h: FWH_DEC_EN2—FWH Decode Enable 2 Register (LPC I/F—D31:F0)

Note: This register determines which memory ranges will be decoded on the PCI bus and forwarded to the FWH. The Intel® 6300ESB ICH will subtractively decode cycles on PCI unless POS_DEC_EN is set to 1.

Table 215. Offset F0h: FWH_DEC_EN2—FWH Decode Enable 2 Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	F0h	***Attribute:***	Read/Write
Default Value:	0Fh	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3	FWH_70_EN	Enables decoding two 1M FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FF70 0000h - FF7F FFFFh FF30 0000h - FF3F FFFFh	R/W
2	FWH_60_EN	Enables decoding two 1M FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FF60 0000h - FF6F FFFFh FF20 0000h - FF2F FFFFh	R/W
1	FWH_50_EN	Enables decoding two 1M FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FF50 0000h - FF5F FFFFh FF10 0000h - FF1F FFFFh	R/W
0	FWH_40_EN	Enables decoding two 1M FWH memory ranges. 0 = Disable. 1 = Enable the following ranges for the FWH FF40 0000h - FF4F FFFFh FF00 0000h - FF0F FFFFh	R/W

8.1.36 Offset F2h: FUNC_DIS—Function Disable Register (LPC I/F—D31:F0)

Note: The USB functions must be disabled from highest function number to lowest. For example, if only 2 USB host controllers are wanted, software must disable Function #2. USB functions are expected to only be disabled by BIOS during system initialization.

Table 216. Offset F2h: FUNC_DIS—Function Disable Register (LPC I/F—D31:F0) (Sheet 1 of 2)

Device:	31	***Function:***	0
Offset:	F2h	***Attribute:***	Read/Write
Default Value:	0080h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15	D29_F7_Disable	Software sets this bit to disable the USB EHCI Controller function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = USB EHCI Controller is enabled 1 = USB EHCI Controller is disabled	R/W
14:10	Reserved	Reserved.	
9	D29_F1_Disable	Software sets this bit to disable the USB UHCI Controller #2 function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = USB UHCI Controller #2 is enabled 1 = USB UHCI Controller #2 is disabled	R/W
8	D29_F0_Disable	Software sets this bit to disable the USB UHCI Controller #1 function.BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = USB UHCI Controller #1 is enabled 1 = USB UHCI Controller #1 is disabled	R/W
7	Reserved	Reserved	
6	D31_F6_Disable	Software sets this bit to disable the AC'97 modem controller function. When disabled, the PCI config space registers for that function are not decoded by the Intel® 6300ESB ICH. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = AC'97 Modem is enabled 1 = AC'97 Modem is disabled	R/W
5	D31_F5_Disable	Software sets this bit to disable the AC'97 audio controller function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = AC'97 audio controller is enabled 1 = AC'97 audio controller is disabled	R/W
4	Reserved	Reserved.	

Table 216. Offset F2h: FUNC_DIS—Function Disable Register (LPC I/F—D31:F0) (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	F2h	**Attribute:**	Read/Write
Default Value:	0080h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
3	D31_F3_Disable	Software sets this bit to disable the SMBus Host Controller function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = SMBus controller is enabled 1 = SMBus controller is disabled	R/W
2	D31_F2_Disable	Software sets this bit to disable the SATA Controller function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = SATA controller is enabled 1 = SATA controller is disabled	R/W
1	D31_F1_Disable	Software sets this bit to disable the IDE controller function. BIOS must not enable I/O or memory address space decode, interrupt generation, or any other functionality of functions that are to be disabled. 0 = IDE controller is enabled 1 = IDE controller is disabled	R/W
0	SMB_FOR_BIOS	This bit is used in conjunction with bit 3 in this register. 0 = No effect. 1 = Allows the SMBus I/O space to be accessible by software when bit 3 in this register is set. The PCI configuration space is hidden in this case. Note that when bit 3 is set alone, the decode of both SMBus PCI configuration and I/O space will be disabled.	R/W

NOTE: Software must always disable all functionality within the function before disabling the configuration space.

8.1.37 Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)

Warning: Make sure that reserved bits values are not modified to avoid indeterminate behavior.

Table 217. Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)

Device:	31	***Function:***	0
Offset:	F4-f7h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:8	Reserved	Reserved.	
7	Trapping Disable	Default = 0. 0 = Trapping is enabled for all access to these ports. See Section 5.17.9, "USB Legacy Keyboard Operation" for details on USB Legacy Keyboard Operation. 1 = Disable the USB Legacy Trapping to Ports 60h and 64h when an external PCI Master is accessing these ports.	R/W
6	PXIRQ Routing	Default = 0. 0 = APIC1 boot interrupt is routed to IRQ9#. See Section 5.7.3, "Boot Interrupt" for details on Boot Interrupt. 1 = Routes the APIC1 boot interrupt to the PIRQG# output.	R/W
5:0	Reserved	Reserved.	

8.1.38 Offset F8h: Manufacturer's ID

Table 218. Offset F8h: Manufacturer's ID

Device:	31	***Function:***	0
Offset:	F8h-FBh	***Attribute:***	Read-Only
Default Value:	000S 0F66	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:8	Manufacturer	0Fh = Intel	
7:0	Process/Dot	Process 859.6	

8.2 DMA I/O Registers

Table 219. DMABASE_CA—DMA Base and Current Address Registers (Sheet 1 of 2)

Port	Alias	Register Name/Function	Default	Type
00h	10h	Channel 0 DMA Base and Current Address Register	Undefined	R/W
01h	11h	Channel 0 DMA Base and Current Count Register	Undefined	R/W
02h	12h	Channel 1 DMA Base and Current Address Register	Undefined	R/W
03h	13h	Channel 1 DMA Base and Current Count Register	Undefined	R/W
04h	14h	Channel 2 DMA Base and Current Address Register	Undefined	R/W
05h	15h	Channel 2 DMA Base and Current Count Register	Undefined	R/W
06h	16h	Channel 3 DMA Base and Current Address Register	Undefined	R/W
07h	17h	Channel 3 DMA Base and Current Count Register	Undefined	R/W
08h	18h	Channel 0-3 DMA Command Register	Undefined	WO
		Channel 0-3 DMA Status Register	Undefined	RO
0Ah	1Ah	Channel 0-3 DMA Write Single Mask Register	000001XXb	WO
0Bh	1Bh	Channel 0-3 DMA Channel Mode Register	000000XXb	WO
0Ch	1Ch	Channel 0-3 DMA Clear Byte Pointer Register	Undefined	WO
0Dh	1Dh	Channel 0-3 DMA Master Clear Register	Undefined	WO
0Eh	1Eh	Channel 0-3 DMA Clear Mask Register	Undefined	WO
0Fh	1Fh	Channel 0-3 DMA Write All Mask Register	0Fh	R/W
80h	90h	Reserved Page Register	Undefined	R/W
81h	91h	Channel 2 DMA Memory Low Page Register	Undefined	R/W
82h	-	Channel 3 DMA Memory Low Page Register	Undefined	R/W
83h	93h	Channel 1 DMA Memory Low Page Register	Undefined	R/W
84h - 86h	94h - 96h	Reserved Page Registers	Undefined	R/W
87h	97h	Channel 0 DMA Memory Low Page Register	Undefined	R/W
88h	98h	Reserved Page Register	Undefined	R/W
89h	99h	Channel 6 DMA Memory Low Page Register	Undefined	R/W
8Ah	9Ah	Channel 7 DMA Memory Low Page Register	Undefined	R/W
8Bh	9Bh	Channel 5 DMA Memory Low Page Register	Undefined	R/W
8Ch - 8Eh	9Ch - 9Eh	Reserved Page Registers	Undefined	R/W
8Fh	9Fh	Refresh Low Page Register	Undefined	R/W
C0h	C1h	Channel 4 DMA Base and Current Address Register	Undefined	R/W

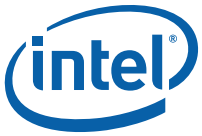


Table 219. DMABASE_CA—DMA Base and Current Address Registers (Sheet 2 of 2)

Port	Alias	Register Name/Function	Default	Type
C2h	C3h	Channel 4 DMA Base and Current Count Register	Undefined	R/W
C4h	C5h	Channel 5 DMA Base and Current Address Register	Undefined	R/W
C6h	C7h	Channel 5 DMA Base and Current Count Register	Undefined	R/W
C8h	C9h	Channel 6 DMA Base and Current Address Register	Undefined	R/W
CAh	CBh	Channel 6 DMA Base and Current Count Register	Undefined	R/W
CCh	CDh	Channel 7 DMA Base and Current Address Register	Undefined	R/W
CEh	CFh	Channel 7 DMA Base and Current Count Register	Undefined	R/W
D0h	D1h	Channel 4-7 DMA Command Register	Undefined	WO
		Channel 4-7 DMA Status Register	Undefined	RO
D4h	D5h	Channel 4-7 DMA Write Single Mask Register	000001XXb	WO
D6h	D7h	Channel 4-7 DMA Channel Mode Register	000000XXb	WO
D8h	D9h	Channel 4-7 DMA Clear Byte Pointer Register	Undefined	WO
DAh	DBh	Channel 4-7 DMA Master Clear Register	Undefined	WO
DCh	DDh	Channel 4-7 DMA Clear Mask Register	Undefined	WO
DEh	DFh	Channel 4-7 DMA Write All Mask Register	0Fh	R/W

8.2.1 DMABASE_CA—DMA Base and Current Address Registers

Table 220. DMABASE_CA—DMA Base and Current Address Registers

Device:	31	**Function:**	0
I/O Address:	Ch. #0 = 00h; Ch. #1 = 02h, Ch. #2 = 04h; Ch. #3 = 06h, Ch. #5 = C4h Ch. #6 = C8h, Ch. #7 = CCh	**Attribute:**	Read-Only
Default Value:	Undefined	**Size:**	16-bit, accessed in two 8-bit quantities
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Base and Current Address	This register determines the address for the transfers to be performed. The address specified points to two separate registers. On writes, the value is stored in the *Base Address* register and copied to the *Current Address* register. On reads, the value is returned from the *Current Address* register. The address increments/decrements in the Current Address register after each transfer, depending on the mode of the transfer. When the channel is in auto-initialize mode, the Current Address register will be reloaded from the Base Address register after a terminal count is generated. For transfers to/from a 16-bit slave (channel's 5-7), the address is shifted left one bit location. Bit 15 will be shifted into Bit 16. The register is accessed in 8 bit quantities. The byte is pointed to by the current byte pointer flip/flop. Before accessing an address register, the byte pointer flip/flop should be cleared to ensure that the low byte is accessed first	R/W

8.2.2 DMABASE_CC—DMA Base and Current Count Registers

Table 221. DMABASE_CC—DMA Base and Current Count Registers

Device:	31	***Function:***	0
I/O Address:	Ch. #0 = 01h; Ch. #1 = 03h, Ch. #2 = 05h; Ch. #3 = 07h, Ch. #5 = C6h Ch. #6 = CAh, Ch. #7 = CEh	***Attribute:***	Read/Write
Default Value:	Undefined	***Size:***	16-bit, accessed in two 8-bit quantities
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Base and Current Count	This register determines the number of transfers to be performed. The address specified points to two separate registers. On writes, the value is stored in the *Base Count* register and copied to the *Current Count* register. On reads, the value is returned from the *Current Count* register. The actual number of transfers is one more than the number programmed in the Base Count Register (i.e., programming a count of 4h results in 5 transfers). The count is decrements in the Current Count register after each transfer. When the value in the register rolls from zero to FFFFh, a terminal count is generated. When the channel is in auto-initialize mode, the Current Count register will be reloaded from the Base Count register after a terminal count is generated. For transfers to/from an 8-bit slave (channels 0-3), the count register indicates the number of bytes to be transferred. For transfers to/from a 16-bit slave (channels 5-7), the count register indicates the number of words to be transferred. The register is accessed in 8 bit quantities. The byte is pointed to by the current byte pointer flip/flop. Before accessing a count register, the byte pointer flip/flop should be cleared to ensure that the low byte is accessed first.	R/W

8.2.3 DMAMEM_LP—DMA Memory Low Page Registers

Table 222. DMABASE_CC—DMA Base and Current Count Registers

Device:	31	**Function:**	0
I/O Address:	Ch. #0 = 87h; Ch. #1 = 83h, Ch. #2 = 81h; Ch. #3 = 82h, Ch. #5 = 8Bh Ch. #6 = 89h, Ch. #7 = 8Ah	**Attribute:**	Read/Write
Default Value:	Undefined	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	DMA Low Page (ISA Address bits [23:16]	This register works in conjunction with the DMA controller's Current Address Register to define the complete 24-bit address for the DMA channel. This register remains static throughout the DMA transfer. Bit 16 of this register is ignored when in 16 bit I/O count by words mode as it is replaced by the bit 15 shifted out from the current address register.	R/W

8.2.4 DMACMD—DMA Command Register

Table 223. DMACMD—DMA Command Register

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 08h Ch. #4-7 = D0h	**Attribute:**	Write-Only
Default Value:	Undefined	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved. Must be 0.	
4	DMA Group Arbitration Priority	Each channel group is individually assigned either fixed or rotating arbitration priority. At part reset, each group is initialized in fixed priority. 0 = Fixed priority to the channel group 1 = Rotating priority to the group.	WO
3	Reserved	Reserved. Must be zero.	
2	DMA Channel Group Enable	Both channel groups are enabled following part reset. 0 = Enable the DMA channel group. 1 = Disable. Disabling channel group 4-7 also disables channel group 0-3, which is cascaded through channel 4.	WO
1:0	Reserved	Reserved. Must be zero.	

8.2.5 DMASTA—DMA Status Register

Table 224. DMASTA—DMA Status Register

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 08h Ch. #4-7 = D0h	**Attribute:**	Read-Only
Default Value:	Undefined	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:4	Channel Request Status	When a valid DMA request is pending for a channel, the corresponding bit is set to 1. When a DMA request is not pending for a particular channel, the corresponding bit is set to 0. The source of the DREQ may be hardware or a software request. Note that channel 4 is the cascade channel, so the request status of channel 4 is a logical OR of the request status for channels 0 through 3. 4 = Channel 0 5 = Channel 1 (5) 6 = Channel 2 (6) 7 = Channel 3 (7)	RO
3:0	Channel Terminal Count Status	When a channel reaches terminal count (TC), its status bit is set to 1. When TC has not been reached, the status bit is set to 0. Channel 4 is programmed for cascade, so the TC bit response for channel 4 is irrelevant: 0 = Channel 0 1 = Channel 1 (5) 2 = Channel 2 (6) 3 = Channel 3 (7)	RO

8.2.6 DMA_WRSMSK—DMA Write Single Mask Register

Table 225. DMA_WRSMSK—DMA Write Single Mask Register

Device:	31	***Function:***	0
I/O Address:	Ch. #0-3 = 0Ah Ch. #4-7 = D4h	***Attribute:***	Write-Only
Default Value:	0000 01xx	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved. Must be zero.	
2	Channel Mask Select	0 = Enable DREQ for the selected channel. The channel is selected through bits [1:0]. Therefore, only one channel may be masked / unmasked at a time. 1 = Disable DREQ for the selected channel.	WO
1:0	DMA Channel Select	These bits select the DMA Channel Mode Register to program. 00 = Channel 0 (4) 01 = Channel 1 (5) 10 = Channel 2 (6) 11 = Channel 3 (7)	WO

8.2.7 DMACH_MODE—DMA Channel Mode Register

Table 226. DMACH_MODE—DMA Channel Mode Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	Ch. #0-3 = 0Bh Ch. #4-7 = D6h	***Attribute:***	Write-Only
Default Value:	0000 00xx	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:6	DMA Transfer Mode	Each DMA channel may be programmed in one of four different modes: 00 = Demand mode 01 = Single mode 10 = Reserved 11 = Cascade mode	WO
5	Address Increment/ Decrement Select	This bit controls address increment/decrement during DMA transfers. 0 = Address increment. (default after part reset or Master Clear) 1 = Address decrement.	WO

Table 226. DMACH_MODE—DMA Channel Mode Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 0Bh Ch. #4-7 = D6h	**Attribute:**	Write-Only
Default Value:	0000 00xx	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
4	Autoinitialize Enable	0 = Autoinitialize feature is disabled and DMA transfers terminate on a terminal count. A part reset or Master Clear disables autoinitialization. 1 = DMA restores the Base Address and Count registers to the current registers following a terminal count (TC).	WO
3:2	DMA Transfer Type	These bits represent the direction of the DMA transfer. When the channel is programmed for cascade mode, (bits[7:6] = “11”) the transfer type is irrelevant. 00 = Verify - No I/O or memory strobes generated 01 = Write - Data transferred from the I/O devices to memory 10 = Read - Data transferred from memory to the I/O device 11 = Illegal	WO
1:0	DMA Channel Select	These bits select the DMA Channel Mode Register that will be written by bits [7:2]. 00 = Channel 0 (4) 01 = Channel 1 (5) 10 = Channel 2 (6) 11 = Channel 3 (7)	WO

8.2.8 DMA Clear Byte Pointer Register

Table 227. DMA Clear Byte Pointer Register

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 0Ch Ch. #4-7 = D8h	**Attribute:**	Write-Only
Default Value:	xxxx xxxx	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Clear Byte Pointer	No specific pattern. Command enabled with a write to the I/O port address. Writing to this register initializes the byte pointer flip/flop to a known state. It clears the internal latch used to address the upper or lower byte of the 16-bit Address and Word Count Registers. The latch is also cleared by part reset and by the Master Clear command. This command precedes the first access to a 16-bit DMA controller register. The first access to a 16-bit register will then access the significant byte, and the second access automatically accesses the most significant byte.	WO

8.2.9 DMA Master Clear Register

Table 228. DMA Master Clear Register

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 0Dh Ch. #4-7 = DAh	**Attribute:**	Write-Only
Default Value:	xxxx xxxx	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Master Clear	No specific pattern. Enabled with a write to the port. This has the same effect as the hardware Reset. The Command, Status, Request, and Byte Pointer flip/flop registers are cleared and the Mask Register is set.	WO

8.2.10 DMA_CLMSK—DMA Clear Mask Register

Table 229. DMA_CLMSK—DMA Clear Mask Register

Device:	31	**Function:**	0
I/O Address:	Ch. #0-3 = 0Eh; Ch. #4-7 = DCh	**Attribute:**	Write-Only
Default Value:	xxxx xxxx	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Clear Mask Register	No specific pattern. Command enabled with a write to the port.	WO

8.2.11 DMA_WRMSK—DMA Write All Mask Register

Table 230. DMA_WRMSK—DMA Write All Mask Register

Device:	31	***Function:***	0
I/O Address:	Ch. #0-3 = 0Fh; Ch. #4-7 = DEh	***Attribute:***	Read/Write
Default Value:	0000 1111	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:4	Reserved	Reserved. Must be 0.	
3:0	Channel Mask Bits	This register permits all four channels to be simultaneously enabled/disabled instead of enabling/disabling each channel individually, as is the case with the Mask Register - Write Single Mask Bit. In addition, this register has a read path to allow the status of the channel mask bits to be read. A channel's mask bit is automatically set to 1 when the Current Byte/Word Count Register reaches terminal count (unless the channel is in auto-initialization mode). Setting the bit(s) to a 1 disables the corresponding DREQ(s). Setting the bit(s) to a 0 enables the corresponding DREQ(s). Bits [3:0] are set to 1 upon part reset or Master Clear. When read, bits [3:0] indicate the DMA channel [3:0] ([7:4]) mask status. Bit 0 = Channel 0 (4)1 = Masked, 0 = Not Masked Bit 1 = Channel 1 (5)1 = Masked, 0 = Not Masked Bit 2 = Channel 2 (6)1 = Masked, 0 = Not Masked Bit 3 = Channel 3 (7)1 = Masked, 0 = Not Masked **NOTE:** Disabling channel 4 also disables channels 0-3 due to the cascade of channel's 0 - 3 through channel 4.	R/W

8.3 Timer I/O Registers

Table 231. Timer I/O Registers

Port	Aliases	Register Name/Function	Default Value	Type
40h	50h	Counter 0 Interval Time Status Byte Format	0XXXXXXXb	RO
		Counter 0 Counter Access Port Register	Undefined	R/W
41h	51h	Counter 1 Interval Time Status Byte Format	0XXXXXXXb	RO
		Counter 1 Counter Access Port Register	Undefined	R/W
42h	52h	Counter 2 Interval Time Status Byte Format	0XXXXXXXb	RO
		Counter 2 Counter Access Port Register	Undefined	R/W
43h	53h	Timer Control Word Register	Undefined	WO
		Timer Control Word Register Read Back	XXXXXXX0b	WO
		Counter Latch Command	X0h	WO

8.3.1 TCW —Timer Control Word Register

Note: This register is programmed prior to any counter being accessed to specify counter modes. Following part reset, the control words for each register are undefined and each counter output is 0. Each timer must be programmed to bring it into a known state.

Note: There are two special commands that may be issued to the counters through this register, the Read Back Command and the Counter Latch Command. When these commands are chosen, several bits within this register are redefined. These register formats are described below.

Table 232. TCW —Timer Control Word Register

Device:	31	**Function:**	0
I/O Address:	43h	**Attribute:**	Write-Only
Default Value:	All bits undefined	**Size:**	8-bit

Bits	Name	Description	Access
7:6	Counter Select	The Counter Selection bits select the counter the control word acts upon as shown below. The Read Back Command is selected when bits[7:6] are both 1. 00 = Counter 0 select 01 = Counter 1 select 10 = Counter 2 select 11 = Read Back Command	WO
5:4	Read/Write Select	These bits are the read/write control bits. The actual counter programming is done through the counter port (40h for counter 0, 41h for counter 1, and 42h for counter 2). 00 = Counter Latch Command 01 = Read/Write Least Significant Byte (LSB) 10 = Read/Write Most Significant Byte (MSB) 11 = Read/Write LSB then MSB	WO
3:1	Counter Mode Selection	These bits select one of six possible modes of operation for the selected counter. **000 = Mode 0Out signal on end of count (=0)** **001 = Mode 1Hardware retriggerable one-shot** **x10 = Mode 2Rate generator (divide by n counter)** **x11 = Mode 3Square wave output** **100 = Mode 4Software triggered strobe** **101 = Mode 5Hardware triggered strobe**	WO
0	Binary/BCD Countdown Select	0 = Binary countdown is used. The largest possible binary count is 2^{16} 1 = Binary coded decimal (BCD) count is used. The largest possible BCD count is 10^4	WO

8.3.1.1 RDBK_CMD—Read Back Command

Note: The Read Back Command is used to determine the count value, programmed mode, and current states of the OUT pin and Null count flag of the selected counter or counters. Status and/or count may be latched in any or all of the counters by selecting the counter during the register write. The count and status remain latched until read, and further latch commands are ignored until the count is read. Both count and status of the selected counters may be latched simultaneously by setting both bit 5 and bit 4 to zero. When both are latched, the first read operation from that counter returns the latched status. The next one or two reads, depending on whether the counter is programmed for one or two byte counts, returns the latched count. Subsequent reads return an unlatched count.

Table 233. RDBK_CMD—Read Back Command

Device: 31 ***Function:*** 0
Attribute: Read-Only
Size: 8-bit

Bits	Name	Description	Access
7:6	Read Back Command	Must be “11” to select the Read Back Command	
5	Latch Count of Selected Counters.	0 = Current count value of the selected counters will be latched 1 = Current count will not be latched	
4	Latch Status of Selected Counters.	0 = Status of the selected counters will be latched 1 = Status will not be latched	
3	Counter 2 Select.	1 = Counter 2 count and/or status will be latched	
2	Counter 1 Select.	1 = Counter 1 count and/or status will be latched	
1	Counter 0 Select.	1 = Counter 0 count and/or status will be latched.	
0	Reserved	Reserved. Must be 0.	

8.3.1.2 LTCH_CMD—Counter Latch Command

Note: The Counter Latch Command latches the current count value. This command is used to insure that the count read from the counter is accurate. The count value is then read from each counter's count register through the Counter Ports Access Ports Register (40h for counter 0, 41h for counter 1, and 42h for counter 2). The count must be read according to the programmed format, i.e., when the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other (read, write, or programming operations for other counters may be inserted between the reads). When a counter is latched once and then latched again before the count is read, the second Counter Latch Command is ignored.

Table 234. LTCH_CMD—Counter Latch Command

Device: 31		*Function:* 0 *Attribute:* Read-Only *Size:* 8-bit	
Bits	**Name**	**Description**	**Access**
7:6	Counter Selection	These bits select the counter for latching. When “11” is written, the write is interpreted as a read back command. 00 = Counter 0 01 = Counter 1 10 = Counter 2	
5:4	Counter Latch Command	00 = Selects the Counter Latch Command.	
3:0	Reserved	Reserved. Must be 0.	

8.3.2 SBYTE_FMT—Interval Timer Status Byte Format Register

Note: Each counter's status byte may be read following a Read Back Command. When latch status is chosen (bit 4=0, Read Back Command) as a read back option for a given counter, the next read from the counter's Counter Access Ports Register (40h for counter 0, 41h for counter 1, and 42h for counter 2) returns the status byte. The status byte returns the following:

Table 235. SBYTE_FMT—Interval Timer Status Byte Format Register

Device:	31	**Function:**	0
I/O Address:	Counter 0 = 40h, Counter 1 = 41h, Counter 2 = 42h	**Attribute:**	Read-Only
Default Value:	Bits[6:0] undefined, Bit 7=0	**Size:**	8-bit

Bits	Name	Description	Access
7	Counter OUT Pin State	0 = OUT pin of the counter is also a 0. 1 = OUT pin of the counter is also a 1.	RO
6	Count Register Status	This bit indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the counter mode, but until the count is loaded into the counting element (CE), the count value will be incorrect. 0 = Count has been transferred from CR to CE and is available for reading. 1 = Null Count. Count has not been transferred from CR to CE and is not yet available for reading.	RO
5:4	Read/Write Selection Status	These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 = Counter Latch Command 01 = Read/Write Least Significant Byte (LSB) 10 = Read/Write Most Significant Byte (MSB) 11 = Read/Write LSB then MSB	RO
3:1	Mode Selection Status	These bits return the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. 000 = Mode 0:Out signal on end of count (=0) 001 = Mode 1: Hardware retriggerable one-shot x10 = Mode 2: Rate generator (divide by n counter) x11 = Mode 3: Square wave output 100 = Mode 4: Software triggered strobe 101 = Mode 5: Hardware triggered strobe	RO
0	Countdown Type Status	This bit reflects the current countdown type. 0 = Binary countdown 1 = Binary Coded Decimal (BCD) countdown.	RO

8.3.3 Counter Access Ports Register

Table 236. Counter Access Ports Register

Device:	31	**Function:**	0
I/O Address:	Counter 0 - 40h, Counter 1 - 41h, Counter 2 - 42h	**Attribute:**	Read/Write
Default Value:	All bits undefined	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Counter Port	Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.	R/W

8.4 8259 Interrupt Controller (PIC) Registers

8.4.1 Interrupt Controller I/O MAP

The interrupt controller registers are located at 20h and 21h for the master controller (IRQ0 - 7), and at A0h and A1h for the slave controller (IRQ8 - 13). These registers have multiple functions, depending upon the data written to them. Listed in the table below are descriptions of the different register possibilities for each address.

Table 237. PIC Registers

Port	Aliases	Register Name/Function	Default Value	Type
20h	24h, 28h, 2Ch, 30h, 34h, 38h, 3Ch	Master PIC ICW1 Init. Cmd Word 1 Register	Undefined	WO
		Master PIC OCW2 Op Ctrl Word 2 Register	001XXXXXb	WO
		Master PIC OCW3 Op Ctrl Word 3 Register	X01XXX10b	R/W
21h	25h, 29h, 2Dh, 31h, 35h, 39h, 3Dh	Master PIC ICW2 Init. Cmd Word 2 Register	Undefined	WO
		Master PIC ICW3 Init. Cmd Word 3 Register	Undefined	WO
		Master PIC ICW4 Init. Cmd Word 4 Register	01h	WO
		Master PIC OCW1 Op Ctrl Word 1 Register	00h	R/W
A0h	A4h, A8h, ACh, B0h, B4h, B8h, BCh	Slave PIC ICW1 Init. Cmd Word 1 Register	Undefined	WO
		Slave PIC OCW2 Op Ctrl Word 2 Register	001XXXXXb	WO
		Slave PIC OCW3 Op Ctrl Word 3 Register	X01XXX10b	R/W

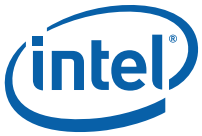


Table 237. PIC Registers

Port	Aliases	Register Name/Function	Default Value	Type
A1h	A5h, A9h, ADh, B1h, B5h, B9h, BDh	Slave PIC ICW2 Init. Cmd Word 2 Register	Undefined	WO
		Slave PIC ICW3 Init. Cmd Word 3 Register	Undefined	WO
		Slave PIC ICW4 Init. Cmd Word 4 Register	01h	WO
		Slave PIC OCW1 Op Ctrl Word 1 Register	00h	R/W
4D0h	-	Master PIC Edge/Level Triggered Register	00h	R/W
4D1h	-	Slave PIC Edge/Level Triggered Register	00h	R/W

8.4.2 ICW1—Initialization Command Word 1 Register

A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

1. The Interrupt Mask register is cleared.
2. IRQ7 input is assigned priority 7.
3. The slave mode address is set to 7.
4. Special mask mode is cleared and Status Read is set to IRR.

Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Table 238. ICW1—Initialization Command Word 1 Register

Device:	31	***Function:***	0
Offset:	Master Controller - 020h Slave Controller - 0A0h	***Attribute:***	Write-Only
Default Value:	All bits undefined	***Size:***	8-bit

Bits	Name	Description	Access
7:5	ICW/OCW select	These bits are MCS-85 specific, and not needed. 000 = Should be programmed to “000”	WO
4	ICW/OCW select	1 = This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.	WO
3	Edge/Level Bank Select (LTIM)	Disabled. Replaced by the edge/level triggered control registers (ELCR).	WO
2	ADI	0 = Ignored for the Intel® 6300ESB ICH. Should be programmed to 0.	WO
1	Single or Cascade (SNGL)	0 = Must be programmed to a 0 to indicate two controllers operating in cascade mode.	WO
0	ICW4 Write Required (IC4)	1 = This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.	WO

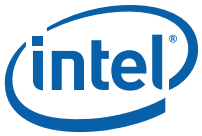
8.4.3 ICW2—Initialization Command Word 2 Register

Note: ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits[7:3] is used by the processor to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Table 239. ICW2—Initialization Command Word 2 Register

Device:	31	**Function:**	0
Offset:	Master Controller - 021h Slave Controller - 0A1h	**Attribute:**	Write-Only
Default Value:	All bits undefined	**Size:**	8-bit

Bits	Name	Description	Access
7:3	Interrupt Vector Base Address	Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.	WO
2:0	Interrupt Request Level	When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code: Code — Master Interrupt — Slave Interrupt 000 — IRQ0 — IRQ8 001 — IRQ1 — IRQ9 010 — IRQ2 — IRQ10 011 — IRQ3 — IRQ11 100 — IRQ4 — IRQ12 101 — IRQ5 — IRQ13 110 — IRQ6 — IRQ14 111 — IRQ7 — IRQ15	WO



8.4.4 ICW3—Master Controller Initialization Command Word 3 Register

Table 240. ICW3—Master Controller Initialization Command Word 3 Register

Device:	31	***Function:***	0
Offset:	21h	***Attribute:***	Write-Only
Default Value:	All bits undefined	***Size:***	8-bit

Bits	Name	Description	Access
7:3		0 = These bits must be programmed to zero.	
2	Cascaded Interrupt Controller IRQ Connection	This bit indicates that the slave controller is cascaded on IRQ2. When IRQ8#-IRQ15 is asserted, it goes through the slave controller's priority resolver. The slave controller's INTR output onto IRQ2. IRQ2 then goes through the master controller's priority solver. When it wins, the INTR signal is asserted to the processor and the returning interrupt acknowledge returns the interrupt vector for the slave controller. 1 = This bit must always be programmed to a 1.	WO
1:0		0 = These bits must be programmed to zero.	

8.4.5 ICW3—Slave Controller Initialization Command Word 3 Register

Table 241. ICW3—Slave Controller Initialization Command Word 3 Register

Device:	31	***Function:***	0
Offset:	A1h	***Attribute:***	Write-Only
Default Value:	All bits undefined	***Size:***	8-bit

Bits	Name	Description	Access
7:3		0 = These bits must be programmed to zero.	
2:0	Slave Identification Code	These bits are compared against the slave identification code broadcast by the master controller from the trailing edge of the first internal INTA# pulse to the trailing edge of the second internal INTA# pulse. These bits must be programmed to 02h to match the code broadcast by the master controller. When 02h is broadcast by the master controller during the INTA# sequence, the slave controller assumes responsibility for broadcasting the interrupt vector.	WO

8.4.6 ICW4—Initialization Command Word 4 Register

Table 242. ICW4—Initialization Command Word 4 Register

Device:	31	***Function:***	0
Offset:	Master Controller - 021h Slave Controller - 0A1h	***Attribute:***	Write-Only
Default Value:	All bits undefined	***Size:***	8-bit

Bits	Name	Description	Access
7:5		0 = These bits must be programmed to zero.	
4	Special Fully Nested Mode (SFNM)	0 = Should normally be disabled by writing a 0 to this bit. 1 = Special fully nested mode is programmed.	WO
3	Buffered Mode (BUF)	0 = Must be programmed to 0 for the Intel® 6300ESB ICH. This is non-buffered mode.	WO
2	Master/Slave in Buffered Mode	Not used. 0 = Should always be programmed to 0.	WO
1	Automatic End of Interrupt (AEOI)	0 = This bit should normally be programmed to 0. This is the normal end of interrupt. 1 = Automatic End of Interrupt (AEOI) mode is programmed.	WO
0	Microprocessor Mode	1 = Must be programmed to 1 to indicate that the controller is operating in an Intel® Architecture-based system. **NOTE:** Programming this bit to 0 will result in improper controller operation.	WO

8.4.7 OCW1—Operational Control Word 1 (Interrupt Mask) Register

Table 243. OCW1—Operational Control Word 1 (Interrupt Mask) Register

Device:	31	***Function:***	0
Offset:	Master Controller - 021h Slave Controller - 0A1h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Interrupt Request Mask	When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.	R/W

8.4.8 OCW2—Operational Control Word 2 Register

Note: Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Table 244. OCW2—Operational Control Word 2 Register

Device:	31	***Function:***	0
Offset:	Master Controller - 020h Slave Controller - 0A0h	***Attribute:***	Write-Only
Default Value:	Bit[4:0]=undefined, Bit[7:5]=001	***Size:***	8-bit

Bits	Name	Description	Access
7:5	Rotate and EOI Codes (R, SL, EOI)	These three bits control the Rotate and End of Interrupt modes and combinations of the two. 000 = Rotate in Auto EOI Mode (Clear) 001 = Non-specific EOI command 010 = No Operation 011 = Specific EOI Command 100 = Rotate in Auto EOI Mode (Set) 101 = Rotate on Non-Specific EOI Command 110 = *Set Priority Command 111 = *Rotate on Specific EOI Command *L0 - L2 Are Used	WO
4:3	OCW2 Select	When selecting OCW2, bits 4:3 = “00”	WO
2:0	Interrupt Level Select (L2, L1, L0)	L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined below, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function; programming L2, L1 and L0 to 0 is sufficient in this case. Bits Interrupt Level Bits Interrupt Level 000 IRQ0/8 100 IRQ4/12 001 IRQ1/9 101 IRQ5/13 010 IRQ2/10 110 IRQ6/14 011 IRQ3/11 111 IRQ7/15	WO

8.4.9 OCW3—Operational Control Word 3 Register

Table 245. OCW3—Operational Control Word 3 Register

Device:	31	**Function:**	0
Offset:	Master Controller - 020h Slave Controller - 0A0h	**Attribute:**	Write-Only
Default Value:	Bit[6,0]=0, Bit[7,4:2]=undefined, Bit[5,1]=1	**Size:**	8-bit

Bits	Name	Description	Access
7	Reserved	Reserved. Must be 0.	
6	Special Mask Mode (SMM)	1 = The Special Mask Mode may be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/disabling of the other channel's mask bits. Bit 5, the ESMM bit, must be set for this bit to have any meaning.	WO
5	Enable Special Mask Mode (ESMM)	0 = Disable. The SMM bit becomes a “don't care”. 1 = Enable the SMM bit to set or reset the Special Mask Mode.	WO
4:3	OCW3 Select	When selecting OCW3, bits 4:3 = ‘01’.	WO
2	Poll Mode Command	0 = Disable. Poll Command is not issued. 1 = Enable. The next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.	WO
1:0	Register Read Command	These bits provide control for reading the In-Service Register (ISR) and the Interrupt Request Register (IRR). When bit 1=0, bit 0 will not affect the register read selection. When bit 1=1, bit 0 selects the register status returned following an OCW3 read. When bit 0=0, the IRR will be read. When bit 0=1, the ISR will be read. Following ICW initialization, the default OCW3 port address read will be “read IRR”. To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register may be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. 00 = No Action 01 = No Action 10 = Read IRQ Register 11 = Read IS Register	WO

8.4.10 ELCR1—Master Controller Edge/Level Triggered Register

Note: In edge mode, (bit[x] = 0), the interrupt is recognized by a low to high transition. In level mode (bit[x] = 1), the interrupt is recognized by a high level. The cascade channel, IRQ2, the heart beat timer (IRQ0), and the keyboard controller (IRQ1), cannot be put into level mode.

Table 246. ELCR1—Master Controller Edge/Level Triggered Register

Device:	31	**Function:**	0
Offset:	4D0h	**Attribute:**	Read-Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7	IRQ7 ECL	0 = Edge. 1 = Level.	R/W
6	IRQ6 ECL	0 = Edge. 1 = Level.	R/W
5	IRQ5 ECL	0 = Edge. 1 = Level.	R/W
4	IRQ4 ECL	0 = Edge. 1 = Level.	R/W
3	IRQ3 ECL	0 = Edge. 1 = Level.	R/W
2:0	Reserved	Reserved. Must be 0.	

8.4.11 ELCR2—Slave Controller Edge/Level Triggered Register

Note: In edge mode, (bit[x] = 0), the interrupt is recognized by a low to high transition. In level mode (bit[x] = 1), the interrupt is recognized by a high level. The real time clock, IRQ8#, and the floating point error interrupt, IRQ13, cannot be programmed for level mode.

Table 247. ELCR2—Slave Controller Edge/Level Triggered Register (Sheet 1 of 2)

Device:	31	**Function:**	0
Offset:	4D1h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7	IRQ15 ECL	0 = Edge. 1 = Level.	R/W
6	IRQ14 ECL	0 = Edge. 1 = Level.	R/W
5	Reserved	Reserved. Must be 0.	
4	IRQ12 ECL	0 = Edge. 1 = Level.	R/W

Table 247. ELCR2—Slave Controller Edge/Level Triggered Register (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	4D1h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
3	IRQ11 ECL	0 = Edge. 1 = Level.	R/W
2	IRQ10 ECL	0 = Edge. 1 = Level.	R/W
1	IRQ9 ECL	0 = Edge. 1 = Level.	R/W
0	Reserved	Reserved. Must be 0.	

8.5 Advanced Interrupt Controller (APIC0)

There are two APICs in the Intel® 6300ESB ICH: APIC0 and APIC1 (device29, function 5). APIC0's direct registers are assigned with base address FEC0xxxxH; however, only primary (legacy) PCI device may write to these registers. APIC1's direct registers are assigned with base address FEC1xxxxH. To support legacy device/drivers on external PCI bus used with the Intel® ICHx, APIC1 has an alternate base address, FEC0xxxxH. This means external PCI devices may write to the IRQ pin assertion register (either FEC0_0020H or FEC1_0020H) to generate interrupts from APIC1.

Since the Intel® 6300ESB ICH does not implement Hub Interface EOI special cycle, the MCH will translate EOI special cycle to a memory write cycle to EOI register at address FEC0_0040H and pass it to the Intel® 6300ESB ICH. This memory write cycle will be passed to both APIC0 and APIC1 internally.

From the CPU/MCH point of view, it should always use address FEC0xxxxH to access APIC0 registers and address FEC1xxxxH to access APIC1 registers. APIC1 will not respond to the CPU/MCU's access to address FEC0xxxxH, other than the EOI cycle stated above.

APIC0 also includes an XAPIC_EN config bit. This bit must be set to enable the I/O (x) APIC extension to the I/O APIC. This allows the extended feature to be disabled if a problem is found. For APIC1, this extension is always enabled.

8.5.1 APIC Register Map

The APIC is accessed through an indirect addressing scheme. Two registers are visible by software for manipulation of most of the APIC registers. These registers are mapped into memory space and are shown in Table 248.

Table 248. APIC Direct Registers

Address	Register	Size	Type
FEC0_0000h	Index Register	8 bits	R/W

Table 248. APIC Direct Registers

FEC0_0010h	Data Register	32 bits	R/W
FECO_0020h	IRQ Pin Assertion Register	32 bits	WO
FECO_0040h	EOI Register	32 bits	WO

Table 249 lists the registers which may be accessed within the APIC through the Index Register. When accessing these registers, accesses must be done a DWORD at a time. For example, software should never access byte 2 from the Data register before accessing bytes 0 and 1. The hardware will not attempt to recover from a bad programming model in this case.

Table 249. APIC Indirect Registers

Index	Register	Size	Type
00	ID	32 bits	R/W
01	Version	32 bits	RO
02	Arbitration ID	32 bits	RO
03	Boot Configuration	32 bits	R/W
03-0F	Reserved		RO
10 -11	Redirection Table 0	64 bits	R/W
12 - 13	Redirection Table 1	64 bits	R/W
…	…	…	…
3E-3F	Redirection Table 23	64 bits	R/W
40-FF	Reserved		RO

8.5.2 IND—Index Register

Note: The Index Register will select which APIC indirect register to be manipulated by software. The selector values for the indirect registers are listed in Table 249. Software will program this register to select the desired APIC internal register.

Table 250. IND—Index Register

Device:	31	***Function:***	0
Offset:	FEC0_0000h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	APIC Index	This is an 8-bit pointer into the I/O APIC register table.	R/W

8.5.3 DAT—Data Register

This is a 32-bit register specifying the data to be read or written to the register pointed to by the Index register. This register may only be accessed in DWORD quantities.

Table 251. DAT—Data Register

Device:	31	**Function:**	0
Offset:	FEC0_0010h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
7:0	APIC Data	This is a 32-bit register for the data to be read or written to the APIC indirect register pointed to by the Index register.	R/W

8.5.4 Offset FEC0_0020h: IRQPA—IRQ Pin Assertion Register

Note: The IRQ Pin Assertion Register is present to provide a mechanism to scale the number of interrupt inputs into the I/O APIC without increasing the number of dedicated input pins. When a device that supports this interrupt assertion protocol requires interrupt service, that device will issue a write to this register. Bits 4:0 written to this register contain the IRQ number for this interrupt. The only valid values are 0-23. Bits 31:5 are ignored. To provide for future expansion, peripherals should always write a value of 0 for Bits 31:5.

See Section 5.7.4, "Interrupt Mapping" for more details on how PCI devices will use this field.

Note: Writes to this register are only allowed by the processor and by masters on the Intel® 6300ESB ICH's PCI bus. Writes by devices on PCI buses above the Intel® 6300ESB ICH are not supported.

Table 252. Offset FEC0_0020h: IRQPA—IRQ Pin Assertion Register

Device:	31	**Function:**	0
Offset:	FEC0_0020h	**Attribute:**	Write-Only
Default Value:	N/A	**Size:**	32-bit

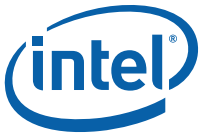
Bits	Name	Description	Access
31:5	Reserved	Reserved. To provide for future expansion, the processor should always write a value of 0 to Bits 31:5.	
4:0	IRQ Number	Bits 4:0 written to this register contain the IRQ number for this interrupt. The only valid values are 0-23.	WO

8.5.5 Offset FEC0 - EOIR: EOI Register

The EOI register is present to provide a mechanism to maintain the level triggered semantics for level-triggered interrupts issued on the parallel bus.

When a write is issued to this register, the I/O APIC will check the lower eight bits written to this register, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared.

Note: This is similar to what already occurs when the APIC sees the EIO message on the serial bus. Note that if multiple I/O Redirection entries, for any reason, assign the same



vector for more than one interrupt input, each of those entries will have the Remote_IRR bit reset to zero. The interrupt which was prematurely reset will not be lost because if its input remained active when the Remote_IRR bit is cleared, the interrupt will be reissued and serviced at a later time.

Table 253. Offset FEC0 - EOIR: EOI Register

Device:	31	***Function:***	0
Offset:	FEC0_0040h	***Attribute:***	Write-Only
Default Value:	N/A	***Size:***	32-bit

Bits	Name	Description	Access
31:8	Reserved	Reserved. To provide for future expansion, the processor should always write a value of zero to Bits 31:8.	
7:0	Redirection Entry Clear	When a write is issued to this register, the I/O APIC will check this field, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared.	WO

8.5.6 Offset 00h: ID—Identification Register

Note: The APIC ID serves as a physical name of the APIC. The APIC bus arbitration ID for the APIC is derived from its I/O APIC ID. This register is reset to zero on power up reset.

Table 254. Offset 00h: ID—Identification Register

Device:	31	***Function:***	0
Offset:	00h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	16-bit

Bits	Name	Description	Access
31:28	Reserved	Reserved.	
27:24	APIC ID	Software must program this value before using the APIC.	R/W
23:16	Reserved	Reserved.	
15	Scratchpad	Scratchpad bit.	
14:0	Reserved	Reserved.	

8.5.7 Offset 01h: VER—Version Register

Note: Each I/O APIC contains a hardwired Version Register that identifies different implementation of APIC and their versions. The maximum redirection entry information also is in this register, to let software know how many interrupt are supported by this APIC.

Table 255. Offset 01h: VER—Version Register

Device:	31	***Function:***	0
Offset:	01h	***Attribute:***	Read-Only
Default Value:	00170020h	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23:16	Maximum Redirection Entries	This is the entry number (0 being the lowest entry) of the highest entry in the redirection table. It is equal to the number of interrupt input pins minus one and is in the range 0 through 239. In the Intel® 6300ESB ICH this field is hardwired to 17h to indicate 24 interrupts.	RO
15	PRQ	This bit is set to 1 to indicate that this version of the I/O APIC implements the IRQ Assertion register and allows PCI devices to write to it to cause interrupts.	RO
14:8	Reserved	Reserved.	
7:0	Version	This is a version number that identifies the implementation version. The version number assigned to the Intel® 6300ESB ICH for the I/O (x) APIC is 20h.	RO

8.5.8 Offset 02h: ARBID—Arbitration ID Register

Note: This register contains the bus arbitration priority for the APIC. When the APIC Clock is running, this register is loaded whenever the APIC ID register is loaded. A rotating priority scheme is used for APIC bus arbitration. The winner of the arbitration becomes the lowest priority agent and assumes an arbitration ID of zero.

Table 256. Offset 02h: ARBID—Arbitration ID Register

Device:	31	***Function:***	0
Offset:	02h	***Attribute:***	Read-Only
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:28	Reserved	Reserved.	
27:24	I/O APIC Identification	This 4-bit field contains the I/O APIC Arbitration ID.	RO
23:0	Reserved	Reserved.	

8.5.9 Offset 03h: BOOT_CONFIG—Boot Configuration Register

Table 257. Offset 02h: ARBID—Arbitration ID Register

Device:	31	**Function:**	0
Offset:	02h	**Attribute:**	Read-Only
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:1	Reserved	Reserved	
0	DT: Delivery Type	Hardwire to 1. Interrupt delivery mechanism is always a Processor System Bus message.	RO

8.5.10 Offset 10h - 11h (Vector 0) through 3E - 3Fh (Vector 23): Redirection Table

The Redirection Table has a dedicated entry for each interrupt input pin. The information in the Redirection Table is used to translate the interrupt manifestation on the corresponding interrupt pin into an APIC message.

The APIC will respond to an edge triggered interrupt as long as the interrupt is held until after the acknowledge cycle has begun. Once the interrupt is detected, a delivery status bit internally to the I/O APIC is set. The state machine will step ahead and wait for an acknowledgment from the APIC bus unit that the interrupt message was sent over the APIC bus. Only then will the I/O APIC be able to recognize a new edge on that interrupt pin. That new edge will only result in a new invocation of the handler if its acceptance by the destination APIC causes the Interrupt Request Register bit to go from 0 to 1. (i.e., when the interrupt was not already pending at the destination).

See Table 259 for Delivery Mode Encoding information.

Table 258. Offset 10h - 11h (Vector 0) through 3E - 3Fh (Vector 23): Redirection Table (Sheet 1 of 2)

Device:	31	**Function:**	0
Offset:	10h-11h (vector 0) through 3E-3Fh (vector 23)	**Attribute:**	Read/Write
Default Value:	Bit 16-1, Bits[15:12]=0. All other bits undefined	**Size:**	64-bit (accessed as two 32 bit quantities)

Bits	Name	Description	Access
63:56	Destination	When bit 11 of this entry is 0 [Physical], then bits [59:56] specify an APIC ID. In this case, bits 63:59 should be programmed by software to 0. When bit 11 of this entry is 1 [Logical], bits [63:56] specify the logical destination address of a set of processors.	R/W
55:48	Extended Destination ID (EDID)	These bits are only sent to a local APIC when in Processor System Bus mode. They become bits [11:4] of the address.	
47:17	Reserved	Reserved. Software should program these bits to 0	

Table 258. Offset 10h - 11h (Vector 0) through 3E - 3Fh (Vector 23): Redirection Table (Sheet 2 of 2)

Device:	31	**Function:**	0
Offset:	10h-11h (vector 0) through 3E-3Fh (vector 23)	**Attribute:**	Read/Write
Default Value:	Bit 16-1, Bits[15:12]=0. All other bits undefined	**Size:**	64-bit (accessed as two 32 bit quantities)

Bits	Name	Description	Access
16	Mask	0 = Not masked: An edge or level on this interrupt pin results in the delivery of the interrupt to the destination. 1 = Masked: Interrupts are not delivered nor held pending. Setting this bit after the interrupt is accepted by a local APIC has no effect on that interrupt. This behavior is identical to the device withdrawing the interrupt before it is posted to the processor. It is software's responsibility to deal with the case where the mask bit is set after the interrupt message has been accepted by a local APIC unit but before the interrupt is dispensed to the processor.	R/W
15	Trigger Mode	This field indicates the type of signal on the interrupt pin that triggers an interrupt. 0 = Edge triggered. 1 = Level triggered.	R/W
14	Remote IRR	This bit is used for level triggered interrupts (in Fixed or Lowest priority Delivery Modes only); its meaning is undefined for edge triggered interrupts. For level triggered interrupts, this bit is set if the I/O APIC successfully sends the level interrupt vector in this entry. This bit is never set for SMI, NMI, INT or ExtINT delivery modes.	R/W
13	Interrupt Input Pin Polarity	This bit specifies the polarity of each interrupt signal connected to the interrupt pins. 0 = Active high. 1 = Active low.	R/W
12	Delivery Status	This field contains the current status of the delivery of this interrupt. Writes to this bit have no effect. 0 = Idle. No activity for this interrupt. 1 = Pending. Interrupt has been injected, but delivery is held up due to the APIC bus being busy or the inability of the receiving APIC unit to accept the interrupt at this time.	RO
11	Destination Mode	This field determines the interpretation of the Destination field. 0 = Physical. Destination APIC ID is identified by bits [59:56]. 1 = Logical. Destinations are identified by matching bit [63:56] with the Logical Destination in the Destination Format Register and Logical Destination Register in each Local APIC.	R/W
10:8	Delivery Mode	This field specifies how the APICs listed in the destination field should act upon reception of this signal. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. These encodings are listed in the note below:	R/W
7:0	Vector	This field contains the interrupt vector for this interrupt. Values range between 10h and FEh.	R/W

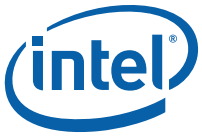


Table 259. Delivery Mode Encoding

Bits	Description
000	**Fixed:** Deliver the signal on the INTR signal of all processor cores listed in the destination. Trigger Mode may be edge or level.
001	**Lowest Priority:** Deliver the signal on the INTR signal of the processor core that is executing at the lowest priority among all the processors listed in the specified destination. Trigger Mode may be edge or level.
010	**SMI (System Management Interrupt):** Requires the interrupt to be programmed as edge triggered. The vector information is ignored but must be programmed to all zeroes for future compatibility. -- not supported.
011	Reserved
100	**NMI:** Deliver the signal on the NMI signal of all processor cores listed in the destination. Vector information is ignored. NMI is treated as an edge triggered interrupt even when it is programmed as level triggered. For proper operation this redirection table entry must be programmed to edge triggered. The NMI delivery mode does not set the RIRR bit. Once the interrupt is detected, it will be sent over the APIC bus. When the redirection table is incorrectly set to level, the loop count will continue counting through the redirection table addresses. Once the count for the NMI pin is reached again, the interrupt will be sent over the APIC bus again. -- not supported.
101	**INIT:** Deliver the signal to all processor cores listed in the destination by asserting the INIT signal. All addressed local APICs will assume their INIT state. INIT is always treated as an edge triggered interrupt even when programmed as level triggered. For proper operation this redirection table entry must be programmed to edge triggered. The INIT delivery mode does not set the RIRR bit. Once the interrupt is detected, it will be sent over the APIC bus. When the redirection table is incorrectly set to level, the loop count will continue counting through the redirection table addresses. Once the count for the INIT pin is reached again, the interrupt will be sent over the APIC bus again. -- not supported.
110	Reserved.
111	**ExtINT:** Deliver the signal to the INTR signal of all processor cores listed in the destination as an interrupt that originated in an externally connected 8259A compatible interrupt controller. The INTA cycle that corresponds to this ExtINT delivery will be routed to the external controller that is expected to supply the vector. Requires the interrupt to be programmed as edge triggered.

8.6 Real Time Clock Registers

8.6.1 I/O Register Address Map

The RTC internal registers and RAM are organized as two banks of 128 bytes each, called the standard and extended banks. The first 14 bytes of the standard bank contain the RTC time and date information along with four registers, A - D, that are used for configuration of the RTC. The extended bank contains a full 128 bytes of battery backed SRAM, and will be accessible even when the RTC module is disabled (through the RTC configuration register). Registers A-D do not physically exist in the RAM.

All data movement between the host processor and the real-time clock is done through registers mapped to the standard I/O space. The register map appears in Table 260.

Table 260. RTC I/O Registers

I/O Locations	When U128E bit = 0	Function
70h and 74h	Also alias to 72h and 76h	Real-Time Clock (Standard RAM) Index Register **NOTE:** Writes to 72h, 74h and 76h do not affect the NMI enable (bit 7 of 70h)
71h and 75h	Also alias to 73h and 77h	Real-Time Clock (Standard RAM) Target Register
72h and 76h		Extended RAM Index Register (when enabled)
73h and 77h		Extended RAM Target Register (when enabled)

NOTES:
1. I/O locations 70h and 71h are the standard ISA location for the real-time clock. The map for this bank is shown in Table 261. Locations 72h and 73h are for accessing the extended RAM. The extended RAM bank is also accessed using an indexed scheme. I/O address 72h is used as the address pointer and I/O address 73h is used as the data register. Index addresses above 127h are not valid. When the extended RAM is not needed, it may be disabled.
2. Software must preserve the value of bit 7 at I/O addresses 70h and 74h. When writing to these addresses, software must first read the value, and then write the same value for bit 7 during the sequential address write.

8.6.2 Indexed Registers

The RTC contains two sets of indexed registers that are accessed using the two separate Index and Target registers (70/71h or 72/73h), as shown in Table 261.

Table 261. RTC (Standard) RAM Bank

Index	Name
00h	Seconds
01h	Seconds Alarm
02h	Minutes
03h	Minutes Alarm
04h	Hours
05h	Hours Alarm
06h	Day of Week
07h	Day of Month
08h	Month
09h	Year
0Ah	Register A
0Bh	Register B
0Ch	Register C
0Dh	Register D
0Eh - 7Fh	114 Bytes of User RAM

8.6.2.1 RTC_REGA—Register A

Note: This register is used for general configuration of the RTC functions. None of the bits are affected by RSMRST# or any other Intel® 6300ESB ICH reset signal.

Table 262. RTC_REGD—Register D (Flag Register)

Device:	31	***Function:***	0
RTC Index:	0A	***Attribute:***	Read/Write
Default Value:	Undefined	***Size:***	8-bit
Lockable:	No	***Power Well:***	RTC

Bits	Name	Description	Access
7	UIP: Update In Progress	This bit may be monitored as a status flag. 0 = The update cycle will not start for at least 492 µs. The time, calendar, and alarm information in RAM is always available when the UIP bit is 0. 1 = The update is soon to occur or is in progress.	R/W
6:4	DV[2:0]: Division Chain Select	These three bits control the divider chain for the oscillator, and are not affected by RSMRST# or any other reset signal. DV[2] corresponds to bit 6. 010 = Normal Operation 11X = Divider Reset 101 = Bypass 15 stages (test mode only) 100 = Bypass 10 stages (test mode only) 011 = Bypass 5 stages (test mode only) 001 = Invalid 000 = Invalid	R/W
3:0	RS[3:0] Rate Select	Selects one of 13 taps of the 15 stage divider chain. The selected tap may generate a periodic interrupt when the PIE bit is set in Register B. Otherwise this tap will set the PF flag of Register C. When the periodic interrupt is not to be used, these bits should all be set to zero. RS3 corresponds to bit 3. 0000 = Interrupt never toggles 0001 = 3.90625 ms 0010 = 7.8125 ms 0011 = 122.070 µs 0100 = 244.141 µs 0101 = 488.281 µs 0110 = 976.5625 µs 0111 = 1.953125 ms 1000 = 3.90625 ms 1001 = 7.8125 ms 1010 = 15.625 ms 1011 = 31.25 ms 1100 = 62.5 ms 1101 = 125 ms 1110 = 250 ms 1111= 500 ms	R/W

8.6.2.2 RTC_REGB—Register B (General Configuration)

Table 263. RTC_REGB—Register B (General Configuration)

Device:	31	***Function:***	0
RTC Index:	0Bh	***Attribute:***	Read-Write
Default Value:	U0U00UUU (U: Undefined)	***Size:***	8-bit
Lockable:	No	***Power Well:***	RTC

Bits	Name	Description	Access
7	SET: Update Cycle Inhibit	Enables/Inhibits the update cycles. This bit is not affected by RSMRST# nor any other reset signal. 0 = Update cycle occurs normally once each second. 1 = A current update cycle will abort and subsequent update cycles will not occur until SET is returned to zero. When set is one, the BIOS may initialize time and calendar bytes safely.	R/W
6	PIE: Periodic Interrupt Enable	This bit is cleared by RSMRST#, but not on any other reset. 0 = Disable. 1 = Allows an interrupt to occur with a time base set with the RS bits of register A.	R/W
5	AIE: Alarm Interrupt Enable	This bit is cleared by RSMRST#, but not on any other reset. 0 = Disable. 1 = Allows an interrupt to occur when the AF is set by an alarm match from the update cycle. An alarm may occur once a second, one an hour, once a day, or one a month.	R/W
4	UIE: Update-Ended Interrupt Enable	This bit is cleared by RSMRST#, but not on any other reset. 0 = Disable. 1 = Allows an interrupt to occur when the update cycle ends.	R/W
3	SQWE: Square Wave Enable	This bit serves no function in the Intel® 6300ESB ICH. It is left in this register bank to provide compatibility with the Motorola* 146818B. The Intel® 6300ESB ICH has no SQW pin. This bit is cleared by RSMRST#, but not on any other reset.	R/W

Table 263. RTC_REGB—Register B (General Configuration)

Device:	31	**Function:**	0
RTC Index:	0Bh	**Attribute:**	Read-Write
Default Value:	U0U00UUU (U: Undefined)	**Size:**	8-bit
Lockable:	No	**Power Well:**	RTC

Bits	Name	Description	Access
2	DM: Data Mode	Specifies either binary or BCD data representation. This bit is not affected by RSMRST# nor any other reset signal. 0 = BCD 1 = Binary	R/W
1	HOURFORM: Hour Format	Indicates the hour byte format. This bit is not affected by RSMRST# nor any other reset signal. 0 = Twelve-hour mode. In twelve-hour mode, the seventh bit represents AM as zero and PM as one. 1 = Twenty-four hour mode.	R/W
0	DSE: Daylight Savings Enable	Triggers two special hour updates per year. The days for the hour adjustment are those specified in United States federal law as of 1987, which is different than previous years. This bit is not affected by RSMRST# nor any other reset signal. 0 = Daylight Savings Time updates do not occur. 1 = a) Update on the first Sunday in April, where time increments from 1:59:59 AM to 3:00:00 AM. b) Update on the last Sunday in October when the time first reaches 1:59:59 AM, it is changed to 1:00:00 AM. The time must increment normally for at least two update cycles (seconds) previous to these conditions for the time change to occur properly.	R/W

8.6.2.3 RTC_REGC—Register C (Flag Register)

Note: Writes to Register C have no effect.

Table 264. RTC_REGC—Register C (Flag Register)

Device:	31	***Function:***	0
RTC Index:	0Ch	***Attribute:***	Read-Only
Default Value:	00U00000 (U: Undefined)	***Size:***	8-bit
Lockable:	No	***Power Well:***	RTC

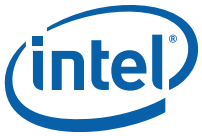
Bits	Name	Description	Access
7	IRQF: Interrupt Request Flag	IRQF = (PF * PIE) + (AF * AIE) + (UF *UFE). This also causes the CH_IRQ_B signal to be asserted. This bit is cleared upon RSMRST# or a read of Register C.	RO
6	PF: Periodic Interrupt Flag	This bit is cleared upon RSMRST# or a read of Register C. 0 = When no taps are specified through the RS bits in Register A, this flag will not be set. 1 = Periodic interrupt Flag will be 1 whenever the tap specified by the RS bits of register A is 1.	RO
5	AF: Alarm Flag	0 = This bit is cleared upon RTCRST# or a read of Register C. 1 = Alarm Flag will be set after all Alarm values match the current time.	RO
4	UF: Update-ended Flag	0 = The bit is cleared upon RSMRST# or a read of Register C. 1 = Set immediately following an update cycle for each second.	RO
3:0	Reserved	Reserved. Will always report 0.	

8.6.3 RTC_REGD—Register D (Flag Register)

Table 265. RTC_REGD—Register D (Flag Register)

Device:	31	***Function:***	0
Offset:	0Dh	***Attribute:***	Read/Write
Default Value:	10UUUUUU (U: Undefined)	***Size:***	8-bit
Lockable:	No	***Power Well:***	RTC

Bits	Name	Description	Access
7	VRT: Valid RAM and Time Bit	0 = This bit should always be written as a 0 for write cycle, however it will return a 1 for read cycles. 1 = This bit is hard-wired to 1 in the RTC power well.	R/W
6	Reserved	Reserved. This bit always returns a 0 and should be set to 0 for write cycles.	
5:0	Date Alarm	These bits store the date of month alarm value. When set to 000000b, then a “does not care” state is assumed. The host must configure the date alarm for these bits to do anything, yet they may be written at any time. When the date alarm is not enabled, these bits will return zeros to mimic the functionality of the Motorola 146818B. These bits are not affected by RESET.	R/W



8.7 CPU Interface Registers

8.7.1 NMI_SC—NMI Status and Control Register

Table 266. NMI_SC—NMI Status and Control Register

Device:	31	***Function:***	0
Offset:	61h	***Attribute:***	Read/Write, Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7	SERR#_NMI_STS: SERR# NMI Source Status	1 = PCI agent detected a system error and pulses the PCI SERR# line. This interrupt source is enabled by setting bit 2 to 0. To reset the interrupt, set bit 2 to 1 and then set it to 0. When writing to port 61h, this bit must be 0.	RO
6	IOCHK_NMI_STS: IOCHK# NMI Source Status	1 = An ISA agent (through SERIRQ) asserted IOCHK# on the ISA bus. This interrupt source is enabled by setting bit 3 to 0. To reset the interrupt, set bit 3 to 0 and then set it to 1. When writing to port 61h, this bit must be a 0.	RO
5	TMR2_OUT_STS: Timer Counter 2 OUT Status	This bit reflects the current state of the 8254 counter 2 output. Counter 2 must be programmed following any PCI reset for this bit to have a determinate value. When writing to port 61h, this bit must be a 0.	RO
4	REF_TOGGLE: Refresh Cycle Toggle	This signal toggles from either 0 to 1 or 1 to 0 at a rate that is equivalent to when refresh cycles would occur. When writing to port 61h, this bit must be a 0.	RO
3	IOCHK_NMI_EN: IOCHK# NMI Enable	0 = Enabled. 1 = Disabled and cleared.	R/W
2	PCI_SERR_EN: PCI SERR# Enable	0 = SERR# NMIs are enabled. 1 = SERR# NMIs are disabled and cleared.	R/W
1	SPKR_DAT_EN: Speaker Data Enable	0 = SPKR output is a 0. 1 = SPKR output is equivalent to the Counter 2 OUT signal value.	R/W
0	TIM_CNT2_EN: Timer Counter 2 Enable	0 = Disable 1 = Enable	R/W

8.7.2 NMI_EN—NMI Enable (and Real Time Clock Index)

Note: The RTC Index field is write-only for normal operation. This field may only be read in Alt-Access Mode. This register is aliased to Port 74h, and all bits are readable at that address.

Table 267. NMI_EN—NMI Enable (and Real Time Clock Index)

Device:	31	***Function:***	0
I/O Address:	70h	***Attribute:***	Read/Write (Special)
Default Value:	80h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7	NMI_EN: NMI Enable	0 = Enable NMI sources. 1 = Disable All NMI sources.	R/W
6:0	RTC_INDX: Real Time Clock Index Address	This data goes to the RTC to select which register or CMOS RAM address is being accessed.	R/W

8.7.3 PORT92—Fast A20 and Init Register

Table 268. PORT92—Fast A20 and Init Register

Device:	31	***Function:***	0
I/O Address:	92h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1	ALT_A20_GATE: Alternate A20 Gate	This bit is ORed with the A20GATE input signal to generate A20M# to the processor. 0 = A20M# signal may potentially go active. 1 = This bit is set when INIT# goes active.	R/W
0	INIT_NOW	When this bit transitions from a 0 to a 1, the Intel® 6300ESB ICH will force INIT# active for 16 PCI clocks.	R/W

8.7.4 COPROC_ERR—Coprocessor Error Register

Table 269. COPROC_ERR—Coprocessor Error Register

Device:	31	**Function:**	0
I/O Address:	F0h	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	COPROC_ERR	Any value written to this register will cause IGNNE# to go active, if FERR# had generated an internal IRQ13. For FERR# to generate an internal IRQ13, the COPROC_ERR_EN bit (Device 31:Function 0, Offset D0, Bit 13) must be 1.	WO

8.7.5 RST_CNT—Reset Control Register

Table 270. RST_CNT—Reset Control Register

Device:	31	**Function:**	0
Offset:	CF9h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3	FULL_RST: Full Reset	This bit is used to determine the states of SLP_S3#, SLP_S4# and SLP_S5# after a CF9 hard reset (SYS_RST =1 and RST_CPU is set to 1), after PWROK going low (with RSMRST# high), or after two TCO timeouts. 0 = The Intel® 6300ESB ICH will keep SLP_S3#, SLP_S4# and SLP_S5# high. 1 = The Intel® 6300ESB ICH will drive SLP_S3#, SLP_S4# and SLP_S5# low for 3 - 5 seconds.	R/W
2	RST_CPU: Reset CPU	When this bit transitions from a 0 to a 1, it initiates a hard or soft reset, as determined by the SYS_RST bit (bit 1 of this register).	R/W
1	SYS_RST: System Reset	This bit is used to determine a hard or soft reset to the processor. 0 = When RST_CPU bit goes from 0 to 1, the Intel® 6300ESB ICH performs a soft reset by activating INIT# for 16 PCI clocks. 1 = When RST_CPU bit goes from 0 to 1, the Intel® 6300ESB ICH performs a hard reset by activating PXPCIRST# for 1 millisecond. It also resets the resume well bits (except for those noted throughout the datasheet). The SLP_S3#, SLP_S4#, and SLP_S5# signals will not go active.	R/W
0	Reserved	Reserved.	

8.8 Power Management Registers (D31:F0)

The power management registers are distributed within the PCI Device 31: Function 0 space, as well as a separate I/O range. Each register is described below. Unless otherwise indicate, bits are in the main (core) power well.

Bits not explicitly defined in each register are assumed to be reserved. When writing to a reserved bit, the value should always be 0. Software should not attempt to use the value read from a reserved bit, as it may not be consistently 1 or 0.

8.8.1 Power Management PCI Configuration Registers (D31:F0)

Table 271 shows a small part of the configuration space for PCI Device 31: Function 0. It includes only those registers dedicated for power management. Some of the registers are only used for Legacy Power management schemes.

Table 271. Power Management PCI Configuration Registers (D31:F0)

Offset	Mnemonic	Register Name/Function	Default	Type
40h-43h	ACPI_BASE	ACPI Base Address	00000001 h	R/W
44h	ACPI_CNTL	ACPI Control	00h	R/W
A0h	GEN_PMCON_1	General Power Management Configuration 1	0000h	R/W
A2h	GEN_PMCON_2	General Power Management Configuration 2	00h	R/W
A4h	GEN_PMCON_3	General Power Management Configuration 3	00h	R/W
A8h	Reserved		0Dh	R/W
B8 - BBh	GPI_ROUT	GPI Route Control	00000000 h	R/W
C0h	MON_FWD_EN	I/O Monitor Forward Enable	00h	RW
C4 - CAhh	MON[*n*]_TRP_RNG	I/O Monitor[4:7] Trap Range	0000h	R/W
CCh	MON_TRP_MSK	I/O Monitor Trap Range Mask	0000h	R/W

8.8.1.1 Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—D31:F0)

Note: Usage: ACPI or Legacy.

Table 272. Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	A0h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10	Reserved	Reserved.	
9	PWRBTN_LVL	This read-only bit indicates the current state of the PWRBTN# signal. 0 = Low. 1 = High.	RO
8:6	Reserved	Reserved.	
5	CPUSLP_EN: CPU SLP# Enable	0 = Disable. 1 = Enables the CPUSLP# signal to go active in the S1-D states. This reduces the processor power. Note that CPUSLP# will go active on entry to S3, S4 and S5 even when this bit is not set.	R/W
4	SMI_LOCK	When this bit is set, writes to the GLB_SMI_EN bit will have no effect Once the SMI_LOCK bit is set, writes of 0 to SMI_LOCK bit will have no effect (i.e. once set, this bit may only be cleared by PXPCIRST#).	R/WO
3:2	Reserved	Reserved.	
1:0	PER_SMI_SEL: Periodic SMI# rate Select	Set by software to control the rate at which periodic SMI# is generated. 00 = 64 seconds 01 = 32 seconds 10 = 16 seconds 11 = 8 seconds	R/W

8.8.1.2 Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—D31:F0)

Note: Usage: ACPI or Legacy.

Table 273. Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	A2h	***Attribute:***	Read-Only Clear
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	System Reset Status (SRS)	The Intel® 6300ESB ICH sets this bit when the SYS_RESET# button is pressed. BIOS is expected to read this bit and clear it when it is set. This bit is also reset by RSMRST# and CF9h resets. SRS bit is set only when the system is in S0 or S1 state.	R/WC
3	CPU Thermal Trip Status (CTS)	This bit is set when PXPCIRST# is inactive and CPUTHRMTRIP# goes active while the system is in an S0 or S1 state. This bit is also reset by RSMRST# and CF9h resets. It is not reset by the shutdown and reboot associated with the CPUTHRMTRIP# event.	R/WC
2	Reserved	Reserved.	
1	CPUPWR_FLR: CPU Power Failure	0 = Software clears this bit by writing a 0 to the bit position. 1 = Indicates that the PWRGD signal from the CPU's VRM went low. Software clears this bit by writing a 0 to this bit position.	R/WC
0	PWROK_FLR: PWROK Failure	0 = Software clears this bit by writing a 1 to the bit position, or when the system goes into a G3 state. 1 = This bit will be set any time PWROK goes low, when the system was in S0, or S1 state. The bit will be cleared only by software by writing a 1 to this bit or when the system goes to a G3 state. **NOTE:** Traditional designs have a reset button logically OR'd with the PWROK signal from the power supply and the CPU's voltage regulator module. When this is done with the Intel® 6300ESB ICH, the PWROK_FLR bit will be set. The Intel® 6300ESB ICH treats this internally as though the RSMRST# signal had gone active. However, it is not treated as a full power failure. When PWROK goes inactive and then active (but RSMRST# stays high), the Intel® 6300ESB ICH will reboot (regardless of the state of the AFTERG3 bit). When the RSMRST# signal also goes low before PWROK goes high, this is a full power failure and the reboot policy is controlled by the AFTERG3 bit. **NOTE:** In the case of true PWROK failure, PWROK will go low first before PWRGD.	R/WC

8.8.1.3 Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—D31:F0)

Note: Usage: ACPI or Legacy.

Table 274. Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	A4h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	RTC

Bits	Name	Description	Access
7:6	SWSMI_RATE_SEL	This 2-bit value indicates when the SWSMI timer will time out. Valid values are: 00 1.5 ms ± 0.5 ms 01 16 ms ± 4 ms 10 32 ms ± 4 ms 11 64 ms ± 4 ms	R/W
5:3	Reserved	Reserved.	
2	RTC_PWR_STS: RTC Power Status	This bit is set when RTCRST# is low. The bit is not cleared by any type of reset.	R/W
1	PWR_FLR: Power Failure	This bit is in the RTC well, and is not cleared by any type of reset except RTCRST#. 0 = Indicates that the trickle current has not failed since the last time the bit was cleared. Software clears this bit by writing a 1 to the bit position. 1 = Indicates that the trickle current (from the main battery or trickle supply) was removed or failed. **NOTE:** Clearing CMOS in a processor-based platform may be done by using a jumper on RTCRST# or GPI, or using SAFEMODE strap. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low.	R/WC
0	AFTERG3_EN	Determines what state to go to when power is re-applied after a power failure (G3 state). This bit is in the RTC well and is not cleared by any type of reset except writes to CF9h or RTCRST#. 0 = System will return to S0 state (boot) after power is re-applied. 1 = System will return to the S5 state (except when it was in S4, in which case it will return to S4). In the S5 state, the only enabled wake event is the Power Button or any enabled wake event that was preserved through the power failure.	R/W

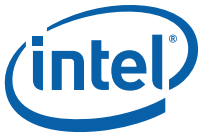
NOTE: RSMRST# is sampled using the RTC clock. Therefore, low times that are less than one RTC clock period may not be detected by the Intel® 6300ESB ICH.

8.8.1.4 Offset ACh: RST_CNT2—Reset Control 2 Register (PM—D31:F0)

Table 275. Offset ACh: RST_CNT2—Reset Control 2 Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	ACh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
		Power Well:	Resume

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23	Change Sub Class Code (CSCC)	When set, the sub class code for the SATA controller will report the value of “04h”, a RAID device. When cleared, the sub class code for the SATA controller will report the value of “01h”, an IDE controller. This bit is in the resume well and is not reset when returning from S3.	
22	Change Device ID (CDID)	When set, the Device ID for the SATA controller will report the value of “25B0h”(RAID). When cleared, the Device ID for the SATA controller will report the value of “25A3h”(hard drive). This bit is in the resume well and is not reset when returning from S3.	
21:19	Reserved	Reserved.	
18	CWORWRE - CF9 Without Resume Well Reset Enable	1 = CF9h write of 6h or Eh will not cause internal Resume Well Reset to be asserted and thus resume well logic will maintain to their states. 0 = CF9h write of 6h or Eh will reset resume well logic,	R/W
17:16	Programmable CPUTHRMTRIP# Behavior (PCTB)	This field selects the handling of CPUTHRMTRIP# event by the Intel® 6300ESB ICH internal logic. **Bits — CPUTHRMTRIP# — Behavior** 00 — CPUTHRMTRIP# — Event will cause asynchronous assertion of SLPS3#, SLPS4#, SLPS5# and immediate transition to S5 01 — CPUTHRMTRIP# — Event will be double-synchronized to RTCCLKs before causing assertion of SLPS3#, SLPS4#,SLPS5# and entry to S5 10 — CPUTHRMTRIP — Event will cause asynchronous assertion of SLPS3#, SLPS4# and SLPS5#. However, internal entry to S5 is delayed until synchronization with RTCCLK 11 — Reserved.	
15:0	Reserved	Reserved.	



8.8.1.5 Offset B8h - BBh: GPI_ROUT—GPI Routing Control Register (PM—D31:F0)

Table 276. Offset B8h - BBh: GPI_ROUT—GPI Routing Control Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	B8h - BBh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
31:2	GPI[15] through GPI[1]	See bits 1:0 for description.	
1:0	GPI0 Route	GPIO[15:0] may be routed to cause an SMI or SCI when the GPI[n]_STS bit is set. When the GPIO is not set to an input, this field has no effect. When the system is in an S1-S5 state and if the GPE0_EN bit is also set, the GPI may cause a Wake event, even if the GPI is NOT routed to cause an SMI# or SCI. Exception: If the system is in S5 state due to a powerbutton override, then the GPIs will not cause wake events, 00 = No effect. 01 = SMI# (when corresponding ALT_GP_SMI_EN bit is also set) 10 = SCI (when corresponding GPE0_EN bit is also set) 11 = Reserved	R/W

8.8.1.6 Offset C0h: MON_FWD_EN—IO Monitor Forward Enable Register (PM—D31:F0)

Note: Usage: Legacy Only.

Note: The Intel® 6300ESB ICH uses this register to enable the monitors to forward cycles to LPC, independent of the POS_DEC_EN bit and the bits that enable the monitor to generate an SMI#. The only criteria is that the address passes the decoding logic as determined by the MON[*n*]_TRP_RNG and MON_TRP_MSK register settings.

Table 277. Offset C0h: MON_FWD_EN—IO Monitor Forward Enable Register (PM—D31:F0)

Device:	31	***Function:***	0
Offset:	C0h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7	MON7_FWD_EN	0 = Disable. Cycles trapped by I/O Monitor 7 will not be forwarded to LPC. 1 = Enable. Cycles trapped by I/O Monitor 7 will be forwarded to LPC.	R/W
6	MON6_FWD_EN	0 = Disable. Cycles trapped by I/O Monitor 6 will not be forwarded to LPC. 1 = Enable. Cycles trapped by I/O Monitor 6 will be forwarded to LPC.	R/W
5	MON5_FWD_EN	0 = Disable. Cycles trapped by I/O Monitor 5 will not be forwarded to LPC. 1 = Enable. Cycles trapped by I/O Monitor 5 will be forwarded to LPC.	R/W
4	MON4_FWD_EN	0 = Disable. Cycles trapped by I/O Monitor 4 will not be forwarded to LPC. 1 = Enable. Cycles trapped by I/O Monitor 4 will be forwarded to LPC.	R/W
3:0	Reserved	Reserved.	

8.8.1.7 Offset C4h, C6h, C8h, CAh: MON[*n*]_TRP_RNG—I/O Monitor [4:7] Trap Range Register for Devices 4-7 (PM—D31:F0)

Note: Usage: Legacy Only.

These registers set the ranges that Device Monitors 4-7 should trap. Offset C4h corresponds to Monitor 4. Offset C6h corresponds to Monitor 5, etc.

When the trap is enabled in the MON_SMI register and the address is in the trap range (and passes the mask set in the MON_TRP_MSK register) the Intel® 6300ESB ICH will generate an SMI#. This SMI# occurs when the address is positively decoded by another device on PCI or by the Intel® 6300ESB ICH (because it would be forwarded to LPC or some other Intel® 6300ESB ICH internal registers). The trap ranges should not point to registers in the Intel® 6300ESB ICH's internal IDE, USB, AC'97. When the cycle is to be claimed by the Intel® 6300ESB ICH and targets one of the permitted Intel® 6300ESB ICH internal registers (interrupt controller, RTC, etc.), the cycle will complete

to the intended target and an SMI# will be generated (this is the same functionality as the Intel® 6300ESB ICH component). When the cycle is to be claimed by the Intel® 6300ESB ICH and the intended target is on LPC, an SMI# will be generated but the cycle will only be forwarded to the intended target when forwarding to LPC is enabled through the TRP_FWD_EN register settings.

Table 278. Offset C4h, C6h, C8h, CAh: MON[n]_TRP_RNG—I/O Monitor [4:7] Trap Range Register for Devices 4-7 (PM—D31:F0)

Device:	31	**Function:**	0
Offset:	C4h, C6h, C8h, CAh	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	MON[*n*]_TRAP_BASE	Base I/O locations that MON[*n*] traps (where *n* = 4, 5, 6 or 7). The range may be mapped anywhere in the CPU I/O space (0-64K). Any access to the range will generate an SMI# when enabled by the associated DEV[*n*]_TRAP_EN bit in the MON_SMI register (PMBASE +40h).	R/W

8.8.1.8 Offset CCh: MON_TRP_MSK—I/O Monitor Trap Range Mask Register for Devices 4-7 (PM—D31:F0)

Note: Usage: Legacy Only.

Table 279. Offset CCh: MON_TRP_MSK—I/O Monitor Trap Range Mask Register for Devices 4-7 (PM—D31:F0)

Device:	31	**Function:**	0
Offset:	CCh	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:12	MON7_MASK	Selects low 4-bit mask for the I/O locations that MON7 will trap. Similar to MON4_MASK.	R/W
11:8	MON6_MASK	Selects low 4-bit mask for the I/O locations that MON6 will trap. Similar to MON4_MASK.	R/W
7:4	MON5_MASK	Selects low 4-bit mask for the I/O locations that MON5 will trap. Similar to MON4_MASK.	R/W
3:0	MON4_MASK	Selects low 4-bit mask for the I/O locations that MON7 will trap. When a mask bit is set to a 1, the corresponding bit in the base I/O selection will not be decoded. For example, if MON4_TRAP_BASE = 1230h, and MON4_MSK = 0100b, the Intel® 6300ESB ICH will decode 1230h and 1234h for Monitor 4.	R/W

8.8.2 APM I/O Decode

Table 280 shows the I/O registers associated with APM support. This register space is enabled in the PCI Device 31: Function 0 space (APMDEC_EN), and cannot be moved (fixed I/O location).

Table 280. APM Register Map

Address	Mnemonic	Register Name/Function	Default	Type
B2h	APM_CNT	Advanced Power Management Control Port	00h	R/W
B3h	APM_STS	Advanced Power Management Status Port	00h	R/W

8.8.2.1 APM_CNT—Advanced Power Management Control Port Register

Note: Usage: Legacy Only.

Table 281. APM_CNT—Advanced Power Management Control Port Register

Device:	31	***Function:***	0
I/O Address:	B2h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Used to pass an APM command between the OS and the SMI handler. Writes to this port not only store data in the APMC register, but also generates an SMI# when the APMC_EN bit is set.	

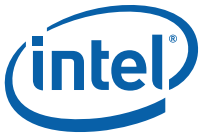
8.8.2.2 APM_STS—Advanced Power Management Status Port Register

Note: Usage: Legacy Only.

Table 282. APM_STS—Advanced Power Management Status Port Register

Device:	31	***Function:***	0
I/O Address:	B3h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Used to pass data between the OS and the SMI handler. Basically, this is a scratchpad register and is not affected by any other register or function (other than a PCI reset).	



8.8.3 Power Management I/O Registers

Table 283 shows the registers associated with ACPI and Legacy power management support. These registers are enabled in the PCI Device 31: Function 0 space (PM_IO_EN), and may be moved to any I/O location (128-byte aligned). The registers are defined to be compliant with the ACPI 1.0 specification, and use the same bit names.

Note: All reserved bits and registers will always return zero when read and will have no effect when written.

Table 283. ACPI and Legacy I/O Register Map

PMBASE+ Offset	Register Name	ACPI Pointer	Default	Attributes
00-01h	PM1 Status	PM1a_EVT_BLK	0000h	R/W
02-03h	PM1 Enable	PM1a_EVT_BLK+2	0000h	R/W
04-07h	PM1 Control	PM1a_CNT_BLK	00000000h	R/W
08-0Bh	PM1 Timer	PMTMR_BLK	00000000h	RO
10h-13h	Processor Control	P_BLK	00000000h	R/W
14h	Level 2 Register	P_BLK+4	00h	RO
28-2Bh	General Purpose Event 0 Status	GPE0_BLK	00000000h	R/W
2C-2Fh	General Purpose Event 0 Enables	GPE0_BLK+4	00000000h	R/W
30-33h	SMI# Control and Enable		00000000h	R/W
34-37h	SMI Status Register		00000000h	R/W
38-39h	Alternate GPI SMI Enable		0000h	R/W
3A-3Bh	Alternate GPI SMI Status		0000h	R/WC
40h	Device Monitor SMI Status and Enable		0000h	R/W
44h	Device Activity Status		0000h	R/W
48h	Device Trap Enable Register		0000h	R/W
4Ch-4Dh	Bus Address Tracker		Last Cycle	RO
4Eh	Bus Cycle Tracker		Last Cycle	RO
60h-7Fh	Reserved for TCO Registers			

8.8.3.1 PM1_STS—Power Management 1 Status Register

Note: Usage: ACPI or Legacy

Note: When bit 10 or 8 in this register is set, and the corresponding _EN bit is set in the PM1_EN register, the Intel® 6300ESB ICH will generate a Wake Event. Once back in an S0 state (or if already in an S0 state when the event occurs), the Intel® 6300ESB ICH will also generate an SCI when the SCI_EN bit is set, or an SMI# when the SCI_EN bit is not set.

Note: Bit 5 does not cause an SMI# or a wake event. Bit 0 does not cause a wake event but may cause an SMI# or SCI.

Table 284. PM1_STS—Power Management 1 Status Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	PMBASE + 00h (ACPI PM1a_EVT_BLK)	***Attribute:***	Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Bits 0-7: Core; Bits 8-10, 12-15;Resume; Bit 11: RTC

Bits	Name	Description	Access
15	WAK_STS: Wake Status	This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#. 0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the system is in one of the sleep states (through the SLP_EN bit) and an enabled wake event occurs. Upon setting this bit, the Intel® 6300ESB ICH will transition the system to the ON state. When the AFTERG3_EN bit is not set and a power failure (such as removed batteries) occurs without the SLP_EN bit set, the system will return to an S0 state when power returns, and the WAK_STS bit will not be set. When the AFTERG3_EN bit is set and a power failure occurs without the SLP_EN bit having been set, the system will go into an S5 state when power returns, and a subsequent wake event will cause the WAK_STS bit to be set. Note that any subsequent wake event would have to be caused by either a Power Button press, or an enabled wake event that was preserved through the power failure (enable bit in the RTC well).	R/WC
14:12	Reserved	Reserved.	
11	PRBTNOR_STS: Power Button Override Status	This bit is set any time a Power Button Override occurs (I.E. the power button is pressed for at least four consecutive seconds), or due to the corresponding bit in the SMBus slave message. The power button override causes an unconditional transition to the S5 state, as well as sets the AFTERG3 bit. The BIOS or SCI handler clears this bit by writing a 1 to it. This bit is not affected by hard resets through CF9h writes, and is not reset by RSMRST#. Thus, this bit is preserved through power failures.	R/WC

Table 284. PM1_STS—Power Management 1 Status Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 00h (ACPI PM1a_EVT_BLK)	**Attribute:**	Read/Write Clear
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Bits 0-7: Core; Bits 8-10, 12-15;Resume; Bit 11: RTC

Bits	Name	Description	Access
10	RTC_STS: RTC Status	This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#. 0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the RTC generates an alarm (assertion of the IRQ8# signal). Additionally when the RTC_EN bit is set, the setting of the RTC_STS bit will generate a wake event.	R/WC
9	Reserved	Reserved.	
8	PWRBTN__STS: Power Button Status	This bit is not affected by hard resets caused by a CF9 write. 0 = When the PWRBTN# signal is held low for more than four seconds, the hardware clears the PWRBTN_STS bit, sets the PWRBTNOR_STS bit, and the system transitions to the S5 state with only PWRBTN# enabled as a wake event. This bit may be cleared by software by writing a one to the bit position. 1 = This bit is set when the PWRBTN# signal is asserted (low), independent of any other enable bit. See PWRBTN_EN for the effect when PWRBTN_STS goes active. PWRBTN_STS is always a wake event. This bit is only set by hardware and can be cleared by software writing a one to this bit position. This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#.	R/WC
7:6	Reserved	Reserved.	
5	GBL _STS: Global Status	0 = The SCI handler should then clear this bit by writing a 1 to the bit location. 1 = Set when an SCI is generated due to BIOS wanting the attention of the SCI handler. BIOS has a corresponding bit, BIOS_RLS, which will cause an SCI and set this bit.	R/WC
4:1	Reserved	Reserved.	
0	TMROF_STS: Timer Overflow Status	0 = The SCI or SMI# handler clears this bit by writing a 1 to the bit location. 1 = This bit gets set any time bit 22 of the 24-bit timer goes high (bits are numbered from 0 to 23). This will occur every 2.3435 seconds. When the TMROF_EN bit is set, then the setting of the TMROF_STS bit will additionally generate an SCI or SMI# (depending on the SCI_EN).	R/WC

8.8.3.2 PM1_EN—Power Management 1 Enable Register

Note: Usage: ACPI or Legacy.

Table 285. PM1_EN—Power Management 1 Enable Register

Device:	31	***Function:***	0
I/O Address:	PMBASE + 02h (ACPI PM1a_EVT_BLK + 2)	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Bits 0-7: Core Bits 8-9, 11-15: Resume Bit 10: RTC

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10	RTC_EN: RTC Event Enable	This bit is in the RTC well to allow an RTC event to wake after a power failure. This bit is not cleared by any reset other than RTCRST# or a Power Button Override event. 0 = No SCI (or SMI#) or wake event is generated then RTC_STS goes active. 1 = An SCI (or SMI#) and a wake event will occur when this bit is set and the RTC_STS bit goes active.	R/W
9	Reserved	Reserved.	
8	PWRBTN_EN	This bit is the power button enable. It works in conjunction with the SCI_EN bit: PWRBTN_EN SCI_EN Effect when PWRBTN_STS is set 0 x No SMI# or SCI 1 0 SMI# 1 1 SCI PWRBTN_EN has no effect on the PWRBTN_STS bit being set by the assertion of the power button. The Power Button is always enabled as a Wake event.	
7:6	Reserved	Reserved.	
5	GBL_EN: Global Enable	When both the GBL_EN and the GBL_STS are set, an SCI is raised. Disable. Enable SCI on GBL_STS going active.	R/W
4:1	Reserved	Reserved.	
0	TMROF_EN. Timer Overflow Interrupt Enable	Works in conjunction with the SCI_EN bit as described below: TMROF_EN SCI_EN Effect when TMROF_STS is set 0 x No SMI# or SC. If system was in S1-S5, no wake event 1 0 SMI# If system was in S1-S5, then a wake vent occurs before the SMI# 1 1 SCI If system was in S1-S5, then a wake vent occurs before the SMI#	R/W

8.8.3.3 PM1_CNT—Power Management 1 Control

Note: Usage: ACPI or Legacy.

Table 286. PM1_CNT—Power Management 1 Control

Device:	31	***Function:***	0
I/O Address:	PMBASE + 04h (ACPI PM1a_CNT_BLK)	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Bits 0-7: Core, Bits 8-12: RTC Bits 13-15: Resume

Bits	Name	Description	Access
31:14	Reserved	Reserved.	
13	SLP_EN: Sleep Enable	Setting this bit causes the system to sequence into the Sleep state defined by the SLP_TYP field.	WO
12:10	SLP_TYP: Sleep Type	This 3-bit field defines the type of Sleep the system should enter when the SLP_EN bit is set to 1. 000 = ON: Typically maps to S0 state. 001 = Asserts STPCLK#. Puts CPU in Stop-Grant state. Optional to assert CPUSLP# to put CPU in sleep state: Typically maps to S1 state. 010 = Reserved 011 = Reserved 100 = Reserved 101 = Suspend-To-RAM. Assert SLP_S1# and SLP_S3#: Typically maps to S3 state. 110 = Suspend-To-Disk. Assert SLP_S1#, SLP_S3#, and SLP_S4#: Typically maps to S4 state. 111 = Soft Off. Assert SLP_S1#, SLP_S3#, SLP_S4#, and SLP_S5#: Typically maps to S5 state. **NOTE:** These bits are only reset by RTCRST#.	R/W
9:3	Reserved	Reserved.	
2	GBL_RLS: Global Release	0 = This bit always reads as 0. 1 = ACPI software writes a 1 to this bit to raise an event to the BIOS. BIOS software has a corresponding enable and status bits to control its ability to receive ACPI events.	WO
1	Reserved	Reserved.	
0	SCI_EN: SCI Enable	Selects the SCI interrupt or the SMI# interrupt for various events including the bits in the PM1_STS register (bit 10, 8, 0), and bits in GPE0_STS. 0 = These events will generate an SMI#. 1 = These events will generate an SCI.	R/W

8.8.3.4 PM1_TMR—Power Management 1 Timer Register

Note: Usage: ACPI only.

Table 287. PM1_TMR—Power Management 1 Timer Register

Device:	31	***Function:***	0
I/O Address:	PMBASE + 08h (ACPI PMTMR_BLK)	***Attribute:***	Read-Only
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23:0	TMR_VAL: Timer Value	Returns the running count of the PM timer. This counter runs off a 3.579545 MHz clock (14.31818 MHz divided by 4). It is reset to zero during a PCI reset, and then continues counting as long as the system is in the S0 state. Anytime bit 22 of the timer goes HIGH to LOW (bits referenced from 0 to 23), the TMROF_STS bit is set. The High-to-Low transition will occur every 2.3435 seconds. When the TMROF_EN bit is set, an SCI interrupt is also generated.	RO

8.8.3.5 PROC_CNT—Processor Control Register

Note: Usage: ACPI or Legacy.

Table 288. PROC_CNT—Processor Control Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	PMBASE + 10h (ACPI P_BLK)	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No (bits 7:5 are write once)	***Power Well:***	Core

Bits	Name	Description	Access
31:18	Reserved	Reserved.	
17	THTL_STS: Throttle Status	0 = No clock throttling is occurring (maximum processor performance). 1 = Indicates that the clock state machine is in some type of low power state (where the processor is not running at its maximum performance): thermal throttling or hardware throttling.	RO
16:9	Reserved	Reserved.	
8	FORCE_THTL: Force Thermal Throttling	Software may set this bit to force the thermal throttling function. This has the same effect as the THRM# signal being active for 2 seconds. 0 = No forced throttling. 1 = Throttling at the duty cycle specified in THRM_DTY starts immediately (no 2 second delay), and no SMI# is generated.	R/W

Table 288. PROC_CNT—Processor Control Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 10h (ACPI P_BLK)	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No (bits 7:5 are write once)	**Power Well:**	Core

Bits	Name	Description	Access
7:5	THRM_DTY	This write-once 3-bit field determines the duty cycle of the throttling when the thermal override condition occurs. The duty cycle indicates the approximate percentage of time the STPCLK# signal is asserted while in the throttle mode. The STPCLK# throttle period is 1024 PCICLKs. Note that the throttling only occurs when the system is in the C0 state. When in the C2 state, no throttling occurs. There is no enable bit for thermal throttling, because it should not be disabled. Once the THRM_DTY field is written, any subsequent writes will have no effect until PXPCIRST# goes active. **THRM_DTY Throttle ModePCI Clocks** 000 50% Default 512 001 87.5% 896 010 75.0% 768 011 62.5% 640 100 50% 512 101 37.5% 384 110 25% 256 111 12.5% 128	
4	THTL_EN	When set and the system is in a C0 state, it enables a processor-controlled STPCLK# throttling. The duty cycle is selected in the THTL_DTY field. 0 = Disable 1 = Enable	
3:1	THTL_DTY	This 3-bit field determines the duty cycle of the throttling when the THTL_EN bit is set. The duty cycle indicates the approximate percentage of time the STPCLK# signal is asserted (low) while in the throttle mode. The STPCLK# throttle period is 1024 PCICLKs. **THRM_DTY Throttle ModePCI Clocks** 000 50% Default 512 001 87.5% 896 010 75.0% 768 011 62.5% 640 100 50% 512 101 37.5% 384 110 25% 256 111 12.5% 128	
0	Reserved	Reserved.	

8.8.3.6 LV2—Level 2 Register

Note: Usage: ACPI or Legacy.

Table 289. LV2—Level 2 Register

Device:	31	***Function:***	0
I/O Address:	PMBASE + 14h (ACPI P_BLK + 4)	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Reads to this register return all zeros, writes to this register have no effect. Reads to this register generate a “enter a level 2 power state” (C2) to the clock control logic. This will cause the STPCLK# signal to go active, and stay active until a break event occurs. Throttling (due either to THTL_EN or THRM# override) will be ignored.	RO

8.8.3.7 GPE0_STS—General Purpose Event 0 Status Register

Note: This register is symmetrical to the General Purpose Event 0 Enable Register. When the corresponding _EN bit is set, and the _STS bit get set, the Intel® 6300ESB ICH will generate a Wake Event. Once back in an S0 state (or if already in an S0 state when the event occurs), the Intel® 6300ESB ICH will also generate an SCI when the SCI_EN bit is set, or an SMI# when the SCI_EN bit is not set. There will be no SCI/SMI# or wake event on THRMOR_STS since there is no corresponding _EN bit. None of these bits are reset by CF9h write. All are reset by RSMRST#.

Note: Usage: ACPI.

Table 290. GPE0_STS—General Purpose Event 0 Status Register (Sheet 1 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 28h (ACPI PGPE0_BLK)	**Attribute:**	Read/Write Clear
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
31:16	GPIn_STS	These bits are set any time the corresponding GPIO is set up as an input and the corresponding GPIO signal is high (or low when the corresponding GP_INV bit is set). When the corresponding enable bit is set in the GPE0_EN register, then the GPI[n]_STS bit is set: • When the system is in an S1-S5 state, the event will also wake the system. • When the system is in an S0 state (or upon waking back to an S0 state), a SCI will be caused depending on the GPI_ROUT bits for the corresponding GPI. **NOTE:** These bits are sticky and are cleared by writing a 1 back to this bit position. Corresponding bits and GPIOs **Bit # / GPI[n]Bit # / GPI[n]** 16 / 0 / 24 / 8 17 / 1 / 25 / na 18 / 2 / 26 / na 19 / 3 / 27 / 11 20 / 4 / 28 / 12 21 / 5 / 29 / 13 22 / 6 / 30 / na 23 / 7 / 31 / na	R/WC
15:14	Reserved	Reserved.	
13	PME_B0_STS	This bit will be set to 1 by the Intel® 6300ESB ICH when any internal device on bus 0 asserts the equivalent of the PME# signal. Additionally, when the PME_B0_EN bit is set, and the system is in an S0 state, the setting of the PME_B0_STS bit will generate an SCI (or SMI# when SCI_EN is not set). When the PME_B0_STS bit is set, and the system is in an S1-S4 state (or S5 state due to SLP_TYP and SLP_EN), the setting of the PME_B0_STS bit will generate a wake event, and an SCI (or SMI# when SCI_EN is not set) will be generated. When the system is in an S5 state due to power button override, the PME_B0_STS bit will not cause a wake event or SCI. The default for this bit is 0. Writing a 1 to this bit position clears this bit.	R/WC
12	Reserved	Reserved.	

Table 290. GPE0_STS—General Purpose Event 0 Status Register (Sheet 2 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 28h (ACPI PGPE0_BLK)	**Attribute:**	Read/Write Clear
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
11	PME_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the PME# signal goes active. Additionally, when the PME_EN bit is set and the system is in an S0 state, then the setting of the PME_STS bit will generate an SCI or SMI# (when SCI_EN is not set). When the PME_EN bit is set and the system is in an S1-S4 state (or S5 state due to setting SLP_TYP and SLP_EN), then the setting of the PME_STS bit will generate a wake event, and an SCI (or SMI# if SMI_EN is not set) will be generated. When the system is in an S5 state due to power button override, PME_STS will not cause a wake event or SCI.	R/WC
10:9	Reserved	Reserved.	
8	RI_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the RI# input signal goes active.	R/WC
7	SMBus Wake Status (SMB_WAK_STS)	The SMBus controller may independently cause an SMI# or SCI, so this bit does not need to do so (unlike the other bits in this register). 0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware to indicate that the wake event was caused by the Intel® 6300ESB ICH's SMBus logic.This bit will be set by the WAKE/SMI# command type, even when the system is already awake. The SMI handler should then clear this bit. **NOTE:** This bit is set by the SMBus slave command 01h (Wake/SMI#) even when the system is in the S0 state. Therefore, to avoid an instant wake on subsequent transitions to sleep states, software must clear this bit after each reception of the Wake/SMI# command or just prior to entering the sleep state. **NOTE:** When SMB_WAK_STS is set due to SMBus slave receiving a message, it will be cleared by internal logic when a THRMTRIP# event happens or a Power Button Override event. However, THRMTRIP# or Power Button Override event will not clear SMB_WAK_STS when it is set due to SMBALERT# signal going active.	R/WC
6	TCOSCI_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the TCO logic causes an SCI.	R/WC

Table 290. GPE0_STS—General Purpose Event 0 Status Register (Sheet 3 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 28h (ACPI PGPE0_BLK)	**Attribute:**	Read/Write Clear
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
5	AC97_STS	This bit will be set to 1 by when the codecs are attempting to wake the system and the PME events for the codecs are armed for wakeup. A PME is armed by programming the appropriate PMEE bit in the Power Management Control and Status register at bit 8 of offset 54h in each AC'97 function. 0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when the codecs are attempting to wake the system. The AC97_STS bit gets set only from the following two cases: 1. The PMEE bit for the function is set, and the AC-link bit clock has been shut and the routed AC_SDIN line is high (for audio, when routing is disabled, no wake events are allowed). 2. For modem, when audio routing is disabled, the wake event is an OR of all AC_SDIN lines. When routing is enabled, the wake event for modem is the remaining non-routed AC_SDIN line), or o GPI Status Change Interrupt bit (NABMBAR + 30h, bit 0) is 1. This bit is not affected by a hard reset caused by a CF9h write.	R/WC
4	USB2_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when USB UHCI Controller 2 needs to cause a wake. Wake event will be generated when the corresponding USB2_EN bit is set.	R/WC
3	USB1_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware when USB UHCI Controller 1 needs to cause a wake. Wake event will be generated when the corresponding USB1_EN bit is set.	R/WC
2	Reserved	Reserved.	
1	Thermal Interrupt Override Status (THRMOR_STS)	0 = Software clears this bit by writing a 1 to the bit position. 1 = This bit is set by hardware anytime a thermal over-ride condition occurs and starts throttling the processor's clock at the THRM_DTY ratio. This will not cause an SMI#, SCI, or wake event.	R/WC
0	Thermal Interrupt Status (THRM_STS)	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by hardware anytime the THRM# signal is driven active as defined by the THRM_POL bit. Additionally, when the THRM_EN bit is set, the setting of the THRM_STS bit will also generate a power management event (SCI or SMI#).	R/WC

8.8.3.8 GPE0_EN—General Purpose Event 0 Enables Register

Note: This register is symmetrical to the General Purpose Event 0 Status Register. All the bits in this register should be cleared to 0 based on a Power Button Override. The resume well bits are all cleared by RSMRST#. The RTC well bits are cleared by RTCRST#.

Note: Usage: ACPI.

Table 291. GPE0_EN—General Purpose Event 0 Enables Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	PMBASE + 2Ch (ACPI GPE0_BLK + 4)	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Bits 0-7, 12, 16-31 Resume, Bits 8-11, 13 RTC

Bits	Name	Description	Access
31:16	GPIn_EN	These bits enable the corresponding GPI[n]_STS bits being set to cause a SCI, and/or wake event. These bits are cleared by RSMRST#.	R/W
15:14	Reserved	Reserved.	
13	PME_B0_EN	Enables the setting of the PME_B0_STS bit to generate a wake event and/or an SCI or SMI#. PME_B0_STS may be a wake event from the S1-S4 states, or from S5 (when entered through SLP_TYP and SLP_EN) or power failure, but not Power Button Override. This bit defaults to 0. It is only cleared by Software or RTCRST#. It is not cleared by CF9h writes.	R/W
12	Reserved	Reserved.	
11	PME_EN	This bit defaults to 0. It is only cleared by Software or RTCRST#. It is not cleared by CF9h writes. 0 = Disable. 1 = Enables the setting of the PME_STS to generate a wake event and/or an SCI. PME# may be a wake event from the S1 - S4 state or from S5 (when entered through SLP_EN or power failure, but not power button override).	R/W
10	Reserved	Reserved.	
9	Reserved	Reserved.	
8	RI_EN	The value of this bit will be maintained through a G3 state and is not affected by a hard reset caused by a CF9h write. 0 = Disable. 1 = Enables the setting of the RI_STS to generate a wake event.	R/W
7	Reserved	Reserved.	
6	TCOSCI_EN	0 = Disable. 1 = Enables the setting of the TCOSCI_STS to generate an SCI.	R/W
5	AC97_EN	0 = Disable. 1 = Enables the setting of the AC97_STS to generate a wake event.	R/W

Table 291. GPE0_EN—General Purpose Event 0 Enables Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 2Ch (ACPI GPE0_BLK + 4)	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Bits 0-7, 12, 16-31 Resume, Bits 8-11, 13 RTC

Bits	Name	Description	Access
4	USB2_EN	0 = Disable. 1 = Enable the setting of the USB2_STS bit to generate a wake event. The USB2_STS bit is set anytime USB UHCI Controller #2 signals a wake event. Break events are handled through the USB interrupt.	R/W
3	USB1_EN	0 = Disable. 1 = Enable the setting of the USB1_STS bit to generate a wake event. The USB1_STS bit is set anytime USB UHCI Controller #1 signals a wake event. Break events are handled through the USB interrupt.	R/W
2	THRM#_POL	This bit controls the polarity of the THRM# pin needed to set the THRM_STS bit. 0 = Low value on the THRM# signal will set the THRM_STS bit. 1 = HIGH value on the THRM# signal will set the THRM_STS bit.	R/W
1	Reserved	Reserved.	
0	THRM_EN	0 = Disable. 1 = Active assertion of the THRM# signal (as defined by the THRM_POL bit) will set the THRM_STS bit and generate a power management event (SCI or SMI).	R/W

8.8.3.9 SMI_EN—SMI Control and Enable Register

Note: Usage: ACPI or Legacy

Table 292. SMI_EN—SMI Control and Enable Register (Sheet 1 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 30h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:19	Reserved	Reserved.	
18	INTEL_USB2_EN	Enables Intel-Specific USB EHCI SMI logic to cause SMI#.	
17	LEGACY_USB2_EN	Enables legacy USB EHCI logic to cause SMI#.	
16:15	Reserved	Reserved.	

Table 292. SMI_EN—SMI Control and Enable Register (Sheet 2 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 30h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
14	PERIODIC_EN	0 = Disable. 1 = Enables the Intel® 6300ESB ICH to generate an SMI# when the PERIODIC_STS bit is set in the SMI_STS register.	R/W
13	TCO_EN	0 = Disables TCO logic generating an SMI#. Note that when the NMI2SMI_EN bit is set, SMIs that are caused by re-routed NMIs will not be gated by the TCO_EN bit. Even when the TCO_EN bit is 0, NMIs will still be routed to cause SMIs. 1 = Enables the TCO logic to generate SMI#. **NOTE:** This bit can not be written once the TCO_LOCK bit (at offset 08h of TCO I/O Space) is set. This prevents unauthorized software from disabling the generation of TCO-based SMI's	R/W
12	Reserved	Reserved.	
11	MCSMI_EN: Microcontroller SMI Enable	0 = Disable. 1 = Enables the Intel® 6300ESB ICH to trap accesses to the microcontroller range (62h or 66h) and generate an SMI#. Note that "trapped' cycles will be claimed by the Intel® 6300ESB ICH on PCI, but not forwarded to LPC.	R/W
10:8	Reserved	Reserved.	
7	BIOS_RLS: BIOS Release	0 = This bit will always return 0 on reads. Writes of 0 to this bit have no effect. 1 = Enables the generation of an SCI interrupt for ACPI software when a one is written to this bit position by BIOS software.	WO
6	SWSMI_TMR_EN: Software SMI# Timer Enable	0 = Disable. Clearing the SWSMI_TMR_EN bit before the timer expires will reset the timer and the SMI# will not be generated. 1 = Starts Software SMI# Timer. When the SWSMI timer expires (the timeout period depends upon the SWSMI_RATE_SEL bit setting), SWSMI_TMR_STS is set and an SMI# is generated. SWSMI_TMR_EN stays set until cleared by software.	R/W
5	APMC_EN	0 = Disable. Writes to the APM_CNT register will not cause an SMI#. 1 = Enables writes to the APM_CNT register to cause an SMI#.	R/W
4	SLP_SMI_EN	0 = Disables the generation of SMI# on SLP_EN. Note that this bit must be 0 before the software attempts to transition the system into a sleep state by writing a 1 to the SLP_EN bit. 1 = A write of 1 to the SLP_EN bit (bit 13 in PM1_CNT register) will generate an SMI#, and the system will not transition to the sleep state based on that write to the SLP_EN bit.	R/W
3	LEGACY_USB_EN	0 = Disable. 1 = Enables legacy USB circuit to cause SMI#.	R/W

Table 292. SMI_EN—SMI Control and Enable Register (Sheet 3 of 3)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 30h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	BIOS_EN	0 = Disable. 1 = Enables the generation of SMI# when ACPI software writes a 1 to the GBL_RLS bit.	R/W
1	EOS: End of SMI	This bit controls the arbitration of the SMI signal to the processor. This bit must be set for the Intel® 6300ESB ICH to assert SMI# low to the processor. 0 = Once the Intel® 6300ESB ICH asserts SMI# low, the EOS bit is automatically cleared. 1 = When this bit is set, SMI# signal will be deasserted for 4 PCI clocks before its assertion. In the SMI handler, the processor should clear all pending SMIs (by servicing them and then clearing their respective status bits), set the EOS bit, and exit SMM. This will allow the SMI arbiter to re-assert SMI upon detection of an SMI event and the setting of a SMI status bit. **NOTE:** The Intel® 6300ESB ICH is able to generate 1st SMI after reset even though EOS bit is not set. Subsequent SMI require EOS bit is set.	R/W (special)
0	GBL_SMI_EN	0 = No SMI# will be generated by the Intel® 6300ESB ICH. This bit is reset by a PCI reset event. 1 = Enables the generation of SMI# in the system upon any enabled SMI event.	R/W

8.8.3.10 SMI_STS—SMI Status Register

Note: When the corresponding _EN bit is set when the _STS bit is set, the Intel® 6300ESB ICH will cause an SMI# (except bits 8-10 and 12, which do not need enable bits since they are logically ORed with other registers that have enable bits). The Intel® 6300ESB ICH uses the same GPE0_EN register (I/O address: PMBase+2Ch) to enable/disable both SMI and ACPI SCI general purpose input events. ACPI OS assumes that it owns the entire GPE0_EN register per ACPI spec. Problems arise when some of the general-purpose inputs are enabled as SMI by BIOS, and some of the general purpose inputs are enabled for SCI. In this case ACPI OS turns off the enabled bit for any GPIx input signals that are not indicated as SCI general-purpose events at boot, and exit from sleeping states. BIOS should define a dummy control method which prevents the ACPI OS from clearing the SMI GPE0_EN bits.

Note: Usage: ACPI or Legacy.

Table 293. SMI_STS—SMI Status Register (Sheet 1 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 34h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23	WDT_SMI_STS	0 = SMI# not caused by WDT 1st timeout. 1 = Indicates the SMI# was caused by the WDT 1st timeout.	
22:19	Reserved	Reserved.	
18	INTEL_USB2_STS	This non-sticky read-only bit is a logical OR of each of the SMI status bits in the Intel-Specific USB EHCI SMI Status Register ANDed with the corresponding enable bits. This bit will not be active when the enable bits are not set. Writes to this bit will have no effect.	
17	LEGACY_USB2_STS	This non-sticky read-only bit is a logical OR of each of the SMI status bits in the USB EHCI Legacy Support Register ANDed with the corresponding enable bits. This bit will not be active when the enable bits are not set. Writes to this bit will have no effect.	
16	SMBus SMI Status (SMBUS_SMI_STS)	0 = This bit is cleared by writing a 1 to its bit position. This bit is set from the 64 KHz clock domain used by the SMBus. Software must wait at least 15.63 us after the initial assertion of this bit before clearing it. 1 = Indicates that the SMI# was caused by: 1. The SMBus Slave receiving a message, or 2. The SMBALERT# signal goes active and the SMB_SMI_EN bit is set and the SMBALERT_DIS bit is cleared, or 3. The SMBus Slave receiving a Host Notify message and the HOST_NOTIFY_INTREN and the SMB_SMI_EN bits are set, or 4. The Intel® 6300ESB ICH detecting the SMLINK_SLAVE_SMI command while in the S0 state.	R/WC
15	SERIRQ_SMI_STS	0 = SMI# was not caused by SERIRQ decoder. This is not a sticky bit. 1 = Indicates that the SMI# was caused by the SERIRQ decoder.	RO
14	PERIODIC_STS	0 = This bit is cleared by writing a 1 to its bit position. 1 = This bit will be set at the rate determined by the PER_SMI_SEL bits. When the PERIODIC_EN bit is also set, the Intel® 6300ESB ICH will generate an SMI#.	R/WC
13	TCO_STS	0 = SMI# not caused by TCO logic. 1 = Indicates the SMI# was caused by the TCO logic. Note that this is not a wake event.	RO
12	DEVMON_STS: Device Monitor Status	0 = SMI# not caused by Device Monitor. 1 = Set under any of the following conditions: - Any of the DEV[7:4]_TRAP_STS bits are set and the corresponding DEV[7:4]_TRAP_EN bits are also set. - Any of the DEVTRAP_STS bits are set and the corresponding DEVTRAP_EN bits are also set.	RO

Table 293. SMI_STS—SMI Status Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE + 34h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
11	MCSMI_STS: Microcontroller SMI# Status	0 = Indicates that there has been no access to the power management microcontroller range (62h or 66h). This bit is cleared by software writing a 1 to the bit position. 1 = Set when there has been an access to the power management microcontroller range (62h or 66h). When this bit is set, and the MCSMI_EN bit is also set, the Intel® 6300ESB ICH will generate an SMI#.	R/WC
10	GPE1_STS	This bit is a logical OR of the bits in the ALT_GP_SMI_STS register that are also set up to cause an SMI# (as indicated by the GPI_ROUT registers) and have the corresponding bit set in the ALT_GP_SMI_EN register. Bits that are not routed to cause an SMI# will have no effect on this bit. 0 = SMI# was not generated by a GPI assertion. 1 = SMI# was generated by a GPI assertion.	RO
9	GPE0_STS	This bit is a logical OR of the bits in the GPE0_STS register that also have the corresponding bit set in the GPE0_EN register. 0 = SMI# was not generated by a GPE0 event. 1 = SMI# was generated by a GPE0 event.	R/WC
8	PM1_STS_REG	This is an ORs of the bits in the ACPI PM1 Status Register (offset PMBASE+00h) that may cause an SMI#. 0 = SMI# was not generated by a PM1_STS event. 1 = SMI# was generated by a PM1_STS event.	R/WC
7	Reserved	Reserved.	
6	SWSMI_TMR_STS	1 = Set by the hardware when the Software SMI# Timer expires. 0 = Software clears this bit by writing a 1 to the bit location.	R/WC
5	APM_STS	0 = Software clears this bit by writing a 1 to the bit location. 1 = SMI# was generated by a write access to the APM control register with the APMC_EN bit set.	R/WC
4	SLP_SMI_STS	0 = Software clears this bit by writing a 1 to the bit location. 1 = Indicates an SMI# was caused by a write of 1 to SLP_EN bit when SLP_SMI_EN bit is also set.	R/WC
3	LEGACY_USB_STS	This bit is a logical OR of each of the SMI status bits in the USB Legacy Keyboard/Mouse Control Registers ANDed with the corresponding enable bits. This bit will not be active When the enable bits are not set. 0 = SMI# was not generated by USB Legacy event. 1 = SMI# was generated by USB Legacy event.	RO
2	BIOS_STS	0 = This bit cleared by software writing a 1 to its bit position. 1 = SMI# was generated due to ACPI software requesting attention (writing a 1 to the GBL_RLS bit with the BIOS_EN bit set).	R/WC
1:0	Reserved	Reserved.	

8.8.3.11 ALT_GP_SMI_EN—Alternate GPI SMI Enable Register

Note: Usage: ACPI or Legacy.

Table 294. ALT_GP_SMI_EN—Alternate GPI SMI Enable Register

Device:	31	***Function:***	0
I/O Address:	PMBASE +38h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
15:0		These bits are used to enable the corresponding GPIO to cause an SMI#. In order for these bits to have any effect, the following must be true. • The corresponding bit in the ALT_GP_SMI_EN register is set. • The corresponding GPI must be routed in the GPI_ROUT register to cause an SMI. • The corresponding GPIO must be implemented.	

8.8.3.12 ALT_GP_SMI_STS—Alternate GPI SMI Status Register

Note: Usage: ACPI or Legacy.

Table 295. ALT_GP_SMI_STS—Alternate GPI SMI Status Register

Device:	31	***Function:***	0
I/O Address:	PMBASE +3Ah	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
15:0		These bits report the status of the corresponding GPIs. 1 = active, 0 = inactive. **These bits are sticky.** When the following conditions are true, an SMI# will be generated and the GPE0_STS bit set: • The corresponding bit in the ALT_GPI_SMI_EN register is set. • The corresponding GPI must be routed in the GPI_ROUT register to cause an SMI. • The corresponding GPIO must be implemented. All bits are in the resume well. Default for these bits is dependent on the state of the GPI pins.	

8.8.3.13 MON_SMI—Device Monitor SMI Status and Enable Register

Note: Usage: Legacy only.

Table 296. MON_SMI—Device Monitor SMI Status and Enable Register

Device:	31	***Function:***	0
I/O Address:	PMBASE +40h	***Attribute:***	Read/Write, Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:12	DEV[7:4]_TRAP_STS	Bit 12 corresponds to Monitor 4, bit 13 corresponds to Monitor 5 etc. 0 = SMI# was not caused by the associated device monitor. 1 = SMI# was caused by an access to the corresponding device monitor's I/O range.	R/WC
11:8	DEV[7:4]_TRAP_EN	Bit 8 corresponds to Monitor 4, bit 9 corresponds to Monitor 5 etc. 0 = Disable. 1 = Enables SMI# due to an access to the corresponding device monitor's I/O range.	R/W
7:0	Reserved	Reserved.	

8.8.3.14 DEVACT_STS—Device Activity Status Register

Note: This register is used in conjunction with the Periodic SMI# timer to detect any system activity for legacy power management.

Each bit indicates if the an access has occurred to the corresponding device's trap range, or for bits 6:9 if the corresponding PCI interrupt is active. Write 1 to the same bit position to clear it. This register is used by APM power management software to see if there has been system activity. The periodic SMI# timer indicates if it is the right time to read the DEVTRAP_STS register.

Note: Usage: Legacy only.

Table 297. DEVACT_STS—Device Activity Status Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	PMBASE +44h	***Attribute:***	Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13	ADLIB_ACT_STS	Ad-Lib. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location. **NOTE:** This bit is no longer supported and will not be validated.	R/WC
12	KBC_ACT_STS	KBC (60/64h). 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location. **NOTE:** If bit 7 of ETR1 (D31:F0, offset F4h, section 9.1.35) is set. Ports 60/64h accesses initiated from an external PCI agent will not set this bit.	R/WC
11	MIDI_ACT_STS	MIDI. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location. **NOTE:** This bit is no longer supported and will not be validated.	R/WC
10	AUDIO_ACT_STS	Audio (Sound Blaster "OR'd" with MSS). 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location. **NOTE:** This bit is no longer supported and will not be validated.	R/WC
9	PIRQDH_ACT_STS	PIRQ[D or H]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.	R/WC

Table 297. DEVACT_STS—Device Activity Status Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	PMBASE +44h	**Attribute:**	Read/Write Clear
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
8	PIRQCG_ACT_STS	PIRQ[C or G]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.	R/WC
7	PIRQBF_ACT_STS	PIRQ[B or F]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.	R/WC
6	PIRQAE_ACT_STS	PIRQ[A or E]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.	R/WC
5	LEG_ACT_STS	Parallel Port, Serial Port 1, Serial Port 2, Floppy Disk Controller. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.	R/WC
4	Reserved	Reserved.	
3	IDES1_ACT_STS	IDE Secondary Drive 1. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.	R/WC
2	IDES0_ACT_STS	IDE Secondary Drive 0. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.	R/WC
1	IDEP1_ACT_STS	IDE Primary Drive 1. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.	R/WC
0	IDEP0_ACT_STS	IDE Primary Drive 0. 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.	R/WC

8.8.3.15 DEVTRAP_EN— Device Trap Enable Register

Note: This register enables the individual trap ranges to generate an SMI# when the corresponding status bit in the DEVACT_STS register is set. When a range is enabled, I/O cycles associated with that range will not be forwarded to LPC or IDE.

Note: Usage: Legacy only.

Table 298. DEVTRAP_EN— Device Trap Enable Register

Device:	31	***Function:***	0
I/O Address:	PMBASE +48h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13	ADLIB_TRP_EN	Ad-Lib. 0 = Disable. 1 = Enable. **NOTE:** This bit is no longer supported and will not be validated.	R/W
12	KBC_TRP_EN	KBC (60/64h). 0 = Disable. 1 = Enable.	R/W
11	MIDI_TRP_EN	MIDI. 0 = Disable. 1 = Enable. **NOTE:** This bit is no longer supported and will not be validated.	R/W
10	AUDIO_TRP_EN	Audio (Sound Blaster “ORed” with MSS). 0 = Disable. 1 = Enable. **NOTE:** This bit is no longer supported and will not be validated.	R/W
9:6	Reserved	Reserved.	
5	LEG_IO_TRP_EN	Parallel Port, Serial Port 1, Serial Port 2, Floppy Disk Controller. 0 = Disable. 1 = Enable.	R/W
4	Reserved	Reserved.	
3	IDES1_TRP_EN	IDE Secondary Drive 1. 0 = Disable. 1 = Enable.	R/W

Table 298. DEVTRAP_EN— Device Trap Enable Register

Device:	31	***Function:***	0
I/O Address:	PMBASE +48h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
2	IDES0_TRP_EN	IDE Secondary Drive 0. 0 = Disable. 1 = Enable.	R/W
1	IDEP1_TRP_EN	IDE Primary Drive 1. 0 = Disable. 1 = Enable.	R/W
0	IDEP0_TRP_EN	IDE Primary Drive 0. 0 = Disable. 1 = Enable.	R/W

8.8.3.16 BUS_ADDR_TRACK— Bus Address Tracker

This register could be used by the SMI# handler to assist in determining what was the last cycle from the processor. BUS_ADDR_TRACK may not contain “expected” last I/O cycle data when Asynchronous SMIs and Synchronous SMIs are occurring simultaneously. This register only reports “expected” last I/O cycle data when Asynchronous SMIs are disabled.

Note: Usage: Legacy only.

Table 299. BUS_ADDR_TRACK— Bus Address Tracker

Device:	31	***Function:***	0
I/O Address:	PMBASE +4Ch	***Attribute:***	Read-Only
		Size:	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0		Corresponds to the low 16 bits of the last I/O cycle, as would be defined by the PCI AD[15:0] signals on the PCI bus (even though it may not be a real PCI cycle). The value is latched based on SMI# active. This functionality is useful for figuring out which I/O was last being accessed.	

8.8.3.17 BUS_CYC_TRACK— Bus Cycle Tracker

This register could be used by the SMM handler to assist in determining what was the last cycle from the processor. BUS_CYC_TRACK may not contain “expected” last I/O cycle data when Asynchronous SMIs and Synchronous SMIs are occurring simultaneously. This register only reports “expected” last I/O cycle data when Asynchronous SMIs are disabled.

Note: Usage: Legacy only.

Table 300. BUS_CYC_TRACK— Bus Cycle Tracker

Device:	31	**Function:**	0
I/O Address:	PMBASE +4Eh	**Attribute:**	Read-Only
		Size:	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:4		Corresponds to the byte enables, as would be defined by the PCI C/BE# signals on the PCI bus (even though it may not be a real PCI cycle). The value is latched based on SMI# going active.	
3:0		Corresponds to the cycle type, as would be defined by the PCI C/BE# signals on the PCI bus (even though it may not be a real PCI cycle). The value is latched based on SMI# going active.	

8.9 System Management TCO Registers (D31:F0)

The TCO logic is accessed through registers mapped to the PCI configuration space (Device 31:Function 0) and the system I/O space. For TCO PCI Configuration registers, see LPC Device 31:Function 0 PCI Configuration registers.

8.9.1 TCO Register I/O Map

The TCO I/O registers reside in a 32-byte range pointed to by a TCOBASE value, which is, ACPIBASE + 60h in the PCI config space. The following table shows the mapping of the registers within that 32-byte range. Each register is described in the sections below.

Table 301. TCO I/O Register Map

Offset	Type	Register Name: Function
00h	R/W	TCO_RLD: TCO Timer Reload and Current Value
01h	R/W	TCO_TMR: TCO Timer Initial Value
02h	R/W	TCO_DAT_IN: TCO Data In
03h	R/W	TCO_DAT_OUT: TCO Data Out
04h - 05h	R/W	TCO1_STS: TCO Status
06h - 07h	R/W	TCO2_STS: TCO Status
08h - 09h	R/W	TCO1_CNT: TCO Control
0Ah - 0Bh	R/W	TCO2_CNT: TCO Control
0Ch - 0Dh	R/W	TCO_MESSAGE1, TCO_MESSAGE2: Used by BIOS to indicate POST/Boot progress
0Eh	R/W	TCO_WDSTATUS: Watchdog Status Register

Table 301. TCO I/O Register Map

Offset	Type	Register Name: Function
0Fh	RO	Reserved
10h	R/W	SW_IRQ_GEN: Software IRQ Generation Register
11h - 1Fh	RO	Reserved

8.9.2 TCO1_RLD—TCO Timer Reload and Current Value

Table 302. TCO1_RLD—TCO Timer Reload and Current Value

Device:	31	***Function:***	0
I/O Address:	TCOBASE +00h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Reading this register will return the current count of the TCO timer. Writing any value to this register will reload the timer to prevent the timeout. Bits 7:6 will always be 0.	

8.9.3 TCO1_TMR—TCO Timer Initial Value

Table 303. TCO1_TMR—TCO Timer Initial Value

Device:	31	***Function:***	0
I/O Address:	TCOBASE +01h	***Attribute:***	Read/Write
Default Value:	04h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:6	Reserved	Reserved.	
5:0		Value that is loaded into the timer each time the TCO_RLD register is written. Values of 0h - 3h will be ignored and should not be attempted. The timer is clocked at approximately 0.6 seconds, and this allows timeouts ranging from 2.4 seconds to 38 seconds.	R/W

8.9.4 TCO1_DAT_IN—TCO Data In Register

Table 304. TCO1_DAT_IN—TCO Data In Register

Device:	31	***Function:***	0
I/O Address:	TCOBASE +02h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Data Register for passing commands from the OS to the SMI handler. Writes to this register will cause an SMI and set the OS_TCO_SMI bit in the TCO_STS register.	

8.9.5 TCO1_DAT_OUT—TCO Data Out Register

Table 305. TCO1_DAT_OUT—TCO Data Out Register

Device:	31	***Function:***	0
I/O Address:	TCOBASE +03h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0		Data Register for passing commands from the SMI handler to the OS. Writes to this register will set the TCO_INT_STS bit in the TCO_STS register. It will also cause an interrupt, as selected by the TCO_INT_SEL bits.	

8.9.6 TCO1_STS—TCO1 Status Register

Table 306. TCO1_STS—TCO1 Status Register (Sheet 1 of 2)

Device:	31	***Function:***	0
I/O Address:	TCOBASE +04h	***Attribute:***	Read-Only, Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core (Except bit 7, in RTC)

Bits	Name	Description	Access
15:13	Reserved	Reserved.	
12	HUBSERR_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH received an SERR# message through the Hub Interface. The software must read the memory controller hub (or its equivalent) to determine the reason for the SERR#. **NOTE:** When this bit is set AND the SERR_EN bit in CMD register (D30:F0, Offset 04h, bit 8) is also set, the Intel® 6300ESB ICH will set the SSE bit in SECSTS register (D30:F0, offset 1Eh, bit 14) AND will also generate a NMI (or SMI# when NMI routed to SMI#).	R/WC
11	HUBNMI_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH received an NMI message through the Hub Interface. The software must read the memory controller hub (or its equivalent) to determine the reason for the NMI.	R/WC
10	HUBSMI_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH received an SMI message through the Hub Interface. The software must read the memory controller hub (or its equivalent) to determine the reason for the SMI#.	R/WC
9	HUBSCI_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH received an SCI message through the Hub Interface. The software must read the memory controller hub (or its equivalent) to determine the reason for the SCI.	R/WC
8	BIOSWR_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = The Intel® 6300ESB ICH sets this bit and generates and SMI# to indicate an illegal attempt to write to the BIOS. This occurs when either: a) The BIOSWP bit is changed from 0 to 1 and the BLD bit is also set, or b) any write is attempted to the BIOS and the BIOSWP bit is also set.	R/WC

Table 306. TCO1_STS—TCO1 Status Register (Sheet 2 of 2)

Device:	31	**Function:**	0
I/O Address:	TCOBASE +04h	**Attribute:**	Read-Only, Read/Write Clear
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core (Except bit 7, in RTC)

Bits	Name	Description	Access
7	NEWCENTURY_STS	This bit is in the RTC well. 0 = Cleared by writing a 1 to the bit position or by RTCRST# going active. 1 = This bit is set when the Year byte (RTC I/O space, index offset 09h) rolls over from 99 to 00. Setting this bit will cause an SMI# (but not a wake event). Note that the NEWCENTURY_STS bit is not valid when the RTC battery is first installed (or when RTC power has not been maintained). Software may determine when RTC power has not been maintained by checking the RTC_PWR_STS bit, or by other means (such as a checksum on RTC RAM). When RTC power is determined to have not been maintained, BIOS should set the time to a legal value and then clear the NEWCENTURY_STS bit. The NEWCENTURY_STS bit may take up to 3 RTC clocks for the bit to be cleared after a “1” is written to the bit to clear it. After writing a “1” to this bit, software should not exit the SMI handler until verifying that the bit has actually been cleared. This will ensure that the SMI is not re-entered.	R/WC
6:4	Reserved	Reserved.	
3	TIMEOUT	0 = Software clears this bit by writing a 1 to the bit position. 1 = Set by the Intel® 6300ESB ICH to indicate that the SMI was caused by the TCO timer reaching 0.	R/WC
2	TCO_INT_STS	0 = Software clears this bit by writing a 1 to the bit position. 1 = SMI handler caused the interrupt by writing to the TCO_DAT_OUT register.	R/WC
1	SW_TCO_SMI	0 = Software clears this bit by writing a 1 to the bit position. 1 = Software caused an SMI# by writing to the TCO_DAT_IN register.	R/WC
0	NMI2SMI_STS	0 = Cleared by clearing the associated NMI status bit. 1 = Set by the Intel® 6300ESB ICH when an SMI# occurs because an event occurred that would otherwise have caused an NMI (because NMI2SMI_EN is set).	RO

8.9.7 TCO2_STS—TCO2 Status Register

Table 307. TCO2_STS—TCO2 Status Register

Device:	31	***Function:***	0
I/O Address:	TCOBASE +06h	***Attribute:***	Read-Only, Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume (Except Bit 0, in RTC)

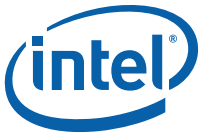
Bits	Name	Description	Access
15:5	Reserved	Reserved.	
4	SMLINK_SLV_SMI_STS	SMLink Slave SMI Status (Allow the software to go directly into pre-determined sleep state. This avoids race conditions. 0 = The bit is reset by RSMRST#, but not due to the PCI Reset associated with exit from S3-S5 states. 1 = The Intel® 6300ESB ICH sets this bit to 1 when it receives the SMI message on the SMLink's Slave Interface. Software clears the bit by writing a 1 to this bit position.	R/W
3	BAD_BIOS	This bit is set by the Intel® 6300ESB ICH when it detects FFh on the first BIOS read (i.e., the BIOS is bad). Intel® 6300ESB ICH clears this bit to 0 if the first BIOS read is not FFh. This is detected when the initial read returns FFh from the FWH. This bit is not intended to be read by the BIOS or software. It is only used for sending the TCO/Heartbeat messages to an External LAN Controller. Reads to this bit always return 0 and writes have no effect.	
2	BOOT_STS:	0 = Cleared by the Intel® 6300ESB ICH based on RSMRST# or by software writing a 1 to this bit. Note that software should first clear the SECOND_TO_STS bit before writing a 1 to clear the BOOT_STS bit. 1 = Set to 1 when the SECOND_TO_STS bit goes from 0 to 1 and the processor has not fetched the first instruction.	
1	SECOND_TO_STS	0 = This bit is cleared by writing a 1 to the bit position or by a RSMRST#. 1 = The Intel® 6300ESB ICH sets this bit to a 1 to indicate that the TCO timer timed out a second time (probably due to system lock). When this bit is set and the NO_REBOOT configuration bit is 0, then the Intel® 6300ESB ICH will reboot the system after the second timeout. The reboot is done by asserting PXPCIRST#.	R/WC
0	INTRD_DET: Intruder Detect	0 = This bit is only cleared by writing a 1 to the bit position, or by RTCRST# assertion. 1 = Set by the Intel® 6300ESB ICH to indicate that an intrusion was detected. This bit is set even when the system is in G3 state.	R/WC

8.9.8 TCO1_CNT—TCO1 Control Register

Table 308. TCO1_CNT—TCO1 Control Register

Device:	31	***Function:***	0
I/O Address:	TCOBASE +08h	***Attribute:***	Read-Only, Read/Write Clear
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:13	Reserved	Reserved.	
12	TCO_LOCK	0 = This bit defaults to 0. A core-well reset is required to change this bit from 1 to 0. 1 = Prevents writes from changing the TCO_EN bit (in offset 30h of Power Management I/O space). Once this bit is set to 1, it can not be cleared by software writing a 0 to this bit location.	R/W
11	TCO_TMR_HLT: TCO Timer Halt	0 = The TCO Timer is enabled to count. 1 = The TCO Timer will halt. It will not count, and thus cannot reach a value that will cause an SMI# or set the SECOND_TO_STS bit. When set, this bit will prevent rebooting and prevent Alert On LAN event messages from being transmitted on the SMLINK (but not Alert On LAN* heartbeat messages).	R/W
10	SEND_NOW	0 = The Intel® 6300ESB ICH will clear this bit when it has completed sending the message. Software must NOT set this bit to 1 again until the Intel® 6300ESB ICH has set it back to 0. 1 = Writing a 1 to this bit will cause the Intel® 6300ESB ICH to send an Event message with the Software Event bit set.	R/W
9	NMI2SMI_EN	0 = Normal NMI functionality. 1 = Forces all NMIs to instead cause SMIs. The functionality of this bit is dependent upon the settings of the NMI_EN bit and the GBL_SMI_EN bit as detailed in the following table: **NMI_EN GBL_SMI_EN Description** 0 0 No SMI# at all because GBL_SMI_EN = 0 0 1 SMI# will be caused due to NMI events 1 0 No SMI# at all because GBL_SMI_EN = 0 1 1 No SMI# due to NMI because NMI_EN = 1	R/W
8	NMI_NOW	0 = This bit is cleared by writing a 1 to the bit position. The NMI handler is expected to clear this bit. Another NMI will not be generated until the bit is cleared. 1 = Writing a 1 to this bit causes an NMI. This allows the BIOS or SMI handler to force an entry to the NMI handler.	R/WC
7:0	Reserved	Reserved.	



8.9.9 TCO2_CNT—TCO2 Control Register

Table 309. TCO2_CNT—TCO2 Control Register

Device:	31	***Function:***	0
I/O Address:	TCOBASE +0Ah	***Attribute:***	Read/Write
Default Value:	0008h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
15:4	Reserved	Reserved.	
3	GPIO11_ALERT_DISABLE	Disable GPIO11/SMBALERT# as an alert source for the heartbeats and the SMBus slave. At reset (via RSMRST# asserted) this bit is set and GPIO[11] alerts are disabled	R/W
2:1	INTRD_SE	Selects the action to take when the INTRUDER# signal goes active. 00 = No interrupt or SMI# 01 = Interrupt (as selected by TCO_INT_SEL). 10 = SMI 11 = Reserved	R/W
0	Reserved	Reserved.	

8.9.10 TCO_MESSAGE1 and TCO_MESSAGE2 Registers

Table 310. TCO_MESSAGE1 and TCO_MESSAGE2 Registers

Device:	31	***Function:***	0
I/O Address:	TCOBASE+0Ch(Message 1) TCOBASE+0Dh(Message 2)	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	TCO_MESSAGE[*n*]	The value written to this register can be passed via SMBus to an External LAN controller. It may be used by the BIOS or system management software to indicate more details on the boot progress. This register will be reset to the default of 00h based on RSMRST# (but not PCI reset).	R/W

8.9.11 Offset TCOBASE + OEh: TCO_WDSTATUS—TCO2 Control Register

Table 311. Offset TCOBASE + OEh: TCO_WDSTATUS—TCO2 Control Register

Device:	31	***Function:***	0
Offset:	TCOBASE + 0Eh	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Resume

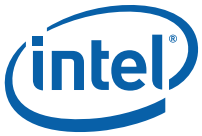
Bits	Name	Description	Access
7:0	WDSTATUS: Watchdog Status	The value written to this register can be passed via SMBus to an External LAN controller. It may be used by the BIOS or system management software to indicate more details on the boot progress. This register will be reset to the default of 00h based on RSMRST# (but not PCI reset).	R/W

8.9.12 Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register

Table 312. Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register

Device:	31	***Function:***	0
Offset:	TCOBASE + 10h	***Attribute:***	Read/Write
Default Value:	11h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1	IRQ12_CAUSE	The state of this bit is logically ANDed with the IRQ12 signal as received by the Intel® 6300ESB ICH's SERIRQ logic. This bit must be a '1' (default) when the Intel® 6300ESB ICH is expected to receive IRQ12 assertions from a SERIRQ device.	R/W
0	IRQ1_CAUSE	The state of this bit is logically ANDed with the IRQ1 signal as received by the Intel® 6300ESB ICH's SERIRQ logic. This bit must be a '1' (default) when the Intel® 6300ESB ICH is expected to receive IRQ1 assertions from a SERIRQ device.	R/W



8.10 General Purpose I/O Registers (D31:F0)

The control for the general purpose I/O signals is handled through a separate 64-byte I/O space. The base offset for this space is selected by the GPIO_BAR register.

8.10.1 GPIO Register I/O Address Map

Table 313. Registers to Control GPIO

Offset	Mnemonic	Register Name	Default	Access
General Registers				
00-03h	GPIO_USE_SEL	GPIO Use Select	1B003100h	R/W
04-07h	GP_IO_SEL	GPIO Input/Output Select	0000 FFFFh	R/W
0C-0Fh	GP_LVL	GPIO Level for Input or Output	1F1F 0000h	R/W
Output Control Registers				
18-1Bh	GPO_BLINK	GPIO Blink Enable	00000000h	R/W
Input Control Registers				
2C-2Fh	GPI_INV	GPIO Signal Invert	00000000h	R/W
30-33h	GPIO_USE_SEL 2	GPIO Use Select	03000000h	R/W
34-37h	GP_IO_SEL2	GPIO Input/Output Select 2	00000000h	R/W
38-3Bh	GP_LVL2	GPIO Level for Input or Output 2	00000FFFh	R/W

8.10.2 Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register

Table 314. Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register

Device:	31	***Function:***	0
Offset:	GPIOBASE + 00h	***Attribute:***	Read/Write
Default Value:	1B003100h	***Size:***	32-bit
Lockable:	Yes	***Power Well:***	Core for 0:7 and 16:21, 23 Resume for 8:15 and 24:31

Bits	Name	Description	Access
23, 21:16, 11, 7:0	GPIO_USE_SEL[0:7, 11, 16:21, 23]	Enables GPIO[n] (where n is the bit number) to be used as a GPIO, rather than for the native function. 1 = Signal used as GPIO (or unmuxed). 0 = Signal used as native function. Notes: 1. The following bits are not implemented because there is no corresponding GPIO: 9:10, 14:15, 22, 26 and 29:31. 2. The following bits are always 1 because they are unmuxed: 8, 12:13, 24, 25, 27:28. 3. When GPIO[n] does not exist, the bit in this register will always read as 0 and writes will have no effect. 4. After a full reset (RSMRST#) all multiplexed signals in the resume and core wells are configured as their native function rather than as a GPIO. After just a PXPCIRST#, the GPIO in the core well are configured as their native function.	

8.10.3 Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register

Table 315. Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +04h	***Attribute:***	Read-Only
Default Value:	0000FFFFh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
31:29, 26	Reserved	Reserved.	
28:27 25:24	GPIO[n]_SEL	0 = Output. The corresponding GPIO signal is an output. 1 = Input. The corresponding GPIO signal is an input.	R/W
23		Always 0. The GPIOs are fixed as outputs.	

Table 315. Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +04h	***Attribute:***	Read-Only
Default Value:	0000FFFFh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
22	Reserved	Reserved.	
21:16		Always 0. The GPIOs are fixed as outputs.	
15:0		Always 1. These GPIOs are fixed as inputs.	

8.10.4 Offset GPIOBASE + 0Ch: GP_LVL—GPIO Level for Input or Output Register

Table 316. Offset GPIOBASE + 0Ch: GP_LVL—GPIO Level for Input or Output Register

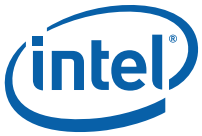
Device:	31	***Function:***	0
Offset:	GPIOBASE +0Ch	***Attribute:***	Read/Write
Default Value:	1B3F0000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	See bit descriptions

Bits	Name	Description	Access
31:29, 26	Reserved	Reserved.	
28:27, 25:24	GP_LVL[n]	When GPIO[n] is programmed to be an output (through the corresponding bit in the GP_IO_SEL register), the bit may be updated by software to drive a high or low value on the output pin. When GPIO[n] is programmed as an input, software may read the bit to determine the level on the corresponding input pin. These bits correspond to GPIO that are in the Resume well, and will be reset to their default values by RSMRST# and also by a write to the CF9h register. 0 = Low 1 = High	R/W

Table 316. Offset GPIOBASE + 0Ch: GP_LVL—GPIO Level for Input or Output Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +0Ch	***Attribute:***	Read/Write
Default Value:	1B3F0000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	See bit descriptions

Bits	Name	Description	Access
23	GP_LVL[n]	These bits may be updated by software to drive a high or low value on the output pin. These bits correspond to GPIO that are in the core well, and will be reset to their default values by PXPCIRST#. 0 = Low 1 = High	R/W
21:16	GP_LVL[n]	These bits may be updated by software to drive a high or low value on the output pin. These bits correspond to GPIO that are in the core well, and will be reset to their default values by PXPCIRST#. 0 = Low 1 = High	R/W
15:0	Reserved	Reserved. These bits are not needed as the level of general inputs can be read through the GPE0_STS and ALT_GP_SMI_STS registers. See Section 8.8.3.7 and Section 8.8.3.12	



8.10.5 Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register

Table 317. Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +18h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	See bit description

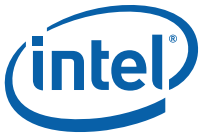
Bits	Name	Description	Access
31:29, 26, 24:20, 17:0	Reserved	Reserved.	
28:27, 25	GP_BLINK[n]	The setting of these bits will have no effect when the corresponding GPIO is programmed as an input. These bits correspond to GPIO that are in the Resume well, and will be reset to their default values by RSMRST# or a write to the CF9h register. 0 = The corresponding GPIO will function normally. 1 = When the corresponding GPIO is programmed as an output, the output signal will blink at a rate of approximately once per second. The high and low times have approximately 0.5 seconds each. The GP_LVL bit is not altered when this bit is set.	R/W
19:18	GP_BLINK[n]	The setting of these bits will have no effect when the corresponding GPIO is programmed as an input. These bits correspond to GPIO that are in the Core well, and will be reset to their default values by PXPCIRST#. 0 = The corresponding GPIO will function normally. 1 = When the corresponding GPIO is programmed as an output, the output signal will blink at a rate of approximately once per second. The high and low times are approximately 0.5 seconds each. The GP_LVL bit is not altered when this bit is set.	R/W

8.10.6 Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register

Table 318. Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +2Ch	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	See bit description

Bits	Name	Description	Access
31:14, 10:9	Reserved	Reserved.	
13:11, 8	GP_INV[n]	These bits are used to allow both active-low and active-high inputs to cause SMI# or SCI. Note that in the S0 or S1 state, the input signal must be active for at least 2 PCI clocks to ensure detection by the Intel® 6300ESB ICH. In the S3, S4 or S5 states the input signal must be active for at least 2 RTC clocks to ensure detection. The setting of these bits will have no effect when the corresponding GPIO is programmed as an output. These bits correspond to GPIO that are in the Resume well, and will be reset to their default values by RSMRST# or a write to the CF9h register. 0 = The corresponding GPIn_STS bit will be set when the Intel® 6300ESB ICH detects the state of the input pin to be high. 1 = The corresponding GPIn_STS bit will be set when the Intel® 6300ESB ICH detects the state of the input pin to be low. **NOTE:** The GPIn_STS bits are in the GPE0_STS Register. See section 9.8.3.7	R/W
7:0	GP_INV[n]	These bits are used to allow both active-low and active-high inputs to cause SMI# or SCI. Note that in the S0 or S1 state, the input signal must be active for at least 2 PCI clocks to ensure detection by the Intel® 6300ESB ICH. The setting of these bits will have no effect when the corresponding GPIO is programmed as an output. These bits correspond to GPIO that are in the Core well, and will be reset to their default values by PXPCIRST#. 0 = The corresponding GPIn_STS bit will be set when the Intel® 6300ESB ICH detects the state of the input pin to be high. 1 = The corresponding GPIn_STS bit will be set when the Intel® 6300ESB ICH detects the state of the input pin to be low.	R/W



8.10.7 Offset GPIOBASE + 30h:GPIO_USE_SEL2—GPIO Use Select 2 Register

Table 319. Offset GPIOBASE + 30h:GPIO_USE_SEL2—GPIO Use Select 2 Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +30h	***Attribute:***	Read/Write
Default Value:	03000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core for 55:32 = bits 23:0 Resume for 63:56 = bits 31:24

Bits	Name	Description	Access
31:26		Always 0. No corresponding GPIO.	
25:24		Always 1. These pins are unmuxed.	
23:12		Always 0. No corresponding GPIO.	
11:8		**GPIO_USE_SEL[43:40]:** Enables GPIO[n] (where n is the bit number) to be used as a GPIO, rather than for the native function. Since these pins may be used as outputs for controlling power planes, switching the pin from functional to GPO mode must be glitch-free. 1 = Signal used as GPIO (or unmuxed). 0 = Signal used as native function. After a full reset(RSMRST#) all multiplexed signals in the resume and core wells are configured as their native function rather than as a GPIO. After just a PXPCIRST#, the GPIO in the core well are configured as GPIO.	
7:0		**GPIO_USE_SEL[39:32]:** Enables GPIO[n] (where n is the bit number) to be used as a GPIO, rather than for the native function. Since these pins may be used as outputs for controlling power planes, switching the pin from functional to GPO mode must be glitch-free. 1 = Signal used as GPIO (or unmuxed). 0 = Signal used as native function. After a full reset(RSMRST#) all multiplexed signals in the resume and core wells are configured as their native function rather than as a GPIO. After just a PXPCIRST#, the GPIO in the core well are configured as GPIO.	
Implementation Note: Bits 26:31 may be in CORE Well.			

8.10.8 Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register

Table 320. Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +34h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core for 23:0; Resume for 31:24

Bits	Name	Description	Access
31:26		Always 0. No corresponding GPIO.	
25:24		Always 0. Output only.	
23:12		Always 0. No corresponding GPIO.	
11:0	GP_IO_SEL2[43:32]	When set to a 1, the corresponding GPIO signal (when enabled in the GPIO_USE_SEL2 register) is programmed as an input. When set to 0, the GPIO signal is programmed as an output.	
Implementation Note: Bits 26:31 may be in CORE Well.			

8.10.9 Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register

Table 321. Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +38h	***Attribute:***	Read/Write
Default Value:	00000FFFh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core for 23:0, RTC for 31:24

Bits	Name	Description	Access
31:26	Reserved	Reserved. Read-only 0.	RO
25:24	GP_LVL[57:56]	The GP_LVL[n] bit may be updated by software to drive a high or low value on the output pin. 1 = high, 0 = low. **NOTE:** These output are open drain. Setting this bit to one does not drive a '1' but allows for an external pullup to cause a high value on the pin. Since these bits correspond to GPIO that are in the RTC well, these bits will be reset by RTCRST#.	

Table 321. Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register

Device:	31	***Function:***	0
Offset:	GPIOBASE +38h	***Attribute:***	Read/Write
Default Value:	00000FFFh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core for 23:0, RTC for 31:24

Bits	Name	Description	Access
23:12	Reserved	Reserved. Read-only 0.	RO
11:0	**GP_LVL2[43:32]**	When GPIO[n] is programmed to be an output (through the corresponding bit in the GP_IO_SEL2 register), the corresponding GP_LVL2[n] bit may be updated by software to drive a high or low value on the output pin. 1 = high, 0 = low. When GPIO[n] is programmed as an input, then the corresponding GP_LVL2 bit reflects the state of the input signal (1 = high, 0 = low). Writes will have no effect. Since these bits correspond to GPIO that are in the core well, these bits will be reset by PXPCIRST#.	
Implementation Note: Bits 26:31 may be in CORE Well.			

This page intentionally left blank.

IDE Controller Registers (D31:F1) 9

9.1 PCI Configuration Registers (IDE—D31:F1)

Note: Registers that are not shown should be treated as Reserved (see Section 6.2, "PCI Configuration Map" for details). All of the IDE registers are in the core well. None of the registers may be locked.

Table 322. PCI Configuration Map (IDE-D31:F1)

Offset	Mnemonic	Register Name/Function	Default	Type
00-01h	VID	Vendor ID	8086h	RO
02-03h	DID	Device ID	25A2h	RO
04-05h	CMD	Command	00h	R/W
06-07h	STS	Device Status	0280h	R/W
08h	RID	Revision ID	00h See Note 1	RO
09h	PI	Programming Interface	8Ah	RO
0Ah	SCC	Sub Class Code	01h	RO
0Bh	BCC	Base Class Code	01h	RO
0Dh	MLT	Master Latency Timer	00h	RO
10-13h	PCMD_BAR	Primary Command Block Base Address	00000001h	R/W
14-17h	PCNL_BAR	Primary Control Block Base Address	00000001h	R/W
18-1Bh	SCMD_BAR	Secondary Command Block Base Address	00000001h	R/W
1C-1Fh	SCNL_BAR	Secondary Control Block Base Address	00000001h	R/W
20-23h	BM_BASE	Bus Master Base Address	00000001h	R/W
24-27h	CPBA	IDE Command Posting Base Address	00000000h	R/W
2C-2Dh	IDE_SVID	Subsystem Vendor ID	00h	R/Write-Once
2E-2Fh	IDE_SID	Subsystem ID	00h	R/Write-Once
3C	INTR_LN	Interrupt Line	00h	R/W
3D	INTR_PN	Interrupt Pin	01h	R/W
40-41h	IDE_TIM	Primary IDE Timing	0000h	R/W
42-43h	ID_TIMS	Secondary IDE Timing	0000h	R/W
44h	SLV_IDETIM	Slave IDE Timing	00h	R/W
48h	SDMA_CNT	Synchronous DMA Control	00h	R/W
4A-4Bh	SDMATIM	Synchronous DMA Timing	0000h	R/W
54h	IDE_CONFIG	IDE I/O Configuration	00h	R/W

NOTES:
1. See the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID Register.
2. The Intel® 6300ESB ICH IDE controller is not arbitrated as a PCI device, therefore it does not need a master latency timer.

9.1.1 Offset 00 - 01h: VID—Vendor ID Register (LPC I/F—D31:F1)

Table 323. Offset 00 - 01h: VID—Vendor ID Register (LPC I/F—D31:F1)

Device:	31	***Function:***	1
Offset:	00 - 01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

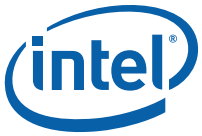
Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel. Intel VID = 8086h.	

9.1.2 Offset 02 - 03h: DID—Device ID Register (LPC I/F—D31:F1)

Table 324. Offset 02 - 03h: DID—Device ID Register (LPC I/F—D31:F1)

Device:	31	***Function:***	1
Offset:	02 - 03h	***Attribute:***	Read-Only
Default Value:	25A2h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
	Device ID	This is a 16-bit value assigned to the Intel® 6300esb ICH	RO



9.1.3 Offset 04h - 05h: CMD—Command Register (IDE—D31:F1)

Table 325. Offset 04h - 05h: CMD—Command Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	04h-05h	***Attribute:***	Read-Only, Read/Write
Default Value:	00h	***Size:***	16-bit

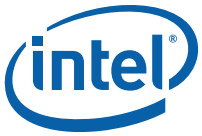
Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10	Interrupt Disable (ID)	Enables the P-ATA host controller to assert the INTA# (in native mode) or IRQ14/15 (in legacy mode). When set, the interrupt will be deasserted. When cleared, the interrupt may be asserted.	
9	Fast Back to Back Enable (FBE)	Reserved as ‘0’.	RO
8	SERR# Enable	Reserved as ‘0’.	RO
7	Wait Cycle Control	Reserved as ‘0’.	RO
6	Parity Error Response	Reserved as ‘0’.	RO
5	VGA Palette Snoop	Reserved as ‘0’.	RO
4	Postable Memory Write Enable (PMWE)	Reserved as ‘0’.	RO
3	Special Cycle Enable (SCE)	Reserved as ‘0’.	RO
2	Bus Master Enable (BME)	Controls the Intel® 6300ESB ICH’s ability to act as a PCI master for IDE Bus Master transfers.	R/W
1	Memory Space Enable (MSE)	0 = Disables access. 1 = Enables access to the IDE Expansion memory range. The EXBAR register (Offset 24h) must be programmed before this bit is set. **NOTE:** BIOS should set this bit to a 1.	R/W
0	IOSE - I/O Space Enable (IOSE)	This bit controls access to the I/O space registers. 0 = Disables access to the Legacy or Native IDE ports (both Primary and Secondary) as well as the Bus Master IO registers. 1 = Enable. The Base Address register for the Bus Master registers should be programmed before this bit is set. **NOTES:** 1. Separate bits are provided (IDE Decode Enable, in the IDE Timing register) to independently disable the Primary or Secondary I/O spaces. 2. When this bit is 0 and the IDE controller is in Native Mode, the Interrupt Pin Register will be masked (the interrupt will not be asserted) See Section 10.1.16, “Offset 60h: USB_RELNUM—USB Release Number Register (USB—D29:F0/F1)” for more information regarding the Interrupt Pin Register. When an interrupt occurs while the masking is in place and the interrupt is still active when the masking ends, the interrupt will be allowed to be asserted.	R/W

9.1.4 Offset 06 - 07h: STS—Device Status Register (IDE—D31:F1)

Table 326. Offset 06 - 07h: STS—Device Status Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	06-07h	***Attribute:***	Read/Write Clear, Read-Only
Default Value:	0280h	***Size:***	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	Reserved as '0'.	RO
14	Signaled System Error (SSE)	Reserved as '0'.	RO
13	Received Master-Abort Status (RMA)	0 Cleared by writing a 1 to it. Bus Master IDE interface function, as a master, generated a master-abort.	R/WC
12	Reserved	Reserved as '0'.	RO
11	Signaled Target-Abort Status (STA)	0 = Cleared by writing a 1 to it. 1 = The Intel® 6300ESB ICH IDE interface function is targeted with a transaction that the Intel® 6300ESB ICH terminates with a target abort.	R/WC
10:9	DEVSEL# Timing Status (DEVT)	01 = Hardwired; however, the Intel® 6300ESB ICH does not have a real DEVSEL# signal associated with the IDE unit, so these bits have no effect.	RO
8	Data Parity Error Detected	Reserved as '0'.	RO
7	Fast Back-to-Back Capable	Reserved as '1'.	RO
6	User Definable Features (UDF)	Reserved as '0'.	RO
5	66 MHz Capable	Reserved as '0'.	RO
4	Capabilities List (CAP)	Reserved as '0'	RO
3	Interrupt Status (IS)	Reflects the state of interrupt at the input of the enable.disable circuit. This bit is a 1 when the interrupt is asserted. 1 = Interrupt is asserted. 0 = Interrupt has been cleared (independent of the state of the Interrupt Disable bit in the command register.	RO
2:0	Reserved	Reserved as '0'.	RO



9.1.5 Offset 08h: RID—Revision ID Register (IDE—D31:F1)

Table 327. Offset 08h: RID—Revision ID Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB ICH *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

9.1.6 Offset 09h: PI—Programming Interface (IDE—D31:F1)

Table 328. Offset 09h: PI—Programming Interface (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	09h	***Attribute:***	Read/Write
Default Value:	8Ah	***Size:***	8-bit

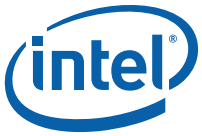
Bits	Name	Description	Access
7		This read-only bit is a 1 to indicate that the Intel® 6300ESB ICH supports bus master operation	
6:4	Reserved	Reserved. Will always return 0.	
3	SOP_MODE_CAP	This read-only bit is a 1 to indicate that the secondary controller supports both legacy and native modes.	
2	SOP_MODE_SEL	This read-write bits determines the mode that the secondary IDE channel is operating in. 0 = Legacy-PCI mode (default) 1 = Native-PCI mode	
1	POP_MODE_CAP	This read-only bit is a 1 to indicate that the primary controller supports both legacy and native modes.	
0	POP_MODE_SEL	This read-write bits determines the mode that the primary IDE channel is operating in. 0 = Legacy-PCI mode (default) 1 = Native-PCI mode **NOTE:** In the Intel® 6300ESB ICH, this bit was read-only with a value of 0.	

9.1.7 Offset 0Ah: SCC—Sub Class Code (IDE—D31:F1)

Table 329. Offset 0Ah: SCC—Sub Class Code (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub Class Code	01h = IDE device, in the context of a mass storage device.	RO



9.1.8 Offset 0Bh: BCC—Base Class Code (IDE—D31:F1)

Table 330. Offset 0Bh: BCC—Base Class Code (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	01 = Mass storage device	RO

9.1.9 Offset 0Dh: MLT—Master Latency Timer (IDE—D31:F1)

Table 331. Offset 0Dh: MLT—Master Latency Timer (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	0Dh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Bus Master Latency	Hardwired to 00h. The IDE controller is implemented internally, and is not arbitrated as a PCI device, so it does not need a Master Latency Timer. These bits are fixed at 0.	RO

9.1.10 Offset 10h - 13h: PCMD_BAR—Primary Command Block Base Address Register (IDE—D31:F1)

Table 332. Offset 10h - 13h: PCMD_BAR—Primary Command Block Base Address Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	10h-13h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:3	Base Address	Base address of the I/O space (8 consecutive I/O locations).	R/W
2:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to one, indicating a request for IO space. Read-Only.	RO
NOTE: This 8-byte I/O space is used in native mode for the Primary Controller's Command Block.			

9.1.11 Offset 14h - 17h: PCNL_BAR—Primary Control Block Base Address Register (IDE—D31:F1)

Table 333. Offset 14h - 17h: PCNL_BAR—Primary Control Block Base Address Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	14h-17h	***Attribute:***	Read-Only
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:2	Base Address	Base address of the I/O space (4 consecutive I/O locations).	R/W
1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to one, indicating a request for IO space.	RO

NOTE: This 4-byte I/O space is used in native mode for the Primary Controller's Command Block.

9.1.12 Offset 18h - 1Bh: SCMD_BAR—Secondary Command Block Base Address Register (IDE D31:F1)

Table 334. Offset 18h - 1Bh: SCMD_BAR—Secondary Command Block Base Address Register (IDE D31:F1)

Device:	31	***Function:***	1
Offset:	18h-1Bh	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:3	Base Address	Base address of the I/O space (8 consecutive I/O locations).	R/W
2:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to one, indicating a request for IO space.	RO

NOTE: This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

9.1.13 Offset 1Ch - 1Fh: SCNL_BAR—Secondary Control Block Base Address Register (IDE D31:F1)

Table 335. Offset 1Ch - 1Fh: SCNL_BAR—Secondary Control Block Base Address Register (IDE D31:F1)

Device:	31	***Function:***	1
Offset:	1Ch - 1Fh	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:2	Base Address	Base address of the I/O space (4 consecutive I/O locations).	R/W
1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to one, indicating a request for IO space.	RO

NOTE: This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

9.1.14 Offset 20h - 23h: BM_BASE—Bus Master Base Address Register (IDE—D31:F1)

Note: The Bus Master IDE interface function uses Base Address register 5 to request a 16-byte IO space to provide a software interface to the Bus Master functions. Only 12 bytes are actually used (6 bytes for primary, 6 bytes for secondary). Only bits [15:4] are used to decode the address.

Table 336. Offset 20h - 23h: BM_BASE—Bus Master Base Address Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	20h - 23h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:4	Base Address	Base address of the I/O space (16 consecutive I/O locations).	
3:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	Hardwired to'1', indicating a request for IO space.	RO

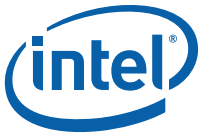
9.1.15 Offset 24h - 27h: CPBA – IDE Command Posting Base Address

Note: The Intel® 6300ESB ICH requests 1 Kbyte of memory space for the Command Posting accesses. This is much more than is needed; by requesting this much space, decoding may be simplified. In addition to the standard PCI Memory Space Enable bit, the Command Posting Enable bits in the IDE I/O Configuration register must be set for the Intel® 6300ESB ICH to properly decode the accesses to this range.

Table 337. Offset 24h - 27h: CPBA – IDE Command Posting Base Address

Device:	31	***Function:***	1
Offset:	24h-27h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:10	Base Address	Base address of the IDE Command Posting memory space (aligned to 1 Kbyte).	R/W
9:4	Hardwired Base Address	These bits are hardwired to ‘0’ to indicate that the size of the range requested is 1 Kbyte.	RO
3	Prefetchable	Hard-wired to ‘0’, indicating that this range is not pre-fetchable.	RO
2:1	Type	Hard-wired to “00”, indicating that this range may be mapped anywhere in 32-bit address space.	RO
0	RTE – Resource Type Indicator	This bit is hard-wired to ‘0’, indicating a request for memory space.	RO



9.1.16 Offset 2Ch - 2Dh: IDE_SVID—Subsystem Vendor ID (IDE—D31:F1)

Table 338. Offset 2Ch - 2Dh: IDE_SVID—Subsystem Vendor ID (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	2Ch-2Dh	***Attribute:***	Read/Write Once
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem Vendor ID (SVID)	The SVID register, in combination with the Subsystem ID (SID) register, enables the operating system (OS) to distinguish subsystems from each other. Software (BIOS) sets the value in this register. After that, the value may be read, but subsequent writes to this register have no effect. The value written to this register will also be readable through the corresponding SVID registers for the USB UHCI #1, USB UHCI #2, and SMBus functions. **NOTE:** Write accesses to the SVID register should only be done as 16-bit accesses. If two 8-bit write accesses are done, then the value in the register will not be correct.	R/Write-Once

9.1.17 Offset 2Eh - 2Fh: IDE_SID—Subsystem ID (IDE—D31:F1)

Table 339. Offset 2Eh - 2Fh: IDE_SID—Subsystem ID (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	2Eh-2Fh	***Attribute:***	Read/Write Once
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem ID (SID)	The SID register, in combination with the SVID register, enables the operating system (OS) to distinguish subsystems from each other. Software (BIOS) sets the value in this register. After that, the value may be read, but subsequent writes to this register have no effect. The value written to this register will also be readable through the corresponding SID registers for the USB UHCI #1, USB UHCI #2, and SMBus functions. **NOTE:** Write accesses to the SID register should only be done as 16-bit accesses. If two 8-bit write accesses are done, then the value in the register will not be correct.	R/Write-Once

9.1.18 Offset 3Ch: INTR_LN—Interrupt Line Register (IDE—D31:F1)

Table 340. Offset 3Ch: INTR_LN—Interrupt Line Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	3Ch	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is used to communicate to software the interrupt line which the interrupt pin is connected to.	**R/W**

9.1.19 Offset 3Dh: INTR_PN—Interrupt Pin Register (IDE—D31:F1)

Table 341. Offset 3Dh: INTR_PN—Interrupt Pin Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2:0	Interrupt Pi	The value of 01h indicates to “software” that the Intel® 6300ESB ICH will drive INTA#. Note that this is only used in native mode. Also note that the routing to the internal interrupt controller does not necessarily relate to the value in this register. The IDE interrupt is in fact routed to PIRQ[C]# (IRQ18 in APIC mode).	RO

9.1.20 IDE_TIM—IDE Timing Register (IDE—D31:F1)

Note: This register controls the timings driven on the IDE cable for PIO and 8237 style DMA transfers. It also controls operation of the buffer for PIO transfers.

Table 342. IDE_TIM—IDE Timing Register (IDE—D31:F1) (Sheet 1 of 3)

Device:	31	**Function:**	1
Offset:	Primary: 40-41h Secondary: 42-43h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15	IDE Decode Enable (IDE)	Individually enable/disable the Primary or Secondary decode. The IDE I/O Space Enable bit in the Command register must be set in order for this bit to have any effect. Additionally, separate configuration bits are provided (in the IDE I/O Configuration register) to individually disable the primary or secondary IDE interface signals, even when the IDE Decode Enable bit is set. 0 = Disable. 1 = Enables the Intel® 6300ESB ICH to decode the associated Command Blocks (1F0-1F7h for primary, 170-177h for secondary) and Control Block (3F6h for primary and 376h for secondary). This bit effects the IDE decode ranges for both legacy and native-Mode decoding. It also effects the corresponding primary or secondary memory decode range for IDE Expansion.	R/W
14	Drive 1 Timing Register Enable (SITRE)	0 = Use bits 13:12, 9:8 for both drive 0 and drive 1. 1 = Use bits 13:12, 9:8 for drive 0, and use the Slave IDE Timing register for drive 1	R/W
13:12	IORDY Sample Point (ISP)	The setting of these bits determine the number of PCI clocks between IDE IOR#/IOW# assertion and the first IORDY sample point. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	
11	Reserved	Reserved.	

Table 342. IDE_TIM—IDE Timing Register (IDE—D31:F1) (Sheet 2 of 3)

Device:	31	**Function:**	1
Offset:	Primary: 40-41h Secondary: 42-43h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
10	Fast Non-Data PIO (FNDPIO)	Software sets this read/write bit to 1 to enable the fast PIO accesses for non Data Register accesses. When this bit is a 1, the timings for PIO cycles to IDE drive registers other than offset 170h or 1F0h will run using PIO data timings. Default for this bit is 0. This mode should only be enabled for UDMA-based IDE protocols. When this bit is set to '1', bit 14 in this register (Drive 1 Timing Register Enable) must be either: '0' to force equivalent timings for both the master and slave devices, or '1' with equivalent timings programmed in the Slave IDE Timing register. Bits 7 and 3 in this register must be '0' in order to utilize this mode. Likewise, bits 4 and 0 of this register must be programmed to '1'; bits 5 and 1 must be programmed to the same value. The timing parameters must, of course, be compatible with each of the devices on the channel. The two drives must be programmed for the same timings because some non-data register accesses must be received by both the master and slave devices simultaneously, and because different settings would result in timings that change in the middle of some of the non-data accesses. Bits 13:12 and bits 9:8 determine the cycle timings. Bits 6 and 2 do not apply to the non-data accesses.Note that the non-data accesses are not posted, but that accesses to the Data register may be posted. Therefore, the non-data accesses may actually be limited by the access turn-around achievable on Hub Interface rather than the RCT timing.	
9:8	Recovery Time (RCT)	The setting of these bits determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clock	R/W
7	Drive 1 DMA Timing Enable (DTE1)	0 = Disable. 1 = Enable the fast timing mode for DMA transfers only for this drive. PIO transfers to the IDE data port will run in compatible timing.	R/W
6	Drive 1 Prefetch/Posting Enable (PPE1)	0 = Disable. 1 = Enable Prefetch and posting to the IDE data port for this drive.	R/W
5	Drive 1 IORDY Sample Point Enable (IE1)	0 = Disable IORDY sampling for this drive. 1 = Enable IORDY sampling for this drive.	R/W

Table 342. IDE_TIM—IDE Timing Register (IDE—D31:F1) (Sheet 3 of 3)

Device:	31	**Function:**	1
Offset:	Primary: 40-41h Secondary: 42-43h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
4	Drive 1 Fast Timing Bank (TIME1)	0 = Accesses to the data port will use compatible timings for this drive. 1 = When this bit ='1' and bit 14 ='0', accesses to the data port will use bits 13:12 for the IORDY sample point, and bits 9:8 for the recovery time. When this bit ='1' and bit 14 ='1', accesses to the data port will use the IORDY sample point and recover time specified in the slave IDE timing register.	R/W
3	Drive 0 DMA Timing Enable (DTE0)	0 = Disable 1 = Enable fast timing mode for DMA transfers only for this drive. PIO transfers to the IDE data port will run in compatible timing.	R/W
2	Drive 0 Prefetch/Posting Enable (PPE0)	0 = Disable prefetch and posting to the IDE data port for this drive. 1 = Enable prefetch and posting to the IDE data port for this drive.	R/W
1	Drive 0 IORDY Sample Point Enable (IE0)	0 = Disable IORDY sampling is disabled for this drive. 1 = Enable IORDY sampling for this drive.	R/W
0	Drive 0 Fast Timing Bank (TIME0)	0 = Accesses to the data port will use compatible timings for this drive. 1 = Accesses to the data port will use bits 13:12 for the IORDY sample point, and bits 9:8 for the recovery time	R/W

9.1.21 Offset 44H: SLV_IDETIM—Slave (Drive 1) IDE Timing Register (IDE—D31:F1)

Table 343. Offset 44H: SLV_IDETIM—Slave (Drive 1) IDE Timing Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	44h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:6	Secondary Drive 1 IORDY Sample Point (SISP1)	Determines the number of PCI clocks between IDE IOR#/IOW# assertion and the first IORDY sample point, when the access is to drive 1 data port and bit 14 of the IDE timing register for secondary is set. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	R/W
5:4	Secondary Drive 1 Recovery Time (SRCT1)	Determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle, when the access is to drive 1 data port and bit 14 of the IDE timing register for secondary is set. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clocks	R/W
3:2	Primary Drive 1 IORDY Sample Point (PISP1)	Determines the number of PCI clocks between IOR#/IOW# assertion and the first IORDY sample point, when the access is to drive 1 data port and bit 14 of the IDE timing register for primary is set. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	R/W
1:0	Primary Drive 1 Recovery Time (PRCT1)	Determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle, when the access is to drive 1 data port and bit 14 of the IDE timing register for primary is set. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clocks	R/W

9.1.22 Offset 48h: SDMA_CNT—Synchronous DMA Control Register (IDE—D31:F1)

Table 344. Offset 48h: SDMA_CNT—Synchronous DMA Control Register (IDE—D31:F1)

Device:	31	***Function:***	1
Offset:	48h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3	Secondary Drive 1 Synchronous DMA Mode Enable (SSDE1)	0 = Disable (default) 1 = Enable Synchronous DMA mode for secondary channel drive 1	R/W
2	Secondary Drive 0 Synchronous DMA Mode Enable (SSDE0)	0 = Disable (default) 1 = Enable Synchronous DMA mode for secondary drive 0.	R/W
1	Primary Drive 1 Synchronous DMA Mode Enable (PSDE1)	0 = Disable (default) 1 = Enable Synchronous DMA mode for primary channel drive 1	R/W
0	Primary Drive 0 Synchronous DMA Mode Enable (PSDE0)	0 = Disable (default) 1 = Enable Synchronous DMA mode for primary channel drive 0	R/W

9.1.23 Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (IDE—D31:F1)

Table 345. Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (IDE—D31:F1) (Sheet 1 of 2)

Device:	31	***Function:***	1
Offset:	4A - 4Bh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13:12	Secondary Drive 1 Cycle Time (SCT1)	For Synchronous DMA mode. The setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **SCB1 = 0 (33 MHz clk**) 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **SCB1 = 1 (66 MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_SCB1 = 1 (133 MHz clk**) 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W
11:10	Reserved	Reserved.	
9:8	Secondary Drive 0 Cycle Time (SCT0)	For Synchronous DMA mode. The setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **SCB1 = 0 (33 MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **SCB1 = 1 (66 MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_SCB1 = 1 (133 MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W
7:6	Reserved	Reserved.	

Table 345. Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (IDE—D31:F1) (Sheet 2 of 2)

Device:	31	**Function:**	1
Offset:	4A - 4Bh	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
5:4	Primary Drive 1 Cycle Time (PCT1)	For Synchronous DMA mode, the setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **PCB1 = 0 (33 MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **PCB1 = 1 (66 MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_PCB1 = 1 (133 MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W
3:2	Reserved	Reserved.	
1:0	Primary Drive 0 Cycle Time (PCT0)	For Synchronous DMA mode, the setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **PCB1 = 0 (33 MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **PCB1 = 1 (66 MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_PCB1 = 1 (133 MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W

9.1.24 IDE_CONFIG—IDE I/O Configuration Register (IDE—D31:F1)

Table 346. IDE_CONFIG—IDE I/O Configuration Register (IDE—D31:F1)

Device:		*Function:*	1
Offset:	54h	*Attribute:*	R/W
Default Value:	00h	*Size:*	32-bit

Bits	Name	Description	Access
31	Enable Prefetch and Posting Registers (EPPR)	Until this bit is set, the PPE1 and PPE0 bits in the IDE Timing Register (bits 6 and 2 of offset 40h for primary, and bits 6 and 2 of offset 42h for secondary) are ignored by the PIO prefetch and posting hardware. Therefore, even if those bits are set, PIO posting and prefetching will not occur until this bit is set.	R/W
30:0	Reserved	Reserved.	

9.2 Bus Master IDE I/O Registers (D31:F1)

The Bus Master IDE function uses 16 bytes of I/O space, allocated through the BMIBA register, located in Device 31:Function 1 Configuration space, offset 20h. All bus master IDE I/O space registers may be accessed as byte, word, or DWORD quantities. Reading reserved bits returns an indeterminate, inconsistent value, and writes to reserved bits have no effect, but should not be attempted. The description of the I/O registers is shown below in Table 347.

Table 347. Bus Master IDE I/O Registers

Offset	Mnemonic	Register	Default	Type
00	BMICP	Command Register Primary	00h	R/W
01		Reserved	00h	RO
02	BMISP	Status Register Primary	00h	R/WC
03		Reserved	00h	RO
04-07	BMIDP	Descriptor Table Pointer Primary	xxxxxxxxh	R/W
08	BMICS	Command Register Secondary	00h	R/W
09		Reserved	00h	RO
0A	BMISS	Status Register Secondary	00h	R/WC
0B		Reserved	00h	RO
0C-0F	BMIDS	Descriptor Table Pointer Secondary	xxxxxxxxh	R/W

9.2.1 BMIC[P,S]—Bus Master IDE Command Register

Table 348. BMIC[P,S]—Bus Master IDE Command Register

Device:	31	***Function:***	1
Offset:	Primary: 00h Secondary: 08h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:4	Reserved	Reserved. Returns ‘0’.	
3	Read / Write Control (RWC)	This bit sets the direction of the bus master transfer: This bit must NOT be changed when the bus master function is active. 0 = Memory reads 1 = Memory writes	R/W
2:1	Reserved	Reserved. Returns ’0’.	
0	Start/Stop Bus Master (START)	0 = All state information is lost when this bit is cleared. Master mode operation cannot be stopped and then resumed. When this bit is reset while bus master operation is still active (i.e., the Bus Master IDE Active bit of the Bus Master IDE Status register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded instead of being written to system memory. 1 = Enables bus master operation of the controller. Bus master operation begins when this bit is detected changing from a zero to a one. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation may be halted by writing a '0' to this bit. **NOTE:** This bit is intended to be cleared by software after the data transfer is completed, as indicated by either the Bus Master IDE Active bit being cleared or the Interrupt bit of the Bus Master IDE Status register for that IDE channel being set, or both. Hardware does not clear this bit automatically.	R/W

9.2.2 BMIS[P,S]—Bus Master IDE Status Register

Table 349. BMIS[P,S]—Bus Master IDE Status Register

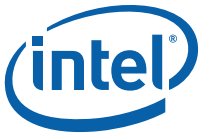
Device:	31	***Function:***	1
Offset:	Primary: 02h Secondary: 0Ah	***Attribute:***	Read/Write Clear
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7	PRD_INT_STS	The Intel® 6300ESB ICH sets this bit when it completes execution of a PRD that has its PRD_INT bit set. This bit is cleared by software writing a 1 to this bit position. When this bit is cleared, the interrupt is cleared. Note that there is a small window where the Intel® 6300ESB ICH completes execution of a PRD, but the interrupt service routine does not clear this bit until the next PRD has completed. In that case, there is a small probability that the interrupt associated with the subsequent PRD is lost. Software must be written to not break in this case.	**R/W**
6	Drive 1 DMA Capable	0 = Not Capable. 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The Intel® 6300ESB ICH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.	**R/W**
5	Drive 0 DMA Capable	0 = Not Capable 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The Intel® 6300ESB ICH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.	**R/W**
4:3	Reserved	Reserved. Returns '0'.	

Table 349. BMIS[P,S]—Bus Master IDE Status Register

Device:	31	**Function:**	1
Offset:	Primary: 02h Secondary: 0Ah	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
2	Interrupt	Software may use this bit to determine if an IDE device has asserted its interrupt line (IRQ 14 for the Primary channel, and IRQ 15 for Secondary). 0 = This bit is cleared by software writing a '1' to the bit position. When this bit is cleared while the interrupt is still active, this bit will remain clear until another assertion edge is detected on the interrupt line. 1 = Set by the rising edge of the IDE interrupt line, regardless of whether or not the interrupt is masked in the 8259 or the internal I/O APIC. When this bit is read as a one, all data transferred from the drive is visible in system memory.	R/WC
1	Error	0 = This bit is cleared by software writing a '1' to the bit position. 1 = This bit is set when the controller encounters a target abort or master abort when transferring data on PCI.	R/WC
0	Bus Master IDE Active (ACT)	0 = This bit is cleared by the Intel® 6300ESB ICH when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared by the Intel® 6300ESB ICH when the Start bit is cleared in the Command register. When this bit is read as a zero, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted. 1 = Set by the Intel® 6300ESB ICH when the Start bit is written to the Command register.	RO



9.2.3 BMID[P,S]—Bus Master IDE Descriptor Table Pointer Register

Note: When this register is read, the current value of the pointer is returned. The Intel® 6300ESB ICH does NOT return the original base value of the pointer once one or more descriptors have been executed. This capability is useful for enabling low-cost disk drives.

Table 350. BMID[P,S]—Bus Master IDE Descriptor Table Pointer Register

Device:	31	***Function:***	1
Offset:	Primary: 04h Secondary: 0Ch	***Attribute:***	Read/Write
Default Value:	All bits undefined	***Size:***	32-bit

Bits	Name	Description	Access
31:2	Address of Descriptor Table (ADDR)	Corresponds to A[31:2]. The Descriptor Table must be DWORD-aligned. The Descriptor Table must not cross a 64 K boundary in memory.	R/W
1:0	Reserved	Reserved.	

November 2007
Order Number: 300641-004US

USB UHCI Controllers Registers 10

10.1 PCI Configuration Registers (D29:F0/F1)

Note: Registers that are not shown should be treated as Reserved (see Section 6.2, "PCI Configuration Map" for details).

Table 351. PCI Configuration Map (USB—D29:F0/F1)

Offset	Mnemonic	Register Name/ Function	Function 0 Default	Function 1 Default	Type
00-01h	VID	Vendor ID	8086h	8086h	RO
02-03h	DID	Device ID	25A9h	25AAh	RO
04-05h	CMD	Command Register	0000h	0000h	R/W
06-07h	STA	Device Status	0280h	0280h	R/W
08h	RID	Revision ID	See NOTE:	See NOTE:	RO
09h	PI	Programming Interface	00h	00h	RO
0Ah	SCC	Sub Class Code	03h	03h	RO
0Bh	BCC	Base Class Code	0Ch	0Ch	RO
ODh	MLT	Master Latency Timer	00h	00h	R/W
0Eh	HTYPE	Header Type	80h	00h	RO
20-23h	BASE	Base Address Register	00000001h	00000001h	R/W
2C-2Dh	SVID	Subsystem Vendor ID	00	00	RO
2E-2Fh	SID	Subsystem ID	00	00	RO
3Ch	INTR_LN	Interrupt Line	00h	00h	R/W
3Dh	INTR_PN	Interrupt Pin	01h	02h	RO
60h	USB_RELNUM	USB Release Number	10h	10	RO
C0-C1h	USB_LEGKEY	USB Legacy Keyboard/ Mouse Control	2000h	2000h	R/W
C4h	USB_RES	USB Resume Enable	00h	00h	R/W

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

10.1.1 Offset 00 - 01h: VID—Vendor Identification Register (USB—D29:F0/F1)

Table 352. Offset 00 - 01h: VID—Vendor Identification Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	00 - 01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel.	RO

10.1.2 Offset 02 - 03h: DID—Device Identification Register (USB—D29:F0/F1)

Table 353. Offset 02 - 03h: DID—Device Identification Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	02-03h	***Attribute:***	Read-Only
Default Value:	Function 0: 25A9h Function 1: 25AAh	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Device ID Value	This is a 16-bit value assigned to the Intel® 6300ESB ICH USB Host Controllers.	RO

10.1.3 Offset 04 - 05h: CMD—Command Register (USB—D29:F0/F1)

Table 354. Offset 04 - 05h: CMD—Command Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	04-05h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:10	Reserved	Reserved.	RO
9	Fast Back-to-back Enable (FBE)	Reserved as '0'.	RO
8	SERR# Enable	Reserved as '0'.	RO
7	Wait Cycle Control	Reserved as '0'.	RO
6	Parity Error Response	Reserved as '0'.	RO
5	VGA Palette Snoop	Reserved as '0'.	RO
4	Postable Memory Write Enable (PMWE)	Reserved as '0'.	RO
3	Special Cycle Enable (SCE)	Reserved as '0'.	RO
2	Bus Master Enable (BME)	When set, the Intel® 6300ESB ICH may act as a master on the PCI bus for USB transfers.	R/W
1	Memory Space Enable (MSE)	Reserved as '0'.	RO
0	I/O Space Enable (IOSE)	This bit controls access to the I/O space registers. 0 = Disable 1 = Enable accesses to the USB I/O registers. The Base Address register for USB should be programmed before this bit is set.	R/W

10.1.4 Offset 06 - 07h: STA—Device Status Register (USB—D29:F0/F1)

Table 355. Offset 06 - 07h: STA—Device Status Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	06 - 07h	***Attribute:***	Read/Write Clear
Default Value:	0280h	***Size:***	16-bit

Bits	Name	Description	Access
15:14	Reserved	Reserved as '00b'.	RO
13	Received Master-Abort Status (RMA)	0 = Software clears this bit by writing a '1' to the bit location. 1 = USB, as a master, generated a master-abort.	R/WC
12	Reserved	Reserved. Always read as '0'.	RO
11	Signaled Target-Abort Status (STA)	0 = Software clears this bit by writing a '1' to the bit location. 1 = USB function is targeted with a transaction that the Intel® 6300ESB ICH terminates with a target abort.	R/WC
10:9	DEVSEL# Timing Status (DEVT)	This 2-bit field defines the timing for DEVSEL# assertion. These read only bits indicate the Intel® 6300ESB ICH's DEVSEL# timing when performing a positive decode. The Intel® 6300ESB ICH generates DEVSEL# with medium timing for USB.	RO
8	Data Parity Error Detected	Reserved as '0'.	RO
7	Fast Back-to-Back Capable	Reserved as '1'.	RO
6	User Definable Features (UDF)	Reserved as '0'.	RO
5	66 MHz Capable	Reserved as '0'.	RO
4:0	Reserved	Reserved.	RO

10.1.5 Offset 08h: RID—Revision Identification Register (USB—D29:F0/F1)

Table 356. Offset 08h: RID—Revision Identification Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

10.1.6 Offset 09h: PI—Programming Interface (USB—D29:F0/F1)

Table 357. Offset 09h: PI—Programming Interface (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	09h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Programming Interface	00h = No specific register level programming interface defined.	RO

10.1.7 Offset 0Ah: SCC—Sub Class Code Register (USB—D29:F0/F1)

Table 358. Offset 0Ah: SCC—Sub Class Code Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	03h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub Class Code	03h = Universal Serial Bus Host Controller.	RO

10.1.8 Offset 0Bh: BCC—Base Class Code Register (USB—D29:F0/F1)

Table 359. Offset 0Bh: BCC—Base Class Code Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	0Ch	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	0Ch = Serial Bus controller.	RO

10.1.9 Offset 0Dh: MLT—Master Latency Timer

Note: Since the USB controller is internally implemented with arbitration through the Hub Interface, not PCI, it does not need a master latency timer. The bits are fixed at '0'.

Table 360. Offset 0Dh: MLT—Master Latency Timer

Device:	29	**Function:**	0/1
Offset:	0Dh	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
	MLT	These bits are fixed at '0'.	

10.1.10 Offset 0Eh: HTYPE—Header Type Register (USB—D29:F0/F1)

Note: For function 1, this register is hard-wired to 00h. For function 0, bit 7 is determined by the values in bits 15 and 9 of the function disable register (D31:F0:F2h).

Table 361. Offset 0Eh: HTYPE—Header Type Register (USB—D29:F0/F1)

Device:	29	**Function:**	0/1
Offset:	0Eh	**Attribute:**	Read-Only
Default Value:	FN 0: 80h FN 1: 00h	**Size:**	8-bit

Bits	Name	Description	Access
7	Multi-Function Bit	**Multi-Function Device** — RO. 0 = Single-function device. 1 = Multi-function device. Since the upper functions in this device can be individually hidden, this bit is based on the function-disable bits in Device 31, Function 0, Offset F2h as follows: **D29:F0 (Bit 15) / D29:F1 (Bit 9) / Multi-Function Bit** 0 / X / 1 X / X / 1 X / 0 / 1 1 / 1 / 0	
6:0	Configuration Layout	Hardwired to 00h, which indicates the standard PCI configuration layout.	

10.1.11 Offset 20 - 23h: BASE—Base Address Register (USB—D29:F0/F1)

Table 362. Offset 20 - 23h: BASE—Base Address Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	20-23h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:5	Base Address	Bits [15:5] correspond to I/O address signals AD [15:5], respectively. This gives 32 bytes of relocatable I/O space.	R/W
4:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is hardwired to '1', indicating that the base address field in this register maps to I/O space.	RO

10.1.12 Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (USB—D29:F0/F1)

Table 363. Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	2Ch - 2Dh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Subsystem Vendor ID (SVID)	The SVID register, in combination with the Subsystem ID (SID) register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads to this register is the same as that which was written by BIOS into the IDE_SVID register.	RO

10.1.13 Offset 2Eh-2Fh: SID—Subsystem ID (USB—D29:F0/F1)

Table 364. Offset 2Eh-2Fh: SID—Subsystem ID (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	2Eh - 2Fh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Subsystem ID (SID)	The SID register, in combination with the SVID register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads to this register is the same as that which was written by BIOS into the IDE_SID register.	R/WO

10.1.14 Offset 3Ch: INTR_LN—Interrupt Line Register (USB—D29:F0/F1)

Table 365. Offset 3Ch: INTR_LN—Interrupt Line Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	3Ch	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

10.1.15 Offset 3Dh: INTR_PN—Interrupt Pin Register (USB—D29:F0/F1)

Table 366. Offset 3Dh: INTR_PN—Interrupt Pin Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	Function 0: 01h Function 1: 02h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	RO
2:0	Interrupt Pin	The values of 01h and 02h, in function 0 and 1, respectively, indicate to software that the corresponding Intel® 6300ESB ICH classic USB controllers drive the INTA# and INTB# PCI signals. Note that this does not determine the mapping to the Intel® 6300ESB ICH PIRQ inputs. Function 0 will drive PIRQA. Function 1 will drive PIRQD. Function 1 does not use the corresponding mapping in order to spread the interrupts with AC'97, which has historically been mapped to PIRQB	RO

10.1.16 Offset 60h: USB_RELNUM—USB Release Number Register (USB—D29:F0/F1)

Table 367. Offset 60h: USB_RELNUM—USB Release Number Register (USB—D29:F0/F1)

Device:	29	***Function:***	0/1
Offset:	60h	***Attribute:***	Read-Only
Default Value:	10h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Serial Bus Release Number	10h = Indicates that the USB controller is compliant with the *USB Specification, Release 1.0*.	RO

10.1.17 Offset C0 - C1h: USB_LEGKEY—USB Legacy Keyboard/ Mouse Control Register (USB—D29:F0/F1)

Note: This register is implemented separately in each of the USB UHCI functions. However, the enable and status bits for the trapping logic are OR'd and shared, respectively, since their functionality is not specific to any one host controller. See Section 19.6, "Logical Device 7 (07H): Port 60/64 Emulation" for details on the relationship between the same bits in each of the host controllers. System Software should ensure that the host controller and an external PCI agent are not simultaneously executing a keyboard sequence to Port 60h and 64h. This event is not supported, and the results will be indeterminate.

Table 368. Offset C0 - C1h: USB_LEGKEY—USB Legacy Keyboard/ Mouse Control Register (USB—D29:F0/F1) (Sheet 1 of 3)

Device:	29	**Function:**	0/1
Offset:	C0-C1h	**Attribute:**	Read/Write
Default Value:	2000h	**Size:**	16-bit

Bits	Name	Description	Access
15	SMI Caused by End of Pass-through (SMIBY-ENDPS)	Indicates if the event occurred. Note that even when the corresponding enable bit is not set in bit 7, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a '1' to the bit location in any of the controllers. 1 = Event Occurred	R/WC
14	Reserved	Reserved.	
13	PCI Interrupt Enable (USBPIRQEN)	Used to prevent the USB controller from generating an interrupt due to transactions on its ports. When disabled, it will be configured to generate an SMI using bit 4 of this register. Default to '1' for compatibility with older USB software. 0 = Disable 1 = Enable	R/W
12	SMI Caused by USB Interrupt (SMIBYUSB)	Indicates if an interrupt event occurred from this controller. The interrupt from the controller is taken before the enable in bit 13 has any effect to create this read-only bit. Note that even when the corresponding enable bit is not set in the Bit 4, this bit may still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software should clear the interrupts through the USB controllers. Writing a '1' to this bit will have no effect. 1 = Event Occurred.	RO
11	SMI Caused by Port 64 Write (TRAPBY64W)	Indicates if the event occurred. Note that even when the corresponding enable bit is not set in the bit 3, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Note that the A20Gate Pass-Through Logic allows specific port 64h writes to complete without setting this bit. 0 = Software clears this bit by writing a '1' to the bit location in any of the controllers. 1 = Event Occurred. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 64 Writes initiated from an external PCI agent will not set this bit.	R/WC
10	SMI Caused by Port 64 Read (TRAPBY64R)	Indicates if the event occurred. Note that even when the corresponding enable bit is not set in the bit 2, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a '1' to the bit location in any of the controllers. 1 = Event Occurred. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 64 Reads initiated from an external PCI agent will not set this bit.	R/WC

Table 368. Offset C0 - C1h: USB_LEGKEY—USB Legacy Keyboard / Mouse Control Register (USB—D29:F0/F1) (Sheet 2 of 3)

Device:	29	**Function:**	0/1
Offset:	C0-C1h	**Attribute:**	Read/Write
Default Value:	2000h	**Size:**	16-bit

Bits	Name	Description	Access
9	SMI Caused by Port 60 Write (TRAPBY60W)	Indicates if the event occurred. Note that even when the corresponding enable bit is not set in the bit 1, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Note that the A20Gate Pass-Through Logic allows specific port 64h writes to complete without setting this bit. 0 = Software clears this bit by writing a '1' to the bit location in any of the controllers. 1 = Event Occurred. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 60 Writes initiated from an external PCI agent will not set this bit.	R/WC
8	SMI Caused by Port 60 Read (TRAPBY60R)	Indicates if the event occurred. Note that even when the corresponding enable bit is not set in the bit 0, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a '1' to the bit location in any of the controllers. 1 = Event Occurred. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 60 Reads initiated from an external PCI agent will not set this bit.	R/WC
7	SMI at End of Pass-through Enable (SMIATENDPS)	May need to cause SMI at the end of a pass-through. May occur when an SMI is generated in the middle of a pass through and needs to be serviced later. 0 = Disable 1 = Enable	R/W
6	Pass-Through State (PSTATE)	0 = When software needs to reset this bit, it should set bit 5 in all of the host controllers to 0. 1 = Indicates that the state machine is in the middle of an A20GATE pass-through sequence.	RO
5	A20Gate Pass-Through Enable (A20PASSEN)	0 = Disable. 1 = Allows A20GATE sequence Pass-Through function. A specific cycle sequence involving writes to port 60h and 64h does not result in the setting of the SMI status bits.	R/W
4	SMI on USB IRQ Enable (USBSMIEN)	0 = Disable 1 = USB interrupt will cause an SMI event.	R/W
3	SMI on Port 64 Writes Enable (64WEN)	0 = Disable 1 = A '1' in bit 11 will cause an SMI event. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 64 Writes initiated from an external PCI agent will not set this bit.	R/W

Table 368. Offset C0 - C1h: USB_LEGKEY—USB Legacy Keyboard/ Mouse Control Register (USB—D29:F0/F1) (Sheet 3 of 3)

Device:	29	**Function:**	0/1
Offset:	C0-C1h	**Attribute:**	Read/Write
Default Value:	2000h	**Size:**	16-bit

Bits	Name	Description	Access
2	SMI on Port 64 Reads Enable (64REN)	0 = Disable 1 = A '1' in bit 10 will cause an SMI event. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 64 Reads initiated from an external PCI agent will not set this bit.	R/W
1	SMI on Port 60 Writes Enable (60WEN)	0 = Disable 1 = A '1' in bit 9 will cause an SMI event. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 60 Writes initiated from an external PCI agent will not set this bit.	R/W
0	SMI on Port 60 Reads Enable (60REN)	0 = Disable 1 = A '1' in bit 8 will cause an SMI event. **NOTE:** If bit 7 of the ETR1 (D31:F0, offset F4h ETR1) is set. See Section 8.1.37, "Offset F4: ETR1—PCI-X Extended Features Register (LPC I/F—D31:F0)" for more information. Port 60 Reads initiated from an external PCI agent will not set this bit.	R/W

10.1.18 Offset C4h: USB_RES—USB Resume Enable Register (USB—D29:F0/F1)

Note: This register is in the Resume Well.

Table 369. Offset C4h: USB_RES—USB Resume Enable Register (USB—D29:F0/F1)

Device:	29	**Function:**	0/1
Offset:	C4h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1	PORT1EN	0 = The USB controller will not look at this port for a wakeup event. 1 = The USB controller will monitor this port for remote wakeup and connect/disconnect events	R/W
0	PORT0EN	0 = The USB controller will not look at this port for a wakeup event. 1 = The USB controller will monitor this port for remote wakeup and connect/disconnect events.	R/W

10.2 USB I/O Registers

Some of the read/write register bits that deal with changing the state of the USB hub ports function so that, on read-back, they reflect the current state of the port, and not necessarily the state of the last write to the register. This allows the software to poll the state of the port and wait until it is in the proper state before proceeding. A Host Controller Reset, Global Reset, or Port Reset will immediately terminate a transfer on the affected ports and disable the port. This affects the USBCMD register, bit [4] and the PORTSC registers, bits [12,6,2]. See individual bit descriptions for more detail.

Table 370. USB I/O Registers

Offset	Mnemonic	Register	Default	Type
00-01	USBCMD	USB Command Register	0000h	R/W*
02-03	USBSTA	USB Status Register	0020h	R/WC
04-05	USBINTR	Interrupt Enable	0000h	R/W
06-07	FRNUM	Frame Number	0000h	R/W (see NOTE:)
08-0B	FRBASEADD	Frame List Base Address	Undefined	R/W
0C	SOFMOD	Start of Frame Modify	40h	R/W
0D-0F		Reserved	0	RO
10-11	PORTSC0	Port 0 Status/Control	0080h	R/WC (see NOTE:)
12-13	PORTSC1	Port 1 Status/Control	0080h	R/WC (see NOTE:)
14-17		Reserved	0	RO

NOTE: These registers are WORD writable only. Byte writes to these registers have unpredictable effects.

10.2.1 Offset 00 - 01h: USBCMD—USB Command Register

Note: The Command Register indicates the command to be executed by the serial bus host controller. Writing to the register causes a command to be executed. The table following the bit description provides additional information on the operation of the Run/Stop and Debug bits.

Table 371. Offset 00 - 01h: USBCMD—USB Command Register (Sheet 1 of 3)

Device:	29	**Function:**	0/1
Offset:	00-01h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15:7	Reserved	Reserved.	
8	Loop Back Test Mode	0 = Disable loop back test mode. 1 = The Intel® 6300ESB ICH is in loop back test mode. When both ports are connected together, a write to one port will be seen on the other port, and the data will be stored in I/O offset 18h.	R/W
7	Max Packet (MAXP)	This bit selects the maximum packet size that may be used for full-speed bandwidth reclamation at the end of a frame. This value is used by the Host Controller to determine whether it should initiate another transaction based on the time remaining in the SOF counter. Use of reclamation packets larger than the programmed size will cause a Babble error when executed during the critical window at frame end. The Babble error results in the offending endpoint being stalled. Software is responsible for ensuring that any packet that could be executed under bandwidth reclamation be within this size limit. 0 = 32 bytes 1 = 64 bytes	R/W
6	Configure Flag (CF)	This bit has no effect on the hardware. It is provided only as a semaphore service for software. 0 = Indicates that software has not completed host controller configuration. 1 = HCD software sets this bit as the last action in its process of configuring the Host Controller.	R/W
5	Software Debug (SWDBG)	The SWDBG bit must only be manipulated when the controller is in the stopped state. This may be determined by checking the HCHalted bit in the USBSTS register. 0 = Normal Mode. 1 = Debug mode. In SW Debug mode, the Host Controller clears the Run/Stop bit after the completion of each USB transaction. The next transaction is executed when software sets the Run/Stop bit back to ‘1’.	R/W
4	Force Global Resume (FGR)	0 = Software resets this bit to ’0’ after 20 ms has elapsed to stop sending the Global Resume signal. At that time, all USB devices should be ready for bus activity. The ’1’ to ’0’ transition causes the port to send a low speed EOP signal. This bit will remain a ’1’ until the EOP has completed. 1 = Host Controller sends the Global Resume signal on the USB and sets this bit to ’1’ when a resume event (connect, disconnect, or K-state) is detected while in global suspend mode.	R/W

Table 371. Offset 00 - 01h: USBCMD—USB Command Register (Sheet 2 of 3)

Device:	29	**Function:**	0/1
Offset:	00-01h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
3	Enter Global Suspend Mode (EGSM)	0 = Software resets this bit to '0' to come out of Global Suspend mode. Software writes this bit to '0' at the same time that Force Global Resume (bit 4) is written to '0' or after writing bit 4 to 0. 1 = Host Controller enters the Global Suspend mode. No USB transactions occur during this time. The Host Controller is able to receive resume signals from USB and interrupt the system. Software must ensure that the Run/Stop bit (bit 0) is cleared prior to setting this bit.	R/W

Table 371. Offset 00 - 01h: USBCMD—USB Command Register (Sheet 3 of 3)

Device:	29	**Function:**	0/1
Offset:	00-01h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
2	Global Reset (GRESET)	0 = This bit is reset by the software after a minimum of 10 ms has elapsed as specified in Chapter 7 of the USB Specification. 1 = Global Reset. The Host Controller sends the global reset signal on the USB and then resets all its logic, including the internal hub registers. The hub registers are reset to their power on state. Chip Hardware Reset has the same effect as Global Reset (bit 2), except that the Host Controller does not send the Global Reset on USB.	R/W
1	Host Controller Reset (HCRESET)	The effects of HCRESET on Hub registers are slightly different from Chip Hardware Reset and Global USB Reset. The HCRESET affects bits [8,3:0] of the Port Status and Control Register (PORTSC) of each port. HCRESET resets the state machines of the Host Controller including the Connect/ Disconnect state machine (one for each port). When the Connect/Disconnect state machine is reset, the output that signals connect/disconnect are negated to 0, effectively signaling a disconnect, even if a device is attached to the port. This virtual disconnect causes the port to be disabled. This disconnect and disabling of the port causes bit 1 (connect status change) and bit 3 (port enable/disable change) of the PORTSC to get set. The disconnect also causes bit 8 of PORTSC to reset. About 64 bit times after HCRESET goes to '0', the connect and low-speed detect will take place, and bits 0 and 8 of the PORTSC will change accordingly. 0 = Reset by the Host Controller when the reset process is complete. 1 = Reset. When this bit is set, the Host Controller module resets its internal timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated.	R/W
0	Run/Stop (RS)	When set to '1', the Intel® 6300ESB ICH proceeds with execution of the schedule. The Intel® 6300ESB ICH continues execution as long as this bit is set. When this bit is cleared, the Intel® 6300ESB ICH completes the current transaction on the USB and then halts. The HC Halted bit in the status register indicates when the Host Controller has finished the transaction and has entered the stopped state. The Host Controller clears this bit when the following fatal errors occur: consistency check failure, PCI Bus errors. 0 = Stop 1 = Run **NOTE:** This bit should only be cleared when there are no active Transaction Descriptors in the executable schedule or software will reset the host controller prior to setting this bit again.	R/W

Table 372. Run/Stop, Debug Bit Interaction SWDBG (Bit 5), Run/Stop (Bit 0) Operation

SWDBG (Bit 5)	Run/Stop (Bit 0)	Description
0	0	When executing a command, the Host Controller completes the command and then stops. The 1.0 ms frame counter is reset and command list execution resumes from start of frame using the frame list pointer selected by the current value in the FRNUM register. (While Run/Stop = 0, the FRNUM register may be reprogrammed).
0	1	Execution of the command list resumes from Start Of Frame using the frame list pointer selected by the current value in the FRNUM register. The Host Controller remains running until the Run/Stop bit is cleared (by software or hardware).
1	0	When executing a command, the Host Controller completes the command and then stops and the 1.0 ms frame counter is frozen at its current value. All statuses are preserved. The Host Controller begins execution of the command list from where it left off when the Run/Stop bit is set.
1	1	Execution of the command list resumes from where the previous execution stopped. The Run/Stop bit is set to '0' by the Host Controller when a TD is being fetched. This causes the Host Controller to stop again after the execution of the TD (single step). When the Host Controller has completed execution, the HC Halted bit in the Status Register is set.

When the USB Host Controller is in Software Debug Mode (USBCMD Register bit 5=1), the single stepping software debug operation is as follows:

To Enter Software Debug Mode:

1. HCD puts Host Controller in Stop state by setting the Run/Stop bit to '0'.
2. HCD puts Host Controller in Debug Mode by setting the SWDBG bit to '1'.
3. HCD sets up the correct command list and Start Of Frame value for starting point in the Frame List Single Step Loop.
4. HCD sets Run/Stop bit to '1'.
5. Host Controller executes next active TD, sets Run/Stop bit to '0' and stops.
6. HCD reads the USBCMD register to check if the single step execution is completed (HCHalted = 1).
7. HCD checks results of TD execution. Go to step 4 to execute next TD or step 8 to end Software Debug mode.
8. HCD ends Software Debug mode by setting SWDBG bit to '0'.
9. HCD sets up normal command list and Frame List table.
10. HCD sets Run/Stop bit to '1' to resume normal schedule execution.

In Software Debug mode, when the Run/Stop bit is set, the Host Controller starts. When a valid TD is found, the Run/Stop bit is reset. When the TD is finished, the HCHalted bit in the USBSTS register (bit 5) is set.

The SW Debug mode skips over inactive TDs and only halts after an active TD has been executed. When the last active TD in a frame has been executed, the Host Controller waits until the next SOF is sent and then fetches the first TD of the next frame before halting.

This HCHalted bit may also be used outside of Software Debug mode to indicate when the Host Controller has detected the Run/Stop bit and has completed the current transaction. Outside of the Software Debug mode, setting the Run/Stop bit to '0'

always resets the SOF counter so that when the Run/Stop bit is set, the Host Controller starts over again from the frame list location pointed to by the Frame List Index (see FRNUM Register description) rather than continuing where it stopped.

10.2.2 Offset 02 - 03h: USBSTA—USB Status Register

Note: This register indicates pending interrupts and various states of the Host Controller. The status resulting from a transaction on the serial bus is not indicated in this register. Software sets a bit to '0' in this register by writing a '1' to it.

Table 373. Offset 02 - 03h: USBSTA—USB Status Register (Sheet 1 of 2)

Device:	29	***Function:***	X
Offset:	02 - 03h	***Attribute:***	Rea/Write Clear
Default Value:	0020h	***Size:***	16-bit

Bits	Name	Description	Access
15:6	Reserved	Reserved.	
5	HCHalted	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = The Host Controller has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller hardware (debug mode or an internal error). Default.	R/WC
4	Host Controller Process Error	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = The Host Controller has detected a fatal error. This indicates that the Host Controller suffered a consistency check failure while processing a Transfer Descriptor. An example of a consistency check failure would be finding an illegal PID field while processing the packet header portion of the TD. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further schedule execution. A hardware interrupt is generated to the system.	R/WC
3	Host System Error	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = A serious error occurred during a host system access involving the Host Controller module. In a PCI system, conditions that set this bit to '1' include PCI Parity error, PCI Master Abort, and PCI Target Abort. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs. A hardware interrupt is generated to the system.	R/WC

Table 373. Offset 02 - 03h: USBSTA—USB Status Register (Sheet 2 of 2)

Device:	29	**Function:**	X
Offset:	02 - 03h	**Attribute:**	Rea/Write Clear
Default Value:	0020h	**Size:**	16-bit

Bits	Name	Description	Access
2	Resume Detect (RSM_DET)	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = The Host Controller received a "RESUME" signal from a USB device. This is only valid when the Host Controller is in a global suspend state (bit 3 of Command register = 1).	R/WC
1	USB Error Interrupt	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = Completion of a USB transaction resulted in an error condition (e.g., error counter underflow). When the TD on which the error interrupt occurred also had its IOC bit set, both this bit and Bit '0' are set.	R/WC
0	USB Interrupt (USBINT)	0 = Software resets this bit to '0' by writing a '1' to the bit position. 1 = The Host Controller sets this bit when the cause of an interrupt is a completion of a USB transaction whose Transfer Descriptor had its IOC bit set. Also set when a short packet is detected (actual length field in TD is less than maximum length field in TD) and short packet detection is enabled in that TD.	R/WC

10.2.3 Offset Base + (04 - 05h): USBINTR—Interrupt Enable Register

Note: This register enables and disables reporting of the corresponding interrupt to the software. When a bit is set and the corresponding interrupt is active, an interrupt is generated to the host. Fatal errors (Host Controller Processor Error-bit 4, USBSTS Register) cannot be disabled by the host controller. Interrupt sources that are disabled in this register still appear in the Status Register to allow the software to poll for events.

Table 374. Offset Base + (04 - 05h): USBINTR—Interrupt Enable Register (Sheet 1 of 2)

Device:	29	**Function:**	X
Offset:	Base + (04-05h)	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15:5	Reserved	Reserved.	
4	Scratchpad	Scratchpad.	R/W
3	Short Packet Interrupt Enable	0 = Disabled. 1 = Enabled.	R/W

Table 374. Offset Base + (04 - 05h): USBINTR—Interrupt Enable Register (Sheet 2 of 2)

Device:	29	***Function:***	X
Offset:	Base + (04-05h)	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
2	Interrupt On Complete (IOC) Enable	0 = Disabled. 1 = Enabled.	R/W
1	Resume Interrupt Enable	0 = Disabled. 1 = Enabled.	R/W
0	Timeout/CRC Interrupt Enable	0 = Disabled. 1 = Enabled.	R/W

10.2.4 Offset Base + (06 - 07h): FRNUM—Frame Number Register

Note: Bits [10:0] of this register contain the current frame number included in the frame SOF packet. This register reflects the count value of the internal frame number counter. Bits [9:0] are used to select a particular entry in the Frame List during scheduled execution. This register is updated at the end of each frame time.

Note: This register must be written as a word. Byte writes are not supported. This register cannot be written unless the Host Controller is in the STOPPED state as indicated by the HCHalted bit (USBSTS register). A write to this register while the Run/Stop bit is set (USBCMD register) is ignored.

Table 375. Offset Base + (06 - 07h): FRNUM—Frame Number Register

Device:	29	***Function:***	X
Offset:	Base + (06-07h)	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10:0	Frame List Current Index/Frame Number	Provides the frame number in the SOF Frame. The value in this register increments at the end of each time frame (approximately every 1 ms). In addition, bits [9:0] are used for the Frame List current index and correspond to memory address signals [11:2].	R/W

10.2.5 Offset Base + (08 - 0Bh): FRBASEADD—Frame List Base Address

Note: This 32-bit register contains the beginning address of the Frame List in the system memory. HCD loads this register prior to starting the schedule execution by the Host Controller. When written, only the upper 20 bits are used. The lower 12 bits are written as '0' (4-Kbyte alignment). The contents of this register are combined with the frame number counter to enable the Host Controller to step through the Frame List in sequence. The two least significant bits are always 00. This requires DWORD alignment for all list entries. This configuration supports 1024 Frame List entries.

Table 376. Offset Base + (08 - 0Bh): FRBASEADD—Frame List Base Address

Device:	29	***Function:***	X
Offset:	Base + (08-0Bh)	***Attribute:***	Read/Write
Default Value:	Undefined	***Size:***	32-bit

Bits	Name	Description	Access
31:12	Base Address	These bits correspond to memory address signals [31:12], respectively.	R/W
11:0	Reserved	Reserved.	

10.2.6 Offset Base + OCh: SOFMOD—Start of Frame Modify Register

Note: This 1-byte register is used to modify the value used in the generation of SOF timing on the USB. Only the 7 least significant bits are used. When a new value is written into these 7 bits, the SOF timing of the next frame will be adjusted. This feature may be used to adjust out any offset from the clock source that generates the clock that drives the SOF counter. This register may also be used to maintain real-time synchronization with the rest of the system so that all devices have the same sense of real time. Using this register, the frame length may be adjusted across the full range required by the USB specification. Its initial programmed value is system dependent based on the accuracy of hardware USB clock and is initialized by system BIOS. It may be reprogrammed by USB system software at any time. Its value will take effect from the beginning of the next frame. This register is reset upon a Host Controller Reset or Global Reset. Software must maintain a copy of its value for reprogramming when necessary.

Table 377. Offset Base + OCh: SOFMOD—Start of Frame Modify Register

Device:	29	***Function:***	X
Offset:	Base + (0Ch)	***Attribute:***	Read/Write
Default Value:	40h	***Size:***	8-bit

Bits	Name	Description	Access
7	Reserved	Reserved.	
6:0	SOF Timing Value	Guidelines for the modification of frame time are contained in Chapter 7 of the USB Specification. The SOF cycle time (number of SOF counter clock periods to generate a SOF frame length) is equal to 11936 + value in this field. The default value is decimal 64 which gives a SOF cycle time of 12000. For a 12 MHz SOF counter clock input, this produces a 1 ms Frame period. The following table indicates what SOF Timing Value to program into this field for a certain frame period. **Frame Length (# 12 MHz Clocks) (decimal) / SOF Reg. Value (decimal)** 11936 / 0 11937 / 1 . / . . / . 11999 / 63 12000 / 64 12001 / 65 . / . . / . 12062 / 126 12063 / 127	R/W

10.2.7 PORTSC[0,1]—Port Status and Control Register

Note: For Function 0 this applies to the Intel® 6300ESB ICH USB ports 0 and 1; for Function 1 this applies to the Intel® 6300ESB ICH USB ports 2 and 3.

After a Power-up reset, Global reset, or Host Controller reset, the initial conditions of a port are: no device connected, Port disabled, and the bus line status is 00 (single-ended '0').

Table 378. PORTSC[0,1]—Port Status and Control Register (Sheet 1 of 2)

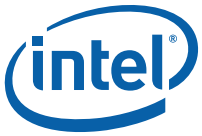
Device:	X	***Function:***	X
Offset:	Port 0/2: 10-11h Port 1/3: 12-13h	***Attribute:***	Read/Write
Default Value:	0080h	***Size:***	16-bit

Bits	Name	Description	Access
15:13	Reserved	Reserved.	RO
12	Suspend	This bit should not be written to a '1' when a global suspend is active (bit 3 =1 in the USBCMD register). Bit 2 and bit 12 of this register define the hub states as follows: **Bits [12, 2] — Hub State** X 0 — Disable 0 1 — Enable 1 1 — Suspend When in suspend state, downstream propagation of data is blocked on this port, except for single-ended '0' resets (global reset and 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 when there is a transaction currently in progress on the USB. 1 = Port in suspend state. 0 = Port not in suspend state. **NOTE:** Normally, if a transaction is in progress when this bit is set, the port will be suspended when the current transaction completes. However, in the case of a specific error condition (out transaction with babble), the Intel® 6300ESB ICH may issue a start-of-frame, then suspend the port.	R/W
11	Overcurrent Indicator	Set by hardware. 0 = Software clears this bit by writing a '1' to the bit position. 1 = Overcurrent pin has gone from inactive to active on this port.	R/WC
10	Overcurrent Active	This bit is set and cleared by hardware. 0 = Indicates that the overcurrent pin is inactive (high). 1 = Indicates that the overcurrent pin is active (low).	RO
9	Port Reset	0 = Port is not in Reset. 1 = Port is in Reset. When set, the port is disabled and sends the USB Reset signaling.	RO
8	Low Speed Device Attached (LS)	Writes have no effect. 0 = Full-speed device is attached. 1 = Low-speed device is attached to this port.	RO

Table 378. PORTSC[0,1]—Port Status and Control Register (Sheet 2 of 2)

Device:	X	**Function:**	X
Offset:	Port 0/2: 10-11h Port 1/3: 12-13h	**Attribute:**	Read/Write
Default Value:	0080h	**Size:**	16-bit

Bits	Name	Description	Access
7	Reserved	Reserved. Always read as '1'.	RO
6	Resume Detect (RSM_DET)	Software sets this bit to a '1' to drive resume signaling. The Host Controller sets this bit to a '1' when a J-to-K transition is detected for at least 32 microseconds while the port is in the Suspend state. The Intel® 6300ESB ICH will then reflect the K-state back onto the bus as long as the bit remains a '1' and the port is still in the suspend state (bit 12, 2 are '11'). Writing a '0' (from '1') causes the port to send a low-speed EOP. This bit will remain a '1' until the EOP has completed. 0 = No resume (K-state) detected/driven on port. 1 = Resume detected/driven on port.	R/W
5:4	Line Status	These bits reflect the D+ (bit 4) and D- (bit 5) signals lines' logical levels. These bits are used for fault detect and recovery as well as for USB diagnostics. This field is updated at EOF2 time (See Chapter 11 of the USB Specification).	RO
3	Port Enable/Disable Change	For the root hub, this bit gets set only when a port is disabled due to disconnect on that port or due to the appropriate conditions existing at the EOF2 point (See Chapter 11 of the USB Specification). 0 = No change. Software clears this bit by writing a '1' to the bit location. 1 = Port enabled/disabled status has changed.	R/WC
2	Port Enabled/Disabled (PORT_EN)	Ports may be enabled by host software only. Ports may 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 and that there may be a delay in disabling or enabling a port when there is a transaction currently in progress on the USB. 0 = Disable 1 = Enable	R/W
1	Connect Status Change	Indicates that a change has occurred in the port's Current Connect Status (see bit 0). The hub device sets this bit for any changes to the port device connect status even when system software has not cleared a connect status change. If, for example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be setting an already-set bit (i.e., the bit will remain set). However, the hub transfers the change bit only once when the Host Controller requests a data transfer to the Status Change endpoint. System software is responsible for determining state change history in such a case. 0 = No change. Software clears this bit by writing a '1' to the bit location. 1 = Change in Current Connect Status.	R/WC
0	Current Connect Status	This value reflects the current state of the port and may not correspond directly to the event that caused the Connect Status Change bit (Bit 1) to be set. 0 = No device is present. 1 = Device is present on port.	RO



USB EHCI Controller Registers (D29:F7) 11

11.1 USB EHCI Configuration Registers (D29:F7)

Note: Registers that are not shown should be treated as Reserved (see Section 6.2, "PCI Configuration Map" for details).

Offset	Reg. Name/Function	Default Value	Type
00-01h	Vendor ID	8086h	RO
02-03h	Device ID	25ADh	RO
04-05h	Command Register	0000h	RW
06-07h	Device Status	0290h	RW
08h	Revision ID (RID)	See NOTE:	RO
09h	Programming Interface	20h	RO
0Ah	Sub Class Code	03h	RO
0Bh	Base Class Code	0Ch	RO
0Dh	Master Latency Timer	00h	RO
0Eh	Header Type	00h	RO
10-13h	Memory Base Address Register	00000000h	RW
2C-2Dh	Subsystem Vendor ID	XXXXh	RW-Special
2E-2Fh	Subsystem ID	XXXXh	RW-Special
34h	Capabilities Pointer	50h	RO
3Ch	Interrupt Line	00h	RW
3Dh	Interrupt Pin	04h	RO
50h	Power Management Capability ID	01h	RO
51h	Next Item Ptr	58h	RO
52-53h	Power Management Capabilities	C9C2h	RO-Special
54-55h	Power Management Control/ Status	0000h	RW
57h	Power Management Data	00h	RO
58h	Debug Port Capability ID	0Ah	RO
59h	Next Item Pointer #2	00h	RO
5A-5Bh	Debug Port Base Offset	2080h	RO
60h	USB Release Number	20h	RO
61h	Frame Length Adjustment	20h	RW
62-63h	Power Wake Capabilities	007Fh	RW
64-65h	Classic USB Override	0000	RO

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

Offset	Reg. Name/Function	Default Value	Type
66-67h	Reserved	0	RO
68-6Bh	USB EHCI Legacy Support Extended Capability	00000001h	Read/Write
6C-6Fh	USB EHCI Legacy Support Control/Status	00000000h	Read/Write
70-73h	Intel Specific USB EHCI SMI	00000000h	Read/Write
74-7F	Reserved	0	Read Only
80h	Access Control	00h	Read/Write
81-FFh	Reserved	0	Read Only
DCh	HS_ Ref_V_USB HS Reference Voltage Register	00000000h	Read/Write

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

11.1.1 Offset 04 - 05h: Command Register

Table 379. Offset 04 - 05h: Command Register

Device:	29	***Function:***	7
Offset:	04-05h	***Attribute:***	Read Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Reserved	Reserved.	
9	Fast Back-to-Back Enable (FBE)	Reserved as 0.	RO
8	SERR# Enable	0 = Disables the Enhanced Host Controller's (EHC's) capability to generate an SERR#. 1 = The EHC is capable of generating internally SERR# when it receives a completion status other than "successful" for '1' of its DMA-initiated memory reads on the Hub Interface (and subsequently on its internal interface).	R/W
7	Wait Cycle Control	Reserved as 0.	RO
6	Parity Error Response	Reserved as 0.	RO
5	VGA Palette Snoop	Reserved as 0.	RO
4	Postable Memory Write Enable (PMWE)	Reserved as 0.	RO
3	Special Cycle Enable (SCE)	Reserved as 0.	RO
2	Bus Master Enable (BME)	0 = Disables this functionality. 1 = Enables the Intel® 6300ESB ICH. May act as a master on the PCI bus for USB transfers.	R/W
1	Memory Space Enable (MSE)	0 = Disables this functionality. 1 = Enables accesses to the USB EHCI registers. The Base Address register for USB EHCI should be programmed before this bit is set	R/W
0	I/O Space Enable (IOSE)	Reserved as 0.	RO

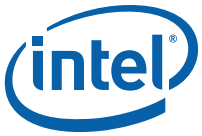
11.1.2 Offset 06 - 07h: Device Status

Table 380. Offset 06 - 07h: Device Status

Device:	29	**Function:**	7
Offset:	06 - 07h	**Attribute:**	Read/ Write
Default Value:	0290h	**Size:**	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	This bit is set by the Intel® 6300ESB ICH whenever a parity error is seen on the internal interface to the USB host controller due to a parity error on Hub Interface, regardless of the setting of bit 6 or bit 8 in the Command register or any other conditions. Software clears this bit by writing a '1' to this bit location. Note that the Parity Error Response bit in the HL-to-PCI bridge should be set in order for the Hub Interface parity errors to be forwarded to the USB2 interface. This is a result of the point-to-point nature of the Hub Interface	R/W
14	Signaled System Error (SSE)	0 = Software clears this bit by writing a '1' to this bit location. 1 = This bit is set by the Intel® 6300ESB ICH whenever it signals SERR# (internally). The SER_EN bit (bit 8 of the Command Register) must be '1' for this bit to be set.	R/W
13	Received Master-Abort Status (RMA)	0 = Software clears this bit by writing a '1' to this bit location. 1 = This bit is set when USB EHCI, as a master, receives a master-abort status on a memory access. This is treated as a Host Error and halts the DMA engines. This event may optionally generate an SERR# by setting the SERR# Enable bit.	R/W
12	Received Target Abort Status (RTA)	0 = Software clears this bit by writing a '1' to this bit location. 1 = This bit is set when USB EHCI, as a master, receives a target abort status on a memory access. This is treated as a Host Error and halts the DMA engines. This event may optionally generate an SERR# by setting the SERR# Enable bit.	R/W
11	Signaled Target-Abort Status (STA)	This bit is used to indicate when the USB EHCI function responds to a cycle with a target abort. There is no reason for this to happen, so this bit will be hard-wired to '0'.	RO
10:9	DEVSEL# Timing Status (DEVT)	This 2-bit field defines the timing for DEVSEL# assertion.	RO
8	Master Data Parity Error Detected	0 = Software clears this bit by writing a '1' to this bit location. 1 = This bit is set by the Intel® 6300ESB ICH whenever a data parity error is detected on a USB EHCI read completion packet on the internal interface to the USB EHCI host controller (due to an equivalent data parity error on Hub Interface) and bit 6 of the Command register is set to 1.	R/W
7	Fast Back-to-Back Capable	Reserved as 1.	RO
6	UDF - User Definable Features	Reserved as 0.	RO
5	66 MHz Capable	Reserved as 0.	RO
4	Capabilities List	This bit is hardwired to '1' indicating the presence of a valid capabilities pointer at offset 34h.	RO
3:0	Reserved	Reserved.	

November 2007
Order Number: 300641-004US



11.1.3 Offset 08h: RID—Revision ID Register

Table 381. Offset 08h: RID—Revision ID Register

Device:	29	***Function:***	7
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

11.1.4 Offset 09h: Programming Interface

Table 382. Offset 09h: Programming Interface

Device:	29	***Function:***	7
Offset:	09h	***Attribute:***	Read-Only
Default Value:	20h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Programming Interface	A value of 20h indicates that this USB High-speed Host Controller conforms to the EHCI Specification.	RO

11.1.5 Offset 0Ah: Sub Class Code

Table 383. Offset 0Ah: Sub Class Code

Device:	29	***Function:***	7
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	03h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub Class Code	A value of 03h indicates that this is a Universal Serial Bus Host Controller.	RO

11.1.6 Offset 0Bh: Base Class Code

Table 384. Offset 0Bh: Base Class Code

Device:	29	**Function:**	7
Offset:	0Bh	**Attribute:**	Read-Only
Default Value:	0Ch	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	A value of 0Ch indicates that this is a Serial Bus controller.	RO

11.1.7 Offset 0Dh: Master Latency Timer

Table 385. Offset 0Dh: Master Latency Timer

Device:	29	**Function:**	7
Offset:	0Dh	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Master Latency Timer	Since the USB EHCI controller is internally implemented with arbitration through the Hub Interface (and not PCI), it does not need a master latency timer. These bits will be fixed to 0.	RO

11.1.8 Offset 10 - 13h: Memory Base Address

Table 386. Offset 10 - 13h: Memory Base Address

Device:	29	**Function:**	7
Offset:	10 - 13h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:10	Base Address	Bits [31:10] correspond to memory address signals [31:10], respectively. This gives 1 Kbyte of locatable memory space aligned to 1 Kbyte boundaries.	RW
9:4	Reserved	Reserved.	
3	Prefetchable	This bit is hardwired to 0, indicating that this range should not be prefetched.	RO
2:1	Type	This field is hardwired to 00b indicating that this range may be mapped anywhere within 32-bit address space.	RO
0	Resource Type Indicator (RTE)	This field is hardwired to 00b indicating that this range may be mapped anywhere within 32-bit address space.	RO

11.1.9 Offset 2C - 2Dh: USB EHCI Subsystem Vendor ID

Table 387. Offset 2C - 2Dh: USB EHCI Subsystem Vendor ID

Device:	29	***Function:***	7
Offset:	2C-2Dh	***Attribute:***	Read/Write-Special
Default Value:	XXXXh	***Size:***	16-bit

Bits	Name	Description	Access
15:0	USB EHCI Subsystem Vendor ID	This register, in combination with the USB EHCI Subsystem ID register, enables the operating system to distinguish each subsystem from the others. Writes to this register are enabled when the WRT_RDONLY bit (offset 80h, bit 0) is set to 1.	RW-Special

11.1.10 Offset 2E - 2Fh: SID—USB EHCI Subsystem ID

Table 388. Offset 2E - 2Fh: SID—USB EHCI Subsystem ID

Device:	29	***Function:***	7
Offset:	2E-2Fh	***Attribute:***	Read/Write-Special
Default Value:	XXXXh	***Size:***	16-bit

Bits	Name	Description	Access
15:0	USB EHCI Subsystem ID	This register, in combination with the Subsystem Vendor ID register, enables the operating system to distinguish each subsystem from other(s). BIOS sets the value in this register to identify the Subsystem ID. Writes to this register are enabled when the WRT_RDONLY bit (offset 80h, bit 0) is set to 1.	RW-Special

11.1.11 Offset 34h: Capabilities Pointer

Table 389. Offset 34h: Capabilities Pointer

Device:	29	***Function:***	7
Offset:	34h	***Attribute:***	Read-Only
Default Value:	50h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Capabilities Pointer	This register points to the starting offset of the USB EHCI capabilities ranges.	RO

11.1.12 Offset 3Ch: Interrupt Line

Table 390. Offset 3Ch: Interrupt Line

Device:	29	**Function:**	7
Offset:	3Ch	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is used as a scratchpad register to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

11.1.13 Offset 3Dh: Interrupt Pin

Table 391. Offset 3Dh: Interrupt Pin

Device:	29	**Function:**	7
Offset:	3Dh	**Attribute:**	Read-Only
Default Value:	04h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Interrupt Pin	The value of 04h indicates that the USB EHCI function within the Intel® 6300ESB ICH's multi-function USB device will drive the fourth interrupt pin from the device–NTD# in PCI terms. The value of 04h in function 7 is required because the PCI specification does not recognize more than four interrupts, and older APM-based OSs require that each function within a multi-function device has a different Interrupt Pin Register value. Internally the USB EHCI controller uses PIRQ[H]#.	RO

11.1.14 Offset 50h: PCI Power Management Capability ID

Table 392. Offset 50h: PCI Power Management Capability ID

Device:	29	**Function:**	7
Offset:	50h	**Attribute:**	Read-Only
Default Value:	01h	**Size:**	8-bit

Bits	Name	Description	Access
7:0		A value of 01h indicates that this is a PCI Power Management capabilities field.	

11.1.15 Offset 51h: Next Item Pointer #1

Table 393. Offset 51h: Next Item Pointer #1

Device:	29	**Function:**	7
Offset:	51h	**Attribute:**	Read/Write Special
Default Value:	58h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Next Item Pointer #1	This register defaults to 58h, which indicates that the next capability registers begin at configuration offset 58h. This register is writable when the WRT_RDONLY bit is set. This allows BIOS to effectively hide the Debug Port capability registers when necessary. This register should be written only during system initialization before the plug-and-play software has enabled any master-initiated traffic. Only values of 58h (Debug Port visible) and 00h (Debug Port invisible) are expected to be programmed in this register.	RW-Special

11.1.16 Offset 52 - 53h: Power Management Capabilities

Table 394. Offset 52 - 53h: Power Management Capabilities (Sheet 1 of 2)

Device:	29	**Function:**	7
Offset:	52-53h	**Attribute:**	Read/Write Special
Default Value:	C9C2h	**Size:**	16-bit

Bits	Name	Description	Access
15:11	PME_Support	This 5-bit field indicates the power states in which the function may assert PME#. The Intel® 6300ESB ICH EHC does not support D1 or D2 states. For all other states, the Intel® 6300ESB ICH EHC is capable of generating PME#. Software should never need to modify this field.	RW-Special
10	D2_Support	Hardwired to 0 = D2 State is not supported.	RW-Special
9	D1_Support	Hardwired to 0 = D1 State is not supported.	RW-Special
8:6	Aux_Current	The Intel® 6300ESB ICH EHC reports 375 mA maximum Suspend well current required when in the D3cold state. This value may be written by BIOS when a more accurate value is known.	RW-Special
5	DSI	The Intel® 6300ESB ICH reports 0, indicating that no device-specific initialization is required.	RW-Special
4	Reserved	Reserved.	

Table 394. Offset 52 - 53h: Power Management Capabilities (Sheet 2 of 2)

Device:	29	**Function:**	7
Offset:	52-53h	**Attribute:**	Read/Write Special
Default Value:	C9C2h	**Size:**	16-bit

Bits	Name	Description	Access
3	PME Clock	The Intel® 6300ESB ICH reports 0, indicating that no PCI clock is required to generate PME#.	RW-Special
2:0	Version	The Intel® 6300ESB ICH reports 010b, indicating that it complies with Revision 1.1 of the PCI Power Management Specification.	RW-Special

NOTES:
1. Normally, this register is read-only to report capabilities to the power management software. To report different power management capabilities depending on the system in which the Intel® 6300ESB ICH is used, bits 15:11 and 8:6 in this register are writable when the WRT_RDONLY bit is set. The value written to this register does not affect the hardware other than changing the value returned during a read.
2. Reset: Core well, but not D3-to-D0 warm reset.

11.1.17 Offset 54 - 55h: Power Management Control/Status

Table 395. Offset 54 - 55h: Power Management Control/Status

Device:	29	**Function:**	7
Offset:	54 - 55h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15	PME_Status	This bit is set when the Intel® 6300ESB ICH EHC would normally assert the PME# signal independent of the state of the PME_En bit. Writing a '1' to this bit will clear it and cause the internal PME to deassert when enabled. Writing a '0' has no effect. This bit must be explicitly cleared by the operating system each time the operating system is loaded.	R/WC
14:13	Data_Scale	The Intel® 6300ESB ICH hardwires these bits to 00b because it does not support the associated Data register.	RO
12:9	Data_Select	The Intel® 6300ESB ICH hardwires these bits to 0000b because it does not support the associated Data register.	RO
8	PME_EN	A '1' enables the Intel® 6300ESB ICH EHC to generate an internal PME signal when PME_Status is 1. This bit must be explicitly cleared by the operating system each time it is initially loaded.	R/W

Table 395. Offset 54 - 55h: Power Management Control/Status

Device:	29	**Function:**	7
Offset:	54 - 55h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1:0	PowerState	This 2-bit field is used both to determine the current power state of EHC function and to set a new power state. The definition of the field values are: 00b – D0 state 11b – D3 hot state When software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally; however, the data is discarded and no state change occurs. When in the D3 hot state, the Intel® 6300ESB ICH must not accept accesses to the EHC memory range, but the configuration space must still be accessible. When not in the D0 state, the generation of the interrupt output is blocked. Specifically, the PIRQ[H] is not asserted by the Intel® 6300ESB ICH when not in the D0 state. When software changes this value from the D3hot state to the D0 state, an internal warm (soft) reset is generated, and software must re-initialize the function.	R/W

NOTE: Reset (bits 15, 8): suspend well, and not D3-to-D0 warm reset nor core well reset.

11.1.18 Offset 58h: Debug Port Capability ID

Table 396. Offset 58h: Debug Port Capability ID

Device:	29	**Function:**	7
Offset:	58h	**Attribute:**	Read-Only
Default Value:	0Ah	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Debug Port Capability ID	This register is hardwired to 0Ah, which indicates that this is the start of a Debug Port Capability structure.	RO

11.1.19 Offset 59h: Next Item Pointer #2

Table 397. Offset 59h: Next Item Pointer #2

Device:	29	**Function:**	7
Offset:	59h	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Next Item Pointer #2	This register is hardwired to 00h which indicates there are no more capability structures in this function.	RO

11.1.20 Offset 5Ah - 5Bh: Debug Port Base Offset

Table 398. Offset 5Ah - 5Bh: Debug Port Base Offset

Device:	29	**Function:**	7
Offset:	5Ah - 5Bh	**Attribute:**	Read-Only
Default Value:	2080h	**Size:**	16-bit

Bits	Name	Description	Access
15:13	BAR Number	This field is hardwired to 20h to indicate the memory BAR begins at offset 20h in the EHCI configuration space.	RO
12:0	Debug Port Offset	This field is hardwired to 080h to indicate that the Debug Port registers begin at offset 80h in the EHCI memory range.	RO

11.1.21 Offset 60h: Serial Bus Release Number

Table 399. Offset 60h: Serial Bus Release Number

Device:	29	**Function:**	7
Offset:	60h	**Attribute:**	Read-Only
Default Value:	20h	**Size:**	8-bit

Bits	Name	Description	Access
7:0		A value of 20h indicates that this controller follows the USB Specification rev. 2.0.	

11.1.22 Offset 61h: Frame Length Adjustment

Note: This feature is used to adjust any offset from the clock source that generates the clock that drives the SOF counter. When a new value is written into these six bits, the length of the frame is adjusted. Its initial programmed value is system dependent based on the accuracy of hardware USB clock and is initialized by system BIOS. This register should only be modified when the *HChalted* bit in the USBSTS register is a '1'. Changing value of this register while the host controller is operating yields undefined results. It

should not be reprogrammed by USB system software unless the default or BIOS programmed values are incorrect or the system is restoring the register while returning from a suspended state.

Table 400. Offset 61h: Frame Length Adjustment

Device:	29	***Function:***	7
Offset:	61h	***Attribute:***	Read/Write
Default Value:	20h	***Size:***	8-bit

Bits	Name	Description	Access
7:6	Reserved	These bits are reserved for future use and should read as 00b.	RO
5:0	Frame Length Timing Value	Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives a SOF cycle time of 60000. **Frame Length (# High Speed bit times) / FLADJ Value** (decimal) — (decimal) 59488 — 0 (00h) 59504 — 1 (01h) 59520 — 2 (02h) ... 59984 — 31 (1Fh) 60000 — 32 (20h) ... 60480 — 62 (3Eh) 60496 — 63 (3Fh)	

11.1.23 Offset 62 - 63h: Port Wake Capability

Note: This register is in the suspend power well. The intended use of this register is to establish a policy about which ports are to be used for wake events. Bit positions 1-4 in the mask correspond to a physical port implemented on the current EHCI controller. A '1' in a bit position indicates that a device connected below the port may be enabled as a wake-up device, and the port may be enabled for disconnect/connect or over-current events as wake-up events. This is an information-only mask register. The bits in this register **do not** affect the actual operation of the EHCI host controller. The system-specific policy may be established by BIOS initializing this register to a system-specific value. System software uses the information in this register when enabling devices and ports for remote wake-up.

Table 401. Offset 62 - 63h: Port Wake Capability

Device:	29	**Function:**	7
Offset:	62 - 63h	**Attribute:**	Read/Write
Default Value:	001Fh	**Size:**	16-bit

Bits	Name	Description	Access
15:5	Reserved	Reserved.	
4:1	Port Wake Up Capability Mask	Bit positions 1 through 4 correspond to a physical port implemented on this host controller. For example, bit position 1 corresponds to port 0, position 2 port 1, position 3 port 2, position 4 port 3.	RW
0	Port Wake Implemented	A '1' in bit 0 indicates that this register is implemented to software.	RW

11.1.24 Offset 64 - 65h: Classic USB Override

Note: This 16-bit register provides a bit corresponding to each of the ports on the EHCI host controller. When a bit is set to '1', the corresponding USB port is routed to the classic (UHCI) host controller and will operate using only the classic signaling rates. The feature is implemented with the following requirements:

- The associated Port Owner bit does *not* reflect the value in this new Override register. This guarantees compatibility with EHCI drivers.
- The associated Port Owner bit does *not* reflect the value in this new Override register. This guarantees compatibility with EHCI drivers.
- BIOS must only write to this register during initialization (while the Configured Flag is '0').
- The register is implemented in the Suspend well to maintain port-routing when the core power goes down
- When a '1' is present in the Override register, the classic controller operates the port regardless of the EHCI port routing logic. The corresponding EHCI port will always appear disconnected in this mode.

Note: EHCI test modes will not work on a port that has been overridden by this register.

Table 402. Offset 64-65h: CUO - Classic USB Override

Device:	29	**Function:**	7
Offset:	64-65h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
		Power Well:	Suspend

Bits	Name	Description	Reset Value	Access
15:4	Reserved	Reserved.		
3:2	Classic USB Port Owner	A “1” in a bit position forces the corresponding USB port to the classic host controller		R/W
0	Reserved	Reserved.		

11.1.25 Offset 68 - 6Bh: USB EHCI Legacy Support Extended Capability

Table 403. Offset 68 - 6Bh: USB EHCI Legacy Support Extended Capability

Device:	29	**Function:**	7
Offset:	68-6Bh	**Attribute:**	Read/Write
Default Value:	00000001h	**Size:**	32-bit
		Power Well:	Suspend

Bits	Name	Description	Reset Value	Access
31:25	Reserved	Reserved. Hardwired to 00h.		
24	HC OS Owned Semaphore	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.		R/W
23:17	Reserved	Reserved. Hardwired to 00h.		RO
16	HC BIOS Owned Semaphore	The BIOS sets this bit to establish ownership of the EHCI controller. System BIOS will clear this bit in response to a request for ownership of the EHCI controller by system software.		R/W
15:8	Next EHCI Capability Pointer	A value of 00h indicates that there are no EHCI Extended Capability structures in this device.		RO
7:0	Capability ID	A value of 01h indicates that this EHCI Extended Capability is the Legacy Support Capability.		RO

11.1.26 Offset 6C - 6Fh: USB EHCI Legacy Support Extended Control/Status

Table 404. Offset 6C - 6Fh: USB EHCI Legacy Support Extended Control/Status (Sheet 1 of 2)

Device:	29	**Function:**	7
Offset:	6C-6Fh	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
		Power Well:	Suspend

Bits	Name	Description	Access
31	SMI on BAR	This bit is set to '1' whenever the Base Address Register (BAR) is written.	R/WC
30	SMI on PCI Command	This bit is set to '1' whenever the PCI Command Register is written.	R/WC
29	SMI on OS Ownership Change	This bit is set to '1' whenever the HC OS Owned Semaphore bit in the USB EHCI Legacy Support Extended Capability register transitions from '1' to a '0' or '0' to a '1'.	R/WC
28:22	Reserved	Hardwired to 00h.	RO
21	SMI on Async Advance	Shadow bit of the Interrupt on Async Advance bit in the USB2STS register. To clear this bit system software must write a '1' to the Interrupt on Async Advance bit in the USB2STS register.	RO
20	SMI on Host System Error	Shadow bit of Host System Error bit in the USB2STS. To clear this bit, system software must write a '1' to the Host System Error bit in the USB2STS register.	RO
19	SMI on Frame List Rollover	Shadow bit of Frame List Rollover bit in the USB2STS register. To clear this bit system software must write a '1' to the Frame List Rollover bit in the USB2STS register.	RO
18	SMI on Port Change Detect	Shadow bit of Port Change Detect bit in the USB2STS register. To clear this bit system software must write a '1' to the Port Change Detect bit in the USB2STS register.	RO
17	SMI on USB Error	Shadow bit of USB Error Interrupt (USBERRINT) bit in the USB2STS register. To clear this bit system software must write a '1' to the USB Error Interrupt bit in the USB2STS register.	RO
16	SMI on USB Complete	Shadow bit of USB Interrupt (USBINT) bit in the USB2STS register. To clear this bit system software must write a '1' to the USB Interrupt bit in the USB2STS register.	RO
15	SMI on BAR Enable	When this bit is '1' and SMI on BAR is '1', the host controller will issue an SMI.	R/W
14	SMI on PCI Command Enable	When this bit is '1' and SMI on PCI Command is '1', the host controller will issue an SMI.	R/W
13	SMI on OS Ownership Enable	When this bit is a '1' AND the OS Ownership Change bit is '1', the host controller will issue an SMI.	R/W
12:6	Reserved	Reserved—RO. Hardwired to 00h.	RO
5	SMI on Async Advance Enable	When this bit is a '1' and the SMI on Async Advance bit is a '1', the host controller will issue an SMI immediately.	R/W

Table 404. Offset 6C - 6Fh: USB EHCI Legacy Support Extended Control/Status (Sheet 2 of 2)

Device:	29	**Function:**	7
Offset:	6C-6Fh	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
		Power Well:	Suspend

Bits	Name	Description	Access
4	SMI on Host System Error Enable	When this bit is a '1' and the SMI on Host System Error is a '1', the host controller will issue an SMI.	R/W
3	SMI on Frame List Rollover Enable	When this bit is a '1' and the SMI on Frame List Rollover bit is a '1', the host controller will issue an SMI.	R/W
2	SMI on Port Change Enable	When this bit is a '1' and the SMI on Port Change Detect bit is a '1', the host controller will issue an SMI.	R/W
1	SMI on USB Error Enable	When this bit is a '1' and the SMI on USB Error bit is a '1', the host controller will issue an SMI immediately.	R/W
0	SMI on USB Complete Enable	When this bit is a '1' and the SMI on USB Complete bit is a '1', the host controller will issue an SMI immediately.	R/W

11.1.27 Offset 70 - 73h: Intel Specific USB EHCI SMI

Note: This register provides a mechanism for BIOS to provide USB EHCI related bug fixes and workarounds. Writing a '1' to that bit location clears bits that are marked as Read/Write-Clear (R/WC). Software should clear all SMI status bits prior to setting the global SMI enable bit and individual SMI enable bit to prevent spurious SMI when returning from a powerdown.

Table 405. Offset 70 - 73h: Intel Specific USB EHCI SMI (Sheet 1 of 2)

Device:	29	**Function:**	7
Offset:	70-73h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
		Power Well:	Suspend

Bits	Name	Description	Access
31:28	Reserved	Reserved. Hardwired to 00h.	RO
27:26	Reserved	Reserved.	R/WC
25:22	SMI on PortOwner	Bits 27:22 correspond to the Port Owner bits for ports 0 (22) through 3 (25). These bits are set to '1' whenever the associated Port Owner bits transition from '0'->'1' or '1'->'0'. Software clears these bits by writing a '1'.	R/WC
21	SMI on PMCSR	This bit is set to '1' whenever software modifies the Power State bits in the Power Management Control/Status (PMCSR) register.	R/WC
20	SMI on Async	This bit is set to '1' whenever the Async Schedule Enable bit transitions from '1'->'0' or '0'->'1'	R/WC

Table 405. Offset 70 - 73h: Intel Specific USB EHCI SMI (Sheet 2 of 2)

Device:	29	**Function:**	7
Offset:	70-73h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
		Power Well:	Suspend

Bits	Name	Description	Access
19	SMI on Periodic	This bit is set to '1' whenever the Periodic Schedule Enable bit transitions from '1'->'0' or '0'->'1'	R/WC
18	SMI on CF	This bit is set to '1' whenever the Configure Flag (CF) transitions from '1'->'0' or '0'->'1'.	R/WC
17	SMI on HCHalted	This bit is set to '1' whenever HCHalted transitions to '1' as a result of the Run/Stop bit being cleared.	R/WC
16	SMI on HCReset	This bit is set to '1' whenever HCRESET transitions to '1'	R/WC
15:12	Reserved	Reserved. Hardwired to 00h.	RO
11:10	Reserved	Reserved.	R/W
9:6	SMI on PortOwner Enable	When any of these bits are '1' and the corresponding SMI on PortOwner bits are '1', the host controller will issue an SMI. Unused ports should have their corresponding bits cleared.	R/W
5	SMI on PMSCR Enable	When this bit is '1' and SMI on PMSCR is '1', the host controller will issue an SMI.	R/W
4	SMI on Async Enable	When this bit is '1' and SMI on Async is '1', the host controller will issue an SMI	R/W
3	SMI on Periodic Enable	When this bit is '1' and SMI on Periodic is '1', the host controller will issue an SMI.	R/W
2	SMI on CF Enable	When this bit is '1' and SMI on CF is '1', then the host controller will issue an SMI.	R/W
1	SMI on HCHalted Enable	When this bit is a '1' and SMI on HCHalted is '1', the host controller will issue an SMI.	R/W
0	SMI on HCReset Enable	When this bit is a '1' and SMI on HCReset is '1', host controller will issue an SMI.	R/W

11.1.28 Offset 80h: Access Control

Table 406. Offset 80h: Access Control

Device:	29	***Function:***	7
Offset:	80h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:1	Reserved	Reserved.	
0	WRT_RDONLY	When set to '1', this bit enables a select group of normally read-only registers in the EHC function to be written by software. Registers that may only be written when this mode is entered are noted in the summary tables and detailed description as Read/Write-Special. The registers fall into two categories: 1. System-configured parameters 2. Status bits	R/W

11.1.29 HS_ Ref_V_USB HS Reference Voltage Register

Table 407. HS_ Ref_V_USB HS Reference Voltage Register

Device:	29	***Function:***	7
Offset:	DCh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:22	Reserved	Reserved	RO
21:16	USB2 HS Ref Voltage	BIOS should always program this register to the recommended value of '111111b'. All other values are reserved.	RW
15:0	Reserved	Reserved	RW
NOTE: System BIOS should program a value of '111111b' into D29:F7:Register Offset DCh, bits 21:16 during POST and resume from S3(STR)/S4(STD) states on ALL steppings of the Intel 6300ESB ICH.			

11.2 Memory-Mapped I/O Registers

The USB 2.0 EHCI memory-mapped I/O space is composed of two sets of registers: Capability Registers and Operational Registers.

Note: When the USB EHCI function is in the D3 PCI power state, accesses to the USB EHCI memory range are ignored and result a master abort. Similarly, when the Memory Space Enable (MSE) bit is not set in the Command register in configuration space, the memory range will not be decoded by the Intel® 6300ESB ICH Enhanced Host Controller (EHC). When the MSE bit is not set, the Intel® 6300ESB ICH must default to allowing any memory accesses for the range specified in the BAR to go to PCI. This is

because the range may not be valid and, therefore, the cycle must be made available to any other targets that may be currently using that range.

11.2.1 Host Controller Capability Registers

These registers specify the limits, restrictions and capabilities of the host controller implementation. Within the Host Controller Capability Registers, only the Structural Parameters register is writable. This register is implemented in the Suspend well and is only reset by the standard suspend-well hardware reset, not by HCRESET or the D3-to-D0 reset.

Offset	Register	Default	Type
00h	Capabilities Registers Length	20h	RO
02-03h	Host Controller Interface Version Number	0100h	RO
04-07h	Structural Parameters	00103206h	RW-Special
08-0Bh	Capability Parameters	00006871h	RO

NOTE: "RW-Special" means that the register is normally read-only, but may be written when the WRT_RDONLY bit is set. Since these registers are expected to be programmed by BIOS during initialization, their contents must not get modified by HCRESET or D3-to-D0 internal reset.

11.2.1.1 Offset 00h: CAPLENGTH—Capability Registers Length

Table 408. Offset 00h: CAPLENGTH—Capability Registers Length

Device:	29	***Function:***	7
Offset:	00h	***Attribute:***	Read-Only
Default Value:	20h	***Size:***	8-bit

Bits	Name	Description	Access
7:0		This register is used as an offset to add to the Memory Base Register to find the beginning of the Operational Register Space. This is fixed at 20h, indicating that the Operation Registers begin at offset 20h.	

11.2.1.2 Offset 02 - 03h: HCIVERSION—Host Controller Interface Version Number

Table 409. Offset 02 - 03h: HCIVERSION—Host Controller Interface Version Number

Device:	29	***Function:***	7
Offset:	02 - 03h	***Attribute:***	Read-Only
Default Value:	0100h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Host Controller Interface Version Number	This is a two-byte register containing a BCD encoding of the version number of interface that this host controller interface conforms.	RO

11.2.1.3 Offset 04 - 07h: HCSPARAMS—Host Controller Structural Parameters

Note: This register is reset by a suspend well reset and not a D3-to-D0 reset or HCRESET.

Note: This register is writable when the WRT_RDONLY bit is set.

Table 410. Offset 04 - 07h: HCSPARAMS—Host Controller Structural Parameters (Sheet 1 of 2)

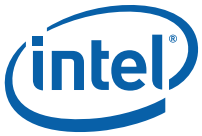
Device:	29	***Function:***	7
Offset:	04-07h	***Attribute:***	Read/Write-Special
Default Value:	00102204h	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved. Default = 0h.	RO
23:20	Debug Port Number (DP_N)	Hardwired to 1h, indicating that the Debug Port is on the lowest numbered port on the Intel® 6300ESB ICH.	
19:17	Reserved	Reserved.	
16	Reserved	Reserved. Hardwired to 0.	

Table 410. Offset 04 - 07h: HCSPARAMS—Host Controller Structural Parameters (Sheet 2 of 2)

Device:	29	**Function:**	7
Offset:	04-07h	**Attribute:**	Read/Write-Special
Default Value:	00102204h	**Size:**	32-bit

Bits	Name	Description	Access
15:12	Number of Companion Controllers (N_CC)	This field indicates the number of companion controllers associated with this USB ECHI host controller. A '0' in this field indicates there are no companion host controllers. Port-ownership hand-off is not supported. Only high-speed devices are supported on the host controller root ports. A value larger than '1' in this field indicates there are companion USB UHCI host controller(s). Port-ownership hand-offs are supported. High, Full- and Low-speed devices are supported on the host controller root ports. The Intel® 6300ESB ICH allows the default value of 2h to be over-written by BIOS. Since the Intel® 6300ESB ICH cannot support more than two companion host controllers, bits 15:14 are implemented as read-only 00b. When removing classic controllers, they should be disabled in the following order: Function 1 and Function 0, which correspond to ports 3:2 and 1:0, respectively.	
11:8	Number of Ports per Companion Controller (N_PCC)	This field indicates the number of ports supported per companion host controller. It is used to indicate the port routing configuration to system software. The Intel® 6300ESB ICH hardwires this field to 2h.	RO
7:4	Reserved	Reserved. These bits are reserved and default to '0'.	
3:0	N_PORTS	This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 1H to FH. The Intel® 6300ESB ICH reports 4h by default. However, software may write a value less than four for some platform configurations. A '0' in this field is undefined. Bit 3 is always hardwired to '0'.	



11.2.1.4 Offset 08 - 0Bh: HCCPARAMS—Host Controller Capability Parameters

Table 411. Offset 08 - 0Bh: HCCPARAMS—Host Controller Capability Parameters

Device:	29	***Function:***	7
Offset:	08-0Bh	***Attribute:***	Read-Only
Default Value:	00006871h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:8	EHCI Extended Capabilities Pointer (EECP)	This field is hardwired to 68h, indicating that the EHCI capabilities list exists and begins at offset 68h in the PCI configuration space.	RO
7:4	Isochronous Scheduling Threshold	This field indicates, relative to the current position of the executing host controller, where software may reliably update the isochronous schedule. When bit [7] is '0', the value of the least significant 3 bits indicates the number of micro-frames a host controller holds a set of isochronous data structures (one or more) before flushing the state. When bit [7] is a '1', then host software assumes the host controller may cache an isochronous data structure for an entire frame. Refer to the EHCI specification for details on how software uses this information for scheduling isochronous transfers. This field is hardwired to 7h.	RO
3	Reserved	Reserved. These bits are reserved and should be set to '0'.	
2	Asynchronous Schedule Park Capability	This bit is hardwired to 0, indicating that the Host Controller does not support this optional feature.	
1	Programmable Frame List Flag	0 = System software must use a frame list length of 1024 elements with this host controller. The USBCMD register *Frame List Size* field is a read-only register and must be set to '0'. 1 = System software may specify and use a smaller frame list and configure the host controller through the USBCMD register *Frame List Size* field. The frame list must always be aligned on a 4K page boundary. This requirement ensures that the frame list is always physically contiguous.	RO
0	64-bit Addressing Capability	This field documents the addressing range capability of this implementation. The value of this field determines whether software should use the 32-bit or 64-bit data structures. Values for this field have the following interpretation: 0 = Data structures using 32-bit address memory pointers 1 = Data structures using 64-bit address memory pointers This bit is hardwired to 1.	RO

11.2.2 Host Controller Operational Registers

This section defines the enhanced host controller operational registers. These registers are located after the capabilities registers. The operational register base must be DWORD-aligned and is calculated by adding the value in the first capabilities register (CAPLENGTH=20h) to the base address of the enhanced host controller register address space. All registers are 32 bits in length.

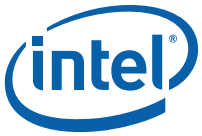
Offset	Register	Default	Special Notes	Type
00-03h	USB EHCI Command	00080000h		R/W
04-07h	USB EHCI Status	00001000h		R/W
08-0Bh	USB EHCI Interrupt Enable	00000000h		R/W
0C-0Fh	USB EHCI Frame Index	00000000h		R/W
10-13h	Control Data Structure Segment	00000000h		R/W
14-17h	Period Frame List Base Address	00000000h		R/W
18-1Bh	Next Asynchronous List Address	00000000h		R/W
1C-3Fh	Reserved	0h		RO
40- 43h	Configure Flag Register	00000000h	Suspend	R/W
44-47h	Port 0 Status and Control	00003000h	Suspend	R/W
48-4Bh	Port 1 Status and Control	00003000h	Suspend	R/W
4C-4Fh	Port 2 Status and Control	00003000h	Suspend	R/W
50-53h	Port 3 Status and Control	00003000h	Suspend	R/W
54-5Fh	Reserved	Undefined		RO
60-73h	Debug Port Registers.	Undefined		RO
74-3FFh	Reserved	Undefined		RO

Note: Software must read and write these registers using only DWORD accesses. These registers are divided into two sets. The first set at offsets 20h to 3Fh are implemented in the core power well. Unless otherwise noted, the core-well registers are reset by the assertion of any of the following:

- Core well hardware reset
- HCRESET
- D3-to-D0 reset

The second set at offsets 60h to the end of the implemented register space are implemented in the Suspend power well. Unless otherwise noted, the suspend-well registers are reset by the assertion of either of the following:

- Suspend well hardware reset
- HCRESET



11.2.2.1 Offset CAPLENGTH + 00 - 03h: USB EHCI CMD—USB EHCI Command Register

Table 412. Offset CAPLENGTH + 00 - 03h: USB EHCI CMD—USB EHCI Command Register (Sheet 1 of 3)

Device:	29	***Function:***	7
Offset:	CAPLENGTH + 00-03h	***Attribute:***	Read/Write
Default Value:	00080000h	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved. These bits are reserved and should be set to '0'.	
23:16	Interrupt Threshold Control	Default 04h. This field is used by system software to select the maximum rate at which the host controller will issue interrupts. The only valid values are defined below. When software writes an invalid value to this register, the results are undefined. **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)	R/W
15:8	Reserved	Reserved. These bits are reserved and should be set to '0'.	
11:8	Unimplemented Asynchronous Park Mode Bits	This field is hardwired to 000b because the host controller does not support this optional feature.	
7	Light Host Controller Reset	The Intel® 6300ESB ICH does not implement this optional reset and hardwires this bit to 0.	RO
6	Interrupt on Async Advance Doorbell	This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Software must write a '1' to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS register. When the Interrupt on Async Advance Enable bit in the USBINTR register is a '1', the host controller will assert an interrupt at the next interrupt threshold. See the EHCI specification for operational details. The host controller sets this bit to a '0' after it has set the Interrupt on Async Advance status bit in the USBSTS register to a '1'. Software should not write a '1 'to this bit when the asynchronous schedule is inactive. Doing so will yield undefined results.	R/W
5	Asynchronous Schedule Enable	Default 0b. This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean: 0 = Do not process the Asynchronous Schedule 1 = Use the ASYNCLISTADDR register to access the Asynchronous Schedule.	R/W

Table 412. Offset CAPLENGTH + 00 - 03h: USB EHCI CMD—USB EHCI Command Register (Sheet 2 of 3)

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 00-03h	**Attribute:**	Read/Write
Default Value:	00080000h	**Size:**	32-bit

Bits	Name	Description	Access
4	Periodic Schedule Enable	Default 0b. This bit controls whether the host controller skips processing the Periodic Schedule. Values mean: 0 = Do not process the Periodic Schedule 1 = Use the PERIODICLISTBASE register to access the Periodic Schedule.	R/W
3:2	Frame List Size	The Intel® 6300ESB ICH hardwires this field to 00b because it only supports the 1024-element frame list size.	RO

Table 412. Offset CAPLENGTH + 00 - 03h: USB EHCI CMD—USB EHCI Command Register (Sheet 3 of 3)

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 00-03h	**Attribute:**	Read/Write
Default Value:	00080000h	**Size:**	32-bit

Bits	Name	Description	Access
1	Host Controller Reset (HCRESET)	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 (i.e., RSMRST# assertion and PWROK deassertion on the Intel® 6300ESB ICH). When software writes a '1' 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. **NOTE:** *PCI Configuration registers and Host Controller Capability Registers are not effected 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), with the side effects described in the EHCI spec. Software must re-initialize the host controller in order to return the host controller to an operational state. This bit is set to '0' by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a '0' to this register. Software should not set this bit to a '1' when the HCHalted bit in the USBSTS register is a '0'. Attempting to reset an actively running host controller will result in undefined behavior. This reset can be used to leave EHCI port test modes.	R/W
0	Run/Stop (RS)	Default 0b. 1=Run. 0=Stop. When set to a 1, the Host Controller proceeds with execution of the schedule. The Host Controller continues execution as long as this bit is set. When this bit is set to 0, the Host Controller completes the current transaction on the USB and then halts. The HC Halted bit in the status register indicates when the Host Controller has finished the transaction and has entered the stopped state. Software should not write a '1' to this field unless the host controller is in the Halted state (i.e., HCHalted in the USBSTS register is a '1'). The Halted bit is cleared immediately when the Run bit is set. The following table explains how the different combinations of Run and Halted should be interpreted: Run/Stop — Halted — Interpretation 0 — 0 — Valid - in the process of halting 0 — 1 — Valid - halted 1 — 0 — Valid - running 1 — 1 — Invalid - the HCHalted bit clears immediately. Memory read cycles initiated by the EHC that receive any status other than Successful will result in this bit being cleared.	R/W

NOTE: The Command Register indicates the command to be executed by the serial bus host controller. Writing to the register causes a command to be executed.

11.2.2.2 Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status

Note: This register indicates pending interrupts and various states of the Host Controller. The status resulting from a transaction on the serial bus is not indicated in this register. Software sets a bit to '0' in this register by writing a '1' to it. See the Interrupts description in Section 4 of the EHCI Specification for additional information concerning USB EHCI interrupt conditions.

Table 413. Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status (Sheet 1 of 2)

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 04-07h	**Attribute:**	Read/Write Clear
Default Value:	00001000h	**Size:**	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved. These bits are reserved and should be set to '0'.	
15	Asynchronous Schedule Status	0 = Default. This bit reports the current real status of the Asynchronous Schedule. When this bit is a '0', the status of the Asynchronous Schedule is disabled. When this bit is a '1', the status of the Asynchronous Schedule is enabled. The Host Controller is not required to *immediately* disable or enable the Asynchronous Schedule when software transitions the *Asynchronous Schedule Enable* bit in the USBCMD register. When this bit and the *Asynchronous Schedule Enable* bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).	RO
14	Periodic Schedule Status	0 = Default. This bit reports the current real status of the Periodic Schedule. When this bit is a '0', the status of the Periodic Schedule is disabled. When this bit is a '1', the status of the Periodic Schedule is enabled. The Host Controller is not required to *immediately* disable or enable the Periodic Schedule when software transitions the *Periodic Schedule Enable* bit in the USBCMD register. When this bit and the *Periodic Schedule Enable* bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).	RO
13	Reclamation	0 = Default. This is a read-only status bit used to detect an empty asynchronous schedule. The operational model and valid transitions for this bit are described in Section 4 of the EHCI Specification.	RO
12	HCHalted	(Defaults to 1). This bit is a '0' whenever the Run/Stop bit is a '1'. The Host Controller sets this bit to '1' after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller hardware (e.g., internal error).	RO
11:6	Reserved	Reserved.	
5	Interrupt on Async Advance	0 = Default. System software may force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a '1' to the *Interrupt on Async Advance Doorbell* bit in the USBCMD register. This bit indicates the assertion of that interrupt source.	R/WC

Table 413. Offset CAPLENGTH + 04 - 07h: USB EHCI STS—USB EHCI Status (Sheet 2 of 2)

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 04-07h	**Attribute:**	Read/Write Clear
Default Value:	00001000h	**Size:**	32-bit

Bits	Name	Description	Access
4	Host System Error	The Host Controller sets this bit to '1' when a serious error occurs during a host system access involving the Host Controller module. A hardware interrupt is generated to the system. Memory read cycles initiated by the EHC that receive any status other than Successful will result in this bit being set. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs. A hardware interrupt is generated to the system (when enabled in the Interrupt Enable Register).	R/WC
3	Frame List Rollover	The Host Controller sets this bit to a '1' when the *Frame List Index* rolls over from its maximum value to '0'. Since the Intel® 6300ESB ICH only supports the 1024-entry Frame List Size, the *Frame List Index* rolls over every time FRNUM[13] toggles.	R/WC
2	Port Change Detect	The Host Controller sets this bit to a '1' when any port for which the *Port Owner* bit is set to '0' has a change bit transition from a '0' to a '1' or a Force Port Resume bit transition from a '0' to a '1' as a result of a J-K transition detected on a suspended port. This bit is allowed to be maintained in the Auxiliary power well. Alternatively, it is also acceptable that on a D3 to D0 transition of the EHCI HC device, this bit is loaded with the OR of all of the PORTSC change bits (including: Force port resume, over-current change, enable/disable change and connect status change). Regardless of the implementation, whenever this bit is readable (i.e., in the D0 state), it must provide a valid view of the Port Status registers.	R/WC
1	USB Error Interrupt (USBERRINT)	The Host Controller sets this bit to '1' when completion of a USB transaction results in an error condition (e.g., error counter underflow). When the TD on which the error interrupt occurred also had its IOC bit set, both this bit and Bit '0' are set. See the EHCI specification for a list of the USB errors that will result in this interrupt being asserted.	R/WC
0	USB Interrupt (USBINT)	The Host Controller sets this bit to '1' when the cause of an interrupt is a completion of a USB transaction whose Transfer Descriptor had its IOC bit set. The Host Controller also sets this bit to '1' when a short packet is detected (actual number of bytes received was less than the expected number of bytes).	R/WC

11.2.2.3 Offset CAPLENGTH + 08 - 0Bh: USB EHCI INTR—USB EHCI Interrupt Enable

Table 414. Offset CAPLENGTH + 08 - 0Bh: USB EHCI INTR—USB EHCI Interrupt Enable

Device:	29	***Function:***	7
Offset:	CAPLENGTH + 08-0Bh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:6	Reserved	Reserved.These bits are reserved and should be '0'.	
5	Interrupt on Async Advance Enable	When this bit is a '1' and the Interrupt on Async Advance bit in the USBSTS register is a '1', the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.	R/W
4	Host System Error Enable	When this bit is a '1' and the Host System Error Status bit in the USBSTS register is a '1', the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.	R/W
3	Frame List Rollover Enable	When this bit is a '1' and the Frame List Rollover bit in the USBSTS register is a '1', the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit.	R/W
2	Port Change Interrupt Enable	When this bit is a '1' and the Port Change Detect bit in the USBSTS register is a '1', the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Change Detect bit.	R/W
1	USB Error Interrupt Enable	When this bit is a '1' and the USBERRINT bit in the USBSTS register is a '1', the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software by clearing the USBERRINT bit in the USBSTS register.	R/W
0	USB Interrupt Enable	When this bit is a '1' and the USBINT bit in the USBSTS register is a '1', the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software by clearing the USBINT bit in the USBSTS register.	R/W

NOTES:
1. For all enable register bits, 1= Enabled, 0= Disabled.
2. This register enables and disables reporting of the corresponding interrupt to the software. When a bit is set and the corresponding interrupt is active, an interrupt is generated to the host. Interrupt sources that are disabled in this register still appear in the Status Register to allow the software to poll for events. Each interrupt enable bit description indicates whether it is dependent on the interrupt threshold mechanism (see Section 4 of the EHCI Specification), or not.

11.2.2.4 Offset CAPLENGTH + 0C - 0Fh: FRINDEX—Frame Index

Note: This register is used by the host controller to index into the periodic frame list. The register updates every 125 microseconds (once each micro-frame). Bits [12:3] are used to select a particular entry in the Periodic Frame List during periodic schedule execution. The number of bits used for the index is fixed at 10 for the Intel® 6300ESB ICH since it only supports 1024-entry frame lists. This register must be written as a

DWORD. Word and byte writes produce undefined results. This register cannot be written unless the Host Controller is in the Halted state as indicated by the *HCHalted* bit (USB EHCI STS register). A write to this register while the Run/Stop bit is set to a '1' (USB EHCI CMD register) produces undefined results. Writes to this register also effect the SOF value. See Section 4 of the EHCI Specification for details.

The SOF frame number value for the bus SOF token is derived or alternatively managed from this register. Please refer to Section 4 of the EHCI Specification for a detailed explanation of the SOF value management requirements on the host controller. The value of FRINDEX must be within 125 µs (1 micro-frame) ahead of the SOF token value. The SOF value may be implemented as an 11-bit shadow register. For this discussion, this shadow register is 11 bits and is named SOFV. SOFV updates every 8 micro-frames. (1 millisecond). An example implementation to achieve this behavior is to increment SOFV each time the FRINDEX[2:0] increments from a '0' to a '1'.

Software must use the value of FRINDEX to derive the current micro-frame number, both for high-speed isochronous scheduling purposes and to provide the **get** micro-frame number function required to client drivers. Therefore, the value of FRINDEX and the value of SOFV must be kept consistent when the chip is reset or software writes to FRINDEX. Writes to FRINDEX must also **write-through** FRINDEX[13:3] to SOFV[10:0]. To keep the update as simple as possible, software should never write a FRINDEX value where the three least significant bits are 111b or 000b.

Table 415. Offset CAPLENGTH + 0C - 0Fh: FRINDEX—Frame Index

Device:	29	***Function:***	7
Offset:	CAPLENGTH + 0C-0Fh	***Attribute:***	*Read/Write*
Default Value:	00000000h	***Size:***	*32-bit*

Bits	Name	Description	Access
31:14	Reserved	Reserved.	
13:0	Frame List Current Index/Frame Number	The value in this register increments at the end of each time frame (e.g., micro-frame). Bits [12:3] are used for the Frame List current index. This means that each location of the frame list is accessed eight times (frames or micro-frames) before moving to the next index.	

11.2.2.5 Offset CAPLENGTH + 10 - 13h: CTRLDSSEGMENT—Control Data Structure Segment Register

Note: This 32-bit register corresponds to the most significant address bits [63:32] for all EHCI data structures. Since the Intel® 6300ESB ICH hardwires the 64-bit Addressing Capability field in HCCPARAMS to '1', then this register is used with the link pointers to construct 64-bit addresses to EHCI control data structures. This register is concatenated with the link pointer from either the PERIODICLISTBASE, ASYNCLISTADDR, or any control data structure link field to construct a 64-bit address. This register allows the host software to locate all control data structures within the same 4 Gbyte memory segment.

Table 416. Offset CAPLENGTH + 10 - 13h: CTRLDSSEGMENT—Control Data Structure Segment Register

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 10-13h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:0	Upper Address[63:32]	This 32-bit field corresponds to address bits 63:32 when forming a control data structure address.	RW

11.2.2.6 Offset CAPLENGTH + 14 - 17h: PERIODICLISTBASE—Periodic Frame List Base Address

Note: This 32-bit register contains the beginning address of the Periodic Frame List in the system memory. Since the Intel® 6300ESB ICH host controller operates in 64-bit mode (as indicated by the '1' in the 64-bit Addressing Capability field in the HCCSPARAMS register), then the most significant 32 bits of every control data structure address comes from the CTRLDSSEGMENT register. HCD loads this register prior to starting the schedule execution by the Host Controller. The memory structure referenced by this physical memory pointer is assumed to be 4-Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence.

Table 417. Offset CAPLENGTH + 14 - 17h: PERIODICLISTBASE—Periodic Frame List Base Address

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 14-17h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:12	Base Address (Low)	These bits correspond to memory address signals [31:12], respectively.	RW
11:0	Reserved	Reserved. Must be written as '0's. During runtime, the value of these bits are undefined.	

11.2.2.7 Offset CAPLENGTH + 18 - 1Bh: ASYNCLISTADDR—Current Asynchronous List Address

Note: This 32-bit register contains the address of the next asynchronous queue head to be executed. Since the Intel® 6300ESB ICH host controller operates in 64-bit mode (as indicated by a '1' in 64-bit Addressing Capability field in the HCCPARAMS register), then the most significant 32 bits of every control data structure address comes from the CTRLDSSEGMENT register. Bits [4:0] of this register cannot be modified by system software and will always return '0's when read. The memory structure referenced by this physical memory pointer is assumed to be 32-byte aligned.

Table 418. Offset CAPLENGTH + 18 - 1Bh: ASYNCLISTADDR—Current Asynchronous List Address

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 18-1Bh	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:5	Link Pointer Low (LPL)	These bits correspond to memory address signals [31:5], respectively. This field may only reference a Queue Head (QH).	RW
4:0	Reserved	Reserved. These bits are reserved and their value has no effect on operation.	

11.2.2.8 Offset CAPLENGTH + 40 - 43h: CONFIGFLAG—Configure Flag Register

Table 419. Offset CAPLENGTH + 40 - 43h: CONFIGFLAG—Configure Flag Register

Device:	29	**Function:**	7
Offset:	CAPLENGTH + 40-43h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:1	Reserved	Reserved. Read from this field will always return 0.	
0	Configure Flag (CF)	Default 0b. 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. See Section 4 of the EHCI Specification for operational details. 0 = Port routing control logic default–routes each port to the classic host controllers. 1 = Port routing control logic default–routes all ports to this host controller.	RW

11.2.2.9 PORTSC- Port N Status and Control

Note: A host controller must implement one or more port registers. Software uses the N_Port information from the Structural Parameters Register to determine how many ports need to be serviced. All ports have the structure defined below. Software must not write to unreported Port Status and Control Registers.

This register is in the suspend power well. It is only reset by hardware when the suspend power is initially applied or in response to a host controller reset. The initial conditions of a port are:

1. No device connected.
2. Port disabled. When a device is attached, the port state transitions to the attached state and system software will process this as with any status change notification. Refer to Section 4 of the EHCI Specification for operational requirements for how change events interact with port suspend mode
3. When a port is being used as the Debug Port, the port may report device connected and enabled when the Configured Flag is a ’0’.

Table 420. PORTSC- Port N Status and Control (Sheet 1 of 4)

Device:	29	***Function:***	7
Offset:	Port 0:CAPLENGTH+44-47h Port 1: CAPLENGTH+48-4Bh Port 2: CAPLENGTH+4C-4Fh Port 3: CAPLENGTH+50-53h	***Attribute:***	Read/Write
Default Value:	00003000h	***Size:***	32-bit

Bits	Name	Description	Access
31:23	Reserved	Reserved. These bits are reserved for future use and will return a value of '0' when read.	
22	Wake on Over-current Enable (WKOC_E)	Default = 0b. Writing this bit to a '1' enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the Over-current Active bit (bit 4 of this register) is set.	R/W
21	Wake on Disconnect Enable (WKDSCNNT_E)	Default = 0b. Writing this bit to a '1' enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the Current Connect Status changes from connected to disconnected (i.e., bit '0' of this register changes from '1' to '0').	R/W
20	Wake on Connect Enable (WKCNNT_E)	Default = 0b. Writing this bit to a '1' enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the Current Connect Status changes from disconnected to connected (i.e., bit '0' of this register changes from '0' to '1'). **NOTE:** This feature is not supported.	R/W
19:16	Port Test Control	Default = 0000b. When this field is '0', the port is NOT operating in a test mode. A non-'0' 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 is (0110b - 1111b are reserved): **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 Refer to Chapter 7 of the USB Specification, Revision 2.0, for details on each test mode.	R/W
15:14	Reserved	Should be written to =00b; other values will result in unspecified behavior.	R/W
13	Port Owner	Default = 1b. This bit unconditionally goes to a '0' when the Configured Flag bit makes a '0' to '1' transition.This bit unconditionally goes to 1b whenever the *Configure Flag* bit is '0' 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 '1' to this bit when the attached device is not a high-speed device. A '1' in this bit means that a companion host controller owns and controls the port. See Section 4 of the EHCI Specification for operational details.	R/W
12	Port Power (PP)	Read-only with a value of '1'. This indicates that the port does have power.	RO

Table 420. PORTSC- Port N Status and Control (Sheet 2 of 4)

Device:	29	**Function:**	7
Offset:	Port 0:CAPLENGTH+44-47h Port 1: CAPLENGTH+48-4Bh Port 2: CAPLENGTH+4C-4Fh Port 3: CAPLENGTH+50-53h	**Attribute:**	Read/Write
Default Value:	00003000h	**Size:**	32-bit

Bits	Name	Description	Access
11:10	Line Status	These bits reflect the current logical levels of the D+ (bit 11) and D- (bit 10) signal lines. These bits are used for detection of low-speed USB devices prior to the port reset and enable sequence. This field is valid only when the port enable bit is '0' and the current connect status bit is set to a '1'. The encoding of the bits are: **Bits[11:10] Meaning** 00 SE0 - Not Low-speed device, perform EHCI reset 10 J-state - Not Low-speed device, perform EHCI reset 01 K-state - Low speed device, release ownership of port 11 Undefined - Not Low-speed device, perform EHCI reset	RO
9	Reserved	Reserved. This bit will return a '0' when read.	
8	Port Reset	Default = 0 1 = Port is in Reset. 0 = Port is not in Reset. When software writes a '1' to this bit from a '0', the bus reset sequence as defined in the USB Specification Revision 2.0 is started. Software writes a '0' to this bit to terminate the bus reset sequence. Software must keep this bit at a '1' 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 '1', it must also write a '0' to the Port Enable bit. When software writes a '0' to this bit there may be a delay before the bit status changes to a '0'. The bit status will not read as a '0' until after the reset has completed. When the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (e.g., set the Port Enable bit to a '1'). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from a '1' to a '0'. For example: If the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state within 2ms of software writing this bit to a '0'. The HCHalted bit in the USBSTS register should be a '0' before software attempts to use this bit. The host controller may hold Port Reset asserted to a '1' when the HCHalted bit is a '1'. This field is '0' if Port Power is '0'. **Warning:** System software should not attempt to reset a port when the HCHalted bit in the USBSTS register is a '1'. Doing so will result in undefined behavior.	R/W

Table 420. PORTSC- Port N Status and Control (Sheet 3 of 4)

Device:	29	***Function:***	7
Offset:	Port 0:CAPLENGTH+44-47h Port 1: CAPLENGTH+48-4Bh Port 2: CAPLENGTH+4C-4Fh Port 3: CAPLENGTH+50-53h	***Attribute:***	Read/Write
Default Value:	00003000h	***Size:***	32-bit

Bits	Name	Description	Access
7	Suspend	1 = Port in suspend state. 0 = Port not in suspend state. Default = 0. Port Enabled Bit and Suspend bit of this register define the port states as follows: **Bits [Port Enabled, Suspend] Port State** 0X — Disable 10 — Enable 11 — Suspend When in suspend state, downstream propagation of data is blocked on this port, except for port reset. Note that the bit status does not change until the port is suspended and that there may be a delay in suspending a port depending on the activity on the port. The host controller will unconditionally set this bit to a '0' when software sets the *Force Port Resume* bit to a '0' (from a '1'). A write of '0' to this bit is ignored by the host controller. When host software sets this bit to a '1' when the port is not enabled (i.e., Port enabled bit is a '0'), the results are undefined.	R/W
6	Force Port Resume	1 = Resume detected/driven on port. 0 = No resume (K-state) detected/driven on port. Default = 0. Software sets this bit to a '1' to drive resume signaling. The Host Controller sets this bit to a '1' when a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a '1' because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register is also set to a '1'. When software sets this bit to a '1', the host controller must not set the Port Change Detect bit. Note that when the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB Specification Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this bit remains a '1'. Software must appropriately time the Resume and set this bit to a '0' when the appropriate amount of time has elapsed. Writing a '0' (from '1') causes the port to return to high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a '1' until the port has switched to the high-speed idle.	R/W
5	Overcurrent Change	Default = 0. 1 = This bit gets set to a '1' when there is a change to Over-current Active. Software clears this bit by writing a '1' to this bit position. The functionality of this bit is not dependent upon the port owner.	R/WC

Table 420. PORTSC- Port N Status and Control (Sheet 4 of 4)

Device:	29	**Function:**	7
Offset:	Port 0:CAPLENGTH+44-47h Port 1: CAPLENGTH+48-4Bh Port 2: CAPLENGTH+4C-4Fh Port 3: CAPLENGTH+50-53h	**Attribute:**	Read/Write
Default Value:	00003000h	**Size:**	32-bit

Bits	Name	Description	Access
4	Overcurrent Active	Default = 0. 1 = This port currently has an over-current condition. 0 = This port does not have an over-current condition. This bit will automatically transition from a '1' to a '0' when the over current condition is removed. The Intel® 6300ESB ICH automatically disables the port when the over-current active bit is '1'.	RO
3	Port Enable/Disable Change	0 = No change in status. This is the default setting. 1 = Port enabled/disabled status has changed. For the root hub, this bit gets set to a '1' 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.) This bit is not set due to the Disabled-to-Enabled transition, nor due to a disconnect. Software clears this bit by writing a '1' to it.	R/WC
2	Port Enabled/Disabled	1 = Enable. 0 = Disable. Default = 0. Ports may only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a '1' to this field. Ports may 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.	R/W
1	Connect Status Change	1 = Change in Current Connect Status. 0 = No change. Default = 0. Indicates a change has occurred in the port's Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even when system software has not cleared an existing connect status change. For example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be "setting" an already-set bit (i.e., the bit will remain set). Software sets this bit to '0' by writing a '1' to it.	R/WC
0	Current Connect Status	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 (Bit 1) to be set.	RO

11.2.3 USB 2.0-Based Debug Port Register

The Debug Port's registers are located in the same memory area, defined by the Base Address Register (BAR) as the standard EHCI registers. The base offset for the debug port registers is declared in the Debug Port Base Offset Capability Register at Configuration offset 5Ah. The specific EHCI port that supports this debug capability is indicated by a 4-bit field (bits 20-23) in the HCSPARAMS register of the EHCI controller.

The map of the Debug Port registers is shown below. Each register is defined individually.

Offset	Register	Type
00h	Control/Status Register	Read/Write
04h	USB PIDs	Read/Write
08h	Data Buffer (Bytes 3:0)	Read/Write
0Ch	Data Buffer (Bytes 7:4)	Read/Write
10h	Config Register	Read/Write

NOTES:
1. All of these registers are implemented in the core well and reset by PXPCIRST#, EHC HCRESET, and a EHC D3-to-D0 transition.
2. The hardware associated with this register provides no checks to ensure that software programs the interface correctly. How the hardware behaves when programmed illegally is undefined.

11.2.3.1 Offset 00h: Control/Status Register

Table 421. Offset 00h: Control/Status Register (Sheet 1 of 3)

Device:	29	***Function:***	7
Offset:	00h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	32-bit

Bits	Name	Description	Access
31	Reserved.	Reserved.	
30	OWNER_CNT	When software writes a '1' to this bit, the ownership of the debug port is forced to the EHCI controller (i.e., immediately taken away from the companion Classic USB Host Controller). When the port was already owned by the EHCI controller, setting this bit has no effect. This bit overrides all of the ownership-related bits in the standard EHCI registers. Reset default is '0'.	R/W
29	Reserved	Reserved.	
28	ENABLED_CNT	This bit = '1' when the debug port is enabled for operation. Software may clear this by writing a '0' to it. The hardware clears the bit for the same conditions where the Port Enable/ Disable Change bit (in the PORTSC register) is set. Software may directly set this bit when the port is already enabled in the associated Port Status and Control register (this is enforced by the hardware). Reset default is '0'.	
27:17	Reserved	Reserved.	

Table 421. Offset 00h: Control/Status Register (Sheet 2 of 3)

Device:	29	**Function:**	7
Offset:	00h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	32-bit

Bits	Name	Description	Access
16	DONE_STS	This bit is set by hardware to indicate that the request is complete. Writing a '1' to this bit will clear it when it is set. Writing a '0' to this bit has no effect. Reset default is '0'.	R/WC
15:12	LINK_ID_STS	This field identifies the link interface. It is hardwired to 0h to indicate that it is a USB Debug Port.	RO
11	Reserved	Reserved. This bit will return '0' when read. Writes will have no effect.	
10	IN_USE_CNT	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 is cleared after reset. (This bit has no affect on hardware.)	
9:7	EXCEPTION_STS	Default=000b This field indicates the exception when the ERROR_GOOD#_STS bit is set. This field should be ignored when the ERROR_GOOD#_STS bit is 0. 000 **No Error.** **Note:** This should not be seen, since this field should only be checked when there is an error. 001 **Transaction error:** Indicates the USB EHCI transaction had an error (CRC, bad PID, timeout, etc.) 010 **Hardware error:** Request was attempted (or in progress) when port was suspended or reset. All Others are reserved.	RO
6	ERROR_GOOD#_STS	The hardware clears this bit to '0' upon the proper completion of a read or write. The hardware sets this bit to indicate that an error has occurred. Details on the nature of the error are provided in the Exception field. Reset default = 0.	RO
5	GO_CNT	0 = Hardware clears this bit when hardware sets the DONE_STS bit. Default is 0. 1 = Causes hardware to perform a read or write request. Writing a '1' to this bit when it is already set may result in undefined behavior.	WO

Table 421. Offset 00h: Control/Status Register (Sheet 3 of 3)

Device:	29	**Function:**	7
Offset:	00h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	32-bit

Bits	Name	Description	Access
4	WRITE_READ#_CNT:	Software sets this bit to indicate that the current request is a write. Software clears this bit to indicate that the current request is a read. Reset default is '0'.	R/W
3:0	DATA_LEN_CNT	This field is used to indicate the size of the data to be transferred. Reset default = 0h. 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. 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 when 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 a read operation. A value of 0h indicates that a zero-length packet was returned and 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.	R/W

NOTES:
1. Software should do Read-Modify-Write operations to this register to preserve the contents of bits not being modified. This include Reserved bits.
2. To preserve the usage of Reserved bits in the future, software should always write the same value read from the bit until it is defined. Reserved bits will always return '0' when read.

11.2.3.2 Offset 04h: USB PIDs Register

Table 422. Offset 04h: USB PIDs Register

Device:	29	***Function:***	7
Offset:	04h	***Attribute:***	Read/WRite
Default Value:	TBD	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved. These bits will return '0' when read. Writes will have no effect.	
23:16	RECEIVED_PID_STS[23:16]	The hardware updates this field with the received PID for transactions in either direction. When the controller is writing data, this field is updated with the handshake PID that is received from the device. When the host controller is reading data, this field is updated with the data packet PID (when the device sent data), or the handshake PID (when the device NAKs the request). This field is valid when the hardware clears the GO_DONE#_CNT bit.	
15:8	SEND_PID_CNT[15:8]	The hardware sends this PID to begin the data packet when sending data to USB (i.e., WRITE_READ#_CNT is asserted). Software will typically set this field to either DATA0 or DATA1 PID values.	
7:0	TOKEN_PID_CNT[7:0]:	The hardware sends this PID as the Token PID for each USB transaction. Software will typically set this field to either IN, OUT or SETUP PID values.	

NOTE: This DWORD register is used to communicate PID information between the USB debug driver and the USB debug port. The debug port uses some of these fields to generate USB packets, and uses other fields to return PID information to the USB debug driver.

11.2.3.3 Offset 08h: Data Buffer Bytes 7:0

Table 423. Offset 08h: Data Buffer Bytes 7:0

Device:	29	***Function:***	7
Offset:	08h	***Attribute:***	Read/Write
Default Value:	0000000000000000h	***Size:***	64-bit

Bits	Name	Description	Access
63:0	DATABUFFER[63:0	These are the 8 bytes of the data buffer. Bits 7:0 correspond to least significant byte (byte 0). Bits 63:56 correspond to the most significant byte (byte 7). The bytes in the Data Buffer must be written with data before software initiates a write request. For a read request, the Data Buffer contains valid data when DONE_STS bit is cleared by the hardware, ERROR_GOOD#_STS is cleared by the hardware, and the DATA_LENGTH_CNT field indicates the number of bytes that are valid.	

NOTE: This register may be accessed as eight separate 8-bit registers or two separate 32-bit registers.

11.2.3.4 Offset 10h: Config Register

Table 424. Offset 10h: Config Register

Device:	29	***Function:***	7
Offset:	10h	***Attribute:***	Read/Write
Default Value:	TBD	***Size:***	32-bit

Bits	Name	Description	Access
31:15	Reserved	Reserved.	
14:8	USB_ADDRESS_CNF	This 7-bit field identifies the USB device address used by the controller for all Token PID generation. This is a R/W field that is set to 7Fh after reset.	
7:4	Reserved	Reserved.	
3:0	USB_ENDPOINT_CNF	This 4-bit field identifies the endpoint used by the controller for all Token PID generation. This is a R/W field that is set to 01h after reset.	

SMBUS Controller Registers (D31:F3) 12

12.1 PCI Configuration Registers (SMBUS—D31:F3)

Offset	Mnemonic	Register Name/ Function	Default	Type
00-01h	VID	Vendor ID	8086	RO
02-03h	DID	Device ID	25A4	RO
04-05h	CMD	Command Register	0000h	R/W
06-07h	STA	Device Status	0280h	R/W
08h	RID	Revision ID	See Note 1	RO
09h	PI	Programming Interface	00h	RO
0Ah	SCC	Sub Class Code	05h	RO
0Bh	BCC	Base Class Code	0Ch	RO
20-23h	SMB_BASE	SMBus Base Address Register	00000001h	R/W
2C-2Dh	SVID	Subsystem Vendor ID	0000h	RO
2E-2Fh	SID	Subsystem ID	0000h	RO
3Ch	INTR_LN	Interrupt Line	00h	R/W
3Dh	INTR_PN	Interrupt Pin	02h	RO
40h	HOSTC	Host Configuration	00h	R/W

NOTES:
1. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the value of the Revision ID Register.
2. Registers that are not shown should be treated as Reserved (See Section 6.2, “PCI Configuration Map” on page 277 for details).

12.1.1 Offset 00 - 01h: VID—Vendor Identification Register (SMBUS—D31:F3)

Table 425. Offset 00 - 01h: VID—Vendor Identification Register (SMBUS—D31:F3)

Device:	31	***Function:***	3
Offset:	00-01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel.	RO

12.1.2 Offset 02 - 03h: DID—Device Identification Register (SMBUS—D31:F3)

Table 426. Offset 02 - 03h: DID—Device Identification Register (SMBUS—D31:F3)

Device:	31	***Function:***	3
Offset:	02-03h	***Attribute:***	Read-Only
Default Value:	25A4h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Device ID value		RO

12.1.3 Offset 04 - 05h: CMD—Command Register (SMBUS—D31:F3)

Table 427. Offset 04 - 05h: CMD—Command Register (SMBUS—D31:F3)

Device:	31	***Function:***	3
Offset:	04-05h	***Attribute:***	Read-Only, Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:10	Reserved	Reserved.	
9	Fast Back-to-Back Enable (FBE)	Reserved as ‘0’.	RO
8	SERR# Enable (SERREN)	Reserved as ‘0’.	RO
7	Wait Cycle Control (WCC)	Reserved as ‘0’.	RO
6	Parity Error Response (PER)	Reserved as ‘0’.	RO
5	VGA Palette Snoop (VPS)	Reserved as ‘0’.	RO
4	Postable Memory Write Enable (PMWE)	Reserved as ‘0’.	RO
3	Special Cycle Enable (SCE)	Reserved as ‘0’.	RO
2	Bus Master Enable (BME)	Reserved as ‘0’.	RO
1	Memory Space Enable (MSE)	Reserved as ‘0’.	RO
0	I/O Space Enable (IOSE)	0 = Disable 1 = Enables access to the SM Bus I/O space registers as defined by the Base Address Register.	R/W

12.1.4 Offset 06 - 07h: STA—Device Status Register (SMBUS—D31:F3)

Table 428. Offset 06 - 07h: STA—Device Status Register (SMBUS—D31:F3)

Device:	31	**Function:**	3
Offset:	06-07h	**Attribute:**	Read-Only, Read/Write Clear
Default Value:	0280h	**Size:**	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	Reserved as ‘0’.	RO
14	Signaled System Error (SSE)	Reserved as ‘0’.	RO
13	Received Master Abort (RMA)	Reserved as ‘0’.	RO
12	Received Target Abort (RTA)	Reserved as ‘0’.	RO
11	Signaled Target-Abort Status	Reserved as ‘0’.	R/WC
10:9	DEVSEL# Timing Status (DEVT)	Reserved as ‘0’.	RO
8	Data Parity Error Detected	Reserved as ‘0’.	RO
7	Fast Back-to-Back Capable	Reserved as ‘0’.	RO
6	User Definable Features (UDF)	Reserved as ‘0’.	RO
5	66 MHz Capable	Reserved as ‘0’.	RO
4:0	Reserved	Reserved.	

12.1.5 Offset 08h: RID—Revision ID Register (SMBUS—D31:F3)

Table 429. Offset 08h: RID—Revision ID Register (SMBUS—D31:F3)

Device:	31	**Function:**	3
Offset:	08h	**Attribute:**	Read-Only
Default Value:	See bit description	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

12.1.6 Offset 09h: PI—Programming Interface (SMBUS—D31:F3)

Table 430. Offset 09h: PI—Programming Interface (SMBUS—D31:F3)

Device:	31	*Function:*	3
Offset:	09h	*Attribute:*	Read-Only
Default Value:	00h	*Size:*	8-bit

Bits	Name	Description	Access
7:0	Programming Interface		RO

12.1.7 Offset 0Ah: SCC—Sub Class Code Register (SMBUS—D31:F3)

Table 431. Offset 0Ah: SCC—Sub Class Code Register (SMBUS—D31:F3)

Device:	31	*Function:*	3
Offset:	0Ah	*Attribute:*	Read-Only
Default Value:	05h	*Size:*	8-bit

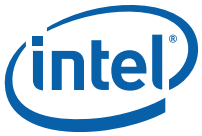
Bits	Name	Description	Access
7:0	Sub Class Code	05h = SM Bus serial controller	RO

12.1.8 Offset 0Bh: BCC—Base Class Code Register (SMBUS—D31:F3)

Table 432. Offset 0Bh: BCC—Base Class Code Register (SMBUS—D31:F3)

Device:	31	*Function:*	3
Offset:	0Bh	*Attribute:*	Read-Only
Default Value:	0Ch	*Size:*	8bit

Bits	Name	Description	Access
7:0	Base Class Code	0Ch = Serial controller.	RO



12.1.9 Offset 20 - 23h: SMB_BASE—SMBUS Base Address Register (SMBUS—D31:F3)

Table 433. Offset 20 - 23h: SMB_BASE—SMBUS Base Address Register (SMBUS—D31:F3)

Device:	31	***Function:***	3
Offset:	20-23h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	RO
15:5	Base Address	Provides the 32-byte system I/O base address for the Intel® 6300ESB ICH SMB logic.	R/W
4:1	Reserved	Reserved.	RO
0	IO Space Indicator	This read-only bit is always ‘1’, indicating that the SMB logic is I/O mapped.	RO

12.1.10 Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (SMBUS—D31:F2/F4)

Table 434. Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (SMBUS—D31:F2/F4)

Device:	31	***Function:***	3
Offset:	2Ch-2Dh	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem Vendor ID (SVID)	The SVID register, in combination with the Subsystem ID (SID) register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads from this register is the same as that which was written by BIOS into the IDE_SVID register.	RO

12.1.11 Offset 2Eh - 2Fh: SID—Subsystem ID (SMBUS—D31:F2/F4)

Table 435. Offset 2Eh - 2Fh: SID—Subsystem ID (SMBUS—D31:F2/F4)

Device:	31	**Function:**	3
Offset:	2Eh-2Fh	**Attribute:**	Read-Only
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Subsystem ID (SID)	The SID register, in combination with the SVID register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads from this register is the same as that which was written by BIOS into the IDE_SID register.	RO

12.1.12 Offset 3Ch: INTR_LN—Interrupt Line Register (SMBUS—D31:F3)

Table 436. Offset 3Ch: INTR_LN—Interrupt Line Register (SMBUS—D31:F3)

Device:	31	**Function:**	3
Offset:	3Ch	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Interrupt line	This data is not used by the Intel® 6300ESB ICH. It is to communicate to software that the interrupt line is connected to PIRQB#.	R/W

12.1.13 Offset 3Dh: INTR_PN—Interrupt Pin Register (SMBUS—D31:F3)

Table 437. Offset 3Dh: INTR_PN—Interrupt Pin Register (SMBUS—D31:F3)

Device:	31	**Function:**	3
Offset:	3Dh	**Attribute:**	Read-Only
Default Value:	02h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	Interrupt PIN	02h = Indicates that the Intel® 6300ESB ICH SMBus Controller will drive PIRQB# as its interrupt line.	RO

12.1.14 Offset 40h: HOSTC—Host Configuration Register (SMBUS—D31:F3)

Table 438. Offset 40h: HOSTC—Host Configuration Register (SMBUS—D31:F3)

Device:	31	***Function:***	3
Offset:	40h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	I^2C_EN	0 = SMBus behavior. 1 = The Intel® 6300ESB ICH is enabled to communicate with I^2C devices. This will change the formatting of some commands.	R/W
1	SMB_SMI_EN	0 = SMBus interrupts will not generate an SMI#. 1 = Any source of an SMB interrupt will instead be routed to generate an SMI#. Refer to Section 5.19.5, "Interrupts/SMI#". This bit needs to be set for SMBALERT# to be enabled.	R/W
0	HST_EN: SMBus Host Enable	0 = Disable the SMBus Host Controller. 1 = Enable. The SMB Host Controller interface is enabled to execute commands. The INTREN bit must be enabled for the SMB Host Controller to interrupt or SMI#. The SMB Host Controller will not respond to any new requests until all interrupt requests have been cleared.	R/W

12.2 SMBUS I/O Registers

Table 439. SMB I/O Registers (Sheet 1 of 2)

Offset	Mnemonic	Register Name/Function	Default	Access
00h	HST_STS	Host Status	00h	R/WC
02h	HST_CNT	Host Control	00h	R/W
03h	HST_CMD	Host Command	00h	R/W
04h	XMIT_SLVA	Transmit Slave Address	00h	R/W
05h	HST_D0	Host Data 0	00h	R/W
06h	HST_D1	Host Data 1	00h	R/W
07h	HOST_BLOCK_DB	Host Block Data Byte	00h	R/W
08h	PEC	Packet Error Check	00h	R/W
09h	RCV_SLVA	Receive Slave Address	44h	R/W
0A-0Bh	SLV_DATA	Slave Data	0000h	R/W
0Ch	AUX_STS	Auxiliary Status	00h	R/WC
0Dh	AUX_CTL	Auxiliary Control	00h	R/W

Table 439. SMB I/O Registers (Sheet 2 of 2)

Offset	Mnemonic	Register Name/Function	Default	Access
0Eh	SMLINK_PIN_CTL	SMLink Pin Control	See register description	R/W
0Fh	SMBUS_PIN_CTL	SMBus Pin Control	See register description	R/W
10h	SLV_STS	Slave Status	00h	R/WC
11h	SLV_CMD	Slave Command	00h	R/W
14h	NOTIFY_DADDR	Notify Device Address	00h	RO
16h	NOTIFY_DLOW	Notify Data Low Byte	00h	RO
17h	NOTIFY_DHIGH	Notify Data High Byte	00h	RO

12.2.1 Offset 00h: HST_STS—Host Status Register

Note: All status bits are set by hardware and cleared by the software writing a '1' to the particular bit position. Writing a '0' to any bit position has no effect.

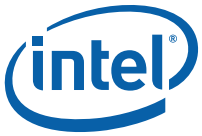


Table 440. Offset 00h: HST_STS—Host Status Register (Sheet 1 of 2)

Device:	31	**Function:**	3
Offset:	00h	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7	BYTE_DONE_STS	This bit will be set to '1' when the host controller has received a byte (for Block Read commands) or when it has completed transmission of a byte (for Block Write commands) when the 32-byte buffer is not being used. Note that this bit will be set even on the last byte of the transfer. Software clears the bit by writing a '1' to the bit position. This bit is not set when transmission is due to an external LAN Controller interface heartbeat. This bit has no meaning for block transfers when the 32-byte buffer is enabled. **NOTE:** When the last byte of a block message is received, the host controller will set this bit. However, it will not immediately set the INTR bit (bit 1 in this register). When the interrupt handler clears the BYTE_DONE_STS bit, another interrupt may be generated if the INTR bit is set. Thus, for a block message of n bytes, the Intel® 6300ESB ICH will generate n+1 interrupts. The interrupt handler needs to be implemented to handle these cases.	R/WC
6	INUSE_STS	This bit is used as semaphore among various independent software threads that may need to use the Intel® 6300ESB ICH's SMBus logic and has no other effect on hardware. 0 = After a full PCI reset, a read to this bit returns a '0'. 1 = After the first read, subsequent reads will return a '1'. A write of a '1' to this bit will reset the next read value to '0'. Writing a '0' to this bit has no effect. Software may poll this bit until it reads a 0 and will then own the usage of the host controller.	R/WC (special)
5	SMBALERT_STS	0 = Interrupt or SMI# was not generated by SMBALERT#. 1 = The source of the interrupt or SMI# was the SMBALERT# signal. This bit is only cleared by software writing a '1' to the bit position or by RSMRST# going low. When the signal is programmed as a GPIO, this bit will never be set.	R/WC
4	FAILED	0 = Cleared by writing a '1' to the bit position. 1 = The source of the interrupt or SMI# was a failed bus transaction. This bit is set in response to the KILL bit being set to terminate the host transaction.	R/WC
3	BUS_ERR	0 = Cleared by writing a '1' to the bit position. 1 = The source of the interrupt of SMI# was a transaction collision.	R/WC

Table 440. Offset 00h: HST_STS—Host Status Register (Sheet 2 of 2)

Device:	31	**Function:**	3
Offset:	00h	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
2	DEV_ERR	0 = Software resets this bit by writing a '1' to this location. The Intel® 6300ESB ICH will then deassert the interrupt or SMI#. 1 = The source of the interrupt or SMI# was due to one of the following: - Illegal Command Field - Unclaimed Cycle (host initiated) - Host Device Time-out Error - CRC Error	R/WC
1	INTR	This bit may be set only by termination of a command. INTR is not dependent on the INTREN bit of the Host Controller Register (offset 02h). It is only dependent on the termination of the command. When the INTREN bit is not set, the INTR bit will be set, although the interrupt will not be generated. Software may poll the INTR bit in this non-interrupt case. 0 = Software resets this bit by writing '1' to this location. The Intel® 6300ESB ICH will then deassert the interrupt or SMI#. 1 = The source of the interrupt or SMI# was the successful completion of its last command.	R/WC (special)
0	HOST_BUSY	0 = Cleared by the Intel® 6300ESB ICH when the current transaction is completed. 1 = Indicates that the Intel® 6300ESB ICH is running a command from the host interface. No SMB registers should be accessed while this bit is set, except the BLOCK DATA BYTE Register. The BLOCK DATA BYTE Register may be accessed when this bit is set only when the SMB_CMD bits in the Host Control Register are programmed for Block command or I^2C Read command. This is necessary in order to check the BYTE_DONE_STS bit.	RO

12.2.2 Offset 02h: HST_CNT—Host Control Register

Note: A read to this register will clear the byte pointer of the 32-byte buffer.

Table 441. Offset 02h: HST_CNT—Host Control Register (Sheet 1 of 3)

Device:	31	***Function:***	3
Offset:	02h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7	PEC_EN	0 = SMBus host controller does not perform the transaction with the PEC phase appended. 1 = Causes the host controller to perform the SMBus transaction with the Packet Error Checking phase appended. For writes, the value of the PEC byte is transferred from the PEC Register. For reads, the PEC byte is loaded into the PEC Register. This bit must be written prior to the write in which the **START** bit is set.	R/W
6	START	0 = This bit will always return '0' on reads. The HOST_BUSY bit in the Host Status register (offset 00h) may be used to identify when the Intel® 6300ESB ICH has finished the command. 1 = Writing a '1' to this bit initiates the command described in the SMB_CMD field. All registers should be setup prior to writing a '1' to this bit position.	WO
5	LAST_BYTE	This bit is used for Block Read commands. 1 = Software sets this bit to indicate that the next byte will be the last byte to be received for the block. This causes the Intel® 6300ESB ICH to send a NACK (instead of an ACK) after receiving the last byte. **NOTE:** Once the SECOND_TO_STS bit in TCO2_STS register (D31:F0, TCOBASE+6h, bit 1) is set, the LAST_BYTE bit also gets set. While the SECOND_TO_STS bit is set, the LAST_BYTE bit cannot be cleared. This prevents the Intel® 6300ESB ICH from running some of the SMBus commands (Block Read/Write, I^2C Read, Block I^2C Write).	WO

Table 441. Offset 02h: HST_CNT—Host Control Register (Sheet 2 of 3)

Device:	31	***Function:***	3
Offset:	02h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
4:2	SMB_CMD	The bit encoding below indicates which command the Intel® 6300ESB ICH is to perform. When enabled, the Intel® 6300ESB ICH will generate an interrupt or SMI# when the command has completed. When the value is for a non-supported or reserved command, the Intel® 6300ESB ICH will set the device error (DEV_ERR) status bit and generate an interrupt when the START bit is set. The Intel® 6300ESB ICH will not operate until DEV_ERR is cleared. **000 = Quick:** The slave address and read/write value (bit 0) are stored in the transmit slave address register. **001 = Byte:** This command uses the transmit slave address and command registers. Bit 0 of the slave address register determines if this is a read or write command. **010 = Byte Data:** This command uses the transmit slave address, command, and DATA0 registers. Bit 0 of the slave address register determines if this is a read or write command. When it is a read, the DATA0 register will contain the read data. **011 = Word Data:** This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. When it is a read, after the command completes, the DATA0 and DATA1 registers will contain the read data. **100 = Process Call:** This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. After the command completes, the DATA0 and DATA1 registers will contain the read data. **101 = Block:** This command uses the transmit slave address, command, DATA0 registers, and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block reads, the count is received and stored in the DATA0 register. Bit 0 of the slave address register selects if this is a read or write command. For writes, data is retrieved from the first n (where n is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. **110 = I^2C Read:** This command uses the transmit slave address, command, DATA0, DATA1 registers, and the Block Data Byte register. The read data is stored in the Block Data Byte register. The Intel® 6300ESB ICH will continue reading data until the NAK is received. **111 = Block Process:** This command uses the transmit slave address, command, DATA0 and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block read, the count is received and stored in the DATA0 register. Bit 0 of the slave address register always indicate a write command. For writes, data is retrieved from the first m (where m is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. **NOTE:** E32B bit in the Auxiliary Control register must be set for this command to work.	R/W

Table 441. Offset 02h: HST_CNT—Host Control Register (Sheet 3 of 3)

Device:	31	**Function:**	3
Offset:	02h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
1	KILL	0 = Normal SMBus Host Controller functionality. 1 = When set, kills the current host transaction taking place, sets the FAILED status bit, and asserts the interrupt (or SMI#). This bit, once set, must be cleared by software to allow the SMBus Host Controller to function normally.	R/W
0	INTREN	0 = Disable. 1 = Enable the generation of an interrupt or SMI# upon the completion of the command.	R/W

12.2.3 Offset 03h: HST_CMD—Host Command Register

Table 442. Offset 03h: HST_CMD—Host Command Register

Device:	31	**Function:**	3
Offset:	03h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0		This 8-bit field is transmitted by the host controller in the command field of the SMBus protocol during the execution of any command.	R/W

12.2.4 Offset 04h: XMIT_SLVA—Transmit Slave Address Register

Note: This register is transmitted by the host controller in the slave address field of the SMBus protocol.

Table 443. Offset 04h: XMIT_SLVA—Transmit Slave Address Register

Device:	31	**Function:**	3
Offset:	04h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
	ADDRESS	7-bit address of the targeted slave.	R/W
	RW	Direction of the host transfer. 0 = Write 1 = Read	R/W

12.2.5 Offset 05h: HST_D0—Data 0 Register

Table 444. Offset 05h: HST_D0—Data 0 Register

Device:	31	***Function:***	3
Offset:	05h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

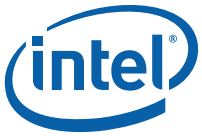
Bits	Name	Description	Access
7:0	DATA0/COUNT	This field contains the eight-bit data sent in the DATA0 field of the SMBus protocol. For block write commands, this register reflects the number of bytes to transfer. This register should be programmed to a value between 1 and 32 for block counts. A count of 0 or a count above 32 will result in unpredictable behavior. The host controller does not check or log illegal block counts.	R/W

12.2.6 Offset 06h: HST_D1—Data 1 Register

Table 445. Offset 06h: HST_D1—Data 1 Register

Device:	31	***Function:***	3
Offset:	06h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	DATA1	This eight-bit register is transmitted in the DATA1 field of the SMBus protocol during the execution of any command.	R/W



12.2.7 Offset 07h: Host_BLOCK_DB—Host Block Data Byte Register

Table 446. Offset 07h: Host_BLOCK_DB—Host Block Data Byte Register

Device:	31	***Function:***	3
Offset:	07h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Block Data (BDTA)	This is either a register or a pointer into a 32-byte block array, depending upon whether the E32B bit is set in the Auxiliary Control register. When the E32B bit is cleared, this is a register containing a byte of data to be sent on a block write or read from on a block read. When the E32B bit is set, reads and writes to this register are used to access the 32-byte block data storage array. An internal index pointer is used to address the array, which is reset to '0' by reading the HCTL register (offset 02h). The index pointer then increments automatically upon each access to this register. The transfer of block data into (read) or out of (write) this storage array during an SMBus transaction always starts at index address 0. When the E2B bit is set, for writes, software will write up to 32 bytes to this register as part of the setup for the command. After the Host Controller has sent the Address, Command, and Byte Count fields, it will send the bytes in the SRAM pointed to by this register. After the byte count has been exhausted, the controller will set the BYTE_DONE_STS bit . See Section 12.2.1,HST_STS-Host Status Register, bit 7. When the E2B bit is cleared for writes, software will place a single byte in this register. After the host controller has sent the address, command, and byte count fields, it will send the byte in this register. When there is more data to send, software will write the next series of bytes to the SRAM pointed to by this register and clear the BYTE_DONE_STS bit. The controller will then send the next byte. During the time between the last byte being transmitted to the next byte being transmitted, the controller will insert wait states on the interface. When the E2B bit is set for reads, after receiving the byte count into the Data0 register, the first series of data bytes go into the SRAM pointed to by this register. When the byte count has been exhausted or the 32-byte SRAM has been filled, the controller will generate an SMI# or interrupt (depending on configuration) and set the BYTE_DONE_STS bit. Software will then read the data. During the time between when the last byte is read from the SRAM to when the BYTE_DONE_STS bit is cleared, the controller will insert wait states on the interface.	R/W

12.2.8 Offset 08h: PEC—Packet Error Check Register

Table 447. Offset 08h: PEC—Packet Error Check Register

Device:	31	**Function:**	3
Offset:	08h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	PEC_DATA	This 8-bit register is written with the 8-bit CRC value that is used as the SMBus PEC data prior to a write transaction. For read transactions, the PEC data is loaded from the SMBus into this register and is then read by software. Software must ensure that the INUSE_STS bit is properly maintained to avoid having this field overwritten by a write transaction following a read transaction.	R/W

12.2.9 Offset 09h: RCV_SLVA—Receive Slave Address Register

Table 448. Offset 09h: RCV_SLVA—Receive Slave Address Register

Device:	31	**Function:**	3
Offset:	09h	**Attribute:**	Read/Write
Default Value:	44h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
7	Reserved	Reserved.	
6:0	SLAVE_ADDR	This field is the slave address that the Intel® 6300ESB ICH decodes for read and write cycles. The default is not 0, so the SMBus Slave Interface may respond even before the processor comes up or if the processor is dead. This register is cleared by RSMRST#, but not by PXPCIRST#.	R/W

12.2.10 Offset 0Ah: SLV_DATA—Receive Slave Data Register

Note: This register contains the 16-bit data value written by the external SMBus master. The processor may then read the value from this register. This register is reset by RSMRST# but not PXPCIRST#.

Table 449. Offset 0Ah: SLV_DATA—Receive Slave Data Register

Device:	31	***Function:***	3
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
15:8	DATA_MSG1: Data Message Byte 1	See Section 5.19.8, “SMBus Slave Interface” for a discussion of this field.	RO
7:0	DATA_MSG0: Data Message Byte 0	See Section 5.19.8, “SMBus Slave Interface” for a discussion of this field.	RO

12.2.11 Offset 0Ch: AUX_STS—Auxiliary Status Register

Table 450. Offset 0Ch: AUX_STS—Auxiliary Status Register

Device:	31	***Function:***	3
Offset:	0Ch	***Attribute:***	Read/Write Clear
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1	SMBus TCO mode (STCO)	This is the status bit that reflects the strap setting of legacy TCO mode vs. Advanced TCO mode. When set, it indicates that the Intel® 6300ESB ICH is in the advanced TCO mode. When cleared, the Intel® 6300ESB ICH is in the legacy/ compatible TCO mode. **NOTE:** For the Intel® 6300ESB ICH this bit is always 0, since Advanced TCO mode is not supported	RO
0	CRC Error (CRCE)	This bit is set when a received message contained a CRC error. When this bit is set, the DERR bit of the host status register will also be set. This bit will be set by the controller when a software abort occurs in the middle of the CRC portion of the cycle or an abort happens after the Intel® 6300ESB ICH has received the final data bit transmitted by an external slave.	R/WC

12.2.12 Offset 0Dh: AUX_CTL—Auxiliary Control Register

Table 451. Offset 0Dh: AUX_CTL—Auxiliary Control Register

Device:	31	**Function:**	3
Offset:	0Dh	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
7:2	Reserved	Reserved.	
1	Enable 32-byte Buffer (E32B)	When set, the Host Block Data register is a pointer into a 32-byte buffer, as opposed to a single register. This enables the block commands to transfer or receive up to 32 bytes before the Intel® 6300ESB ICH generates an interrupt.	R/W
0	Automatically Append CRC (AAC)	When set, the Intel® 6300ESB ICH will automatically append the CRC. This bit must not be changed during SMBus transactions, or undetermined behavior will result	R/W

12.2.13 Offset 0Eh: SMLINK_PIN_CTL—SMLink Pin Control Register

Note: This register is in the resume well and is reset by RSMRST#.

Table 452. Offset 0Eh: SMLINK_PIN_CTL—SMLink Pin Control Register

Device:	31	**Function:**	3
Offset:	0Eh	**Attribute:**	Read/Write
Default Value:	See Note	**Size:**	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SMLINK_CLK_CTL	This Read/Write bit has a default of 1. 0 = The Intel® 6300ESB ICH will drive the SMLINK[0] pin low, independent of what the other SMLINK logic would otherwise indicate for the SMLINK[0] pin. 1 = The SMLINK[0] pin is **not** overdriven low. The other SMLINK logic controls the state of the pin.	R/W
1	SMLINK1_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMLINK[1] pin. It will be '1' to indicate high, '0' to indicate low. This allows software to read the current state of the pin.	RO
0	SMLINK0_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMLINK[0] pin. It will be '1' to indicate high, '0' to indicate low. This allows software to read the current state of the pin.	RO

12.2.14 Offset 0Fh: SMBUS_PIN_CTL—SMBUS Pin Control Register

Note: This register is in the resume well and is reset by RSMRST#.

Table 453. Offset 0Fh: SMBUS_PIN_CTL—SMBUS Pin Control Register

Device:	31	***Function:***	3
Offset:	0Fh	***Attribute:***	Read/Write
Default Value:	See Note	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SMBCLK_CTL	This Read/Write bit has a default of 1. 1 = The SMBCLK pin is **not** overdriven low. The other SMBus logic controls the state of the pin. 0 = The Intel® 6300ESB ICH will drive the SMBCLK pin low, independent of what the other SMB logic would otherwise indicate for the SMBCLK pin.	R/W
1	SMBDATA_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMBDATA pin. It will be '1' to indicate high, '0' to indicate low. This allows software to read the current state of the pin.	RO
0	SMBCLK_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMBCLK pin. It will be '1' to indicate high, '0' to indicate low. This allows software to read the current state of the pin.	RO

12.2.15 Offset 10h: SLV_STS—Slave Status Register

Note: This register is in the resume well and is reset by RSMRST#.

All bits in this register are implemented in the 64 KHz clock domain. Therefore, software must poll this register until a write takes effect before assuming that a write has completed internally.

Table 454. Offset 10h: SLV_STS—Slave Status Register

Device:	31	**Function:**	3
Offset:	10h	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:1	Reserved	Reserved.	
0	HOST_NOTIFY_STS	The Intel® 6300ESB ICH sets this bit to a '1' when it has completely received a successful Host Notify Command on the SMLink pins. Software reads this bit to determine that the source of the interrupt or SMI# was the reception of the Host Notify Command. Software clears this bit after reading any information needed from the Notify address and data registers by writing a '1' to this bit. Note that the Intel® 6300ESB ICH will allow the Notify Address and Data registers to be overwritten once this bit has been cleared. When this bit is 1, the Intel® 6300ESB ICH will NACK the first byte (host address) of any new "Host Notify" commands on the SMLink. Writing a '0' to this bit has no effect.	R/WC

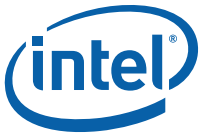
12.2.16 Offset 11H: SLV_CMD—Slave Command Register

Note: This register is in the resume well and is reset by RSMRST#.

Table 455. Offset 11H: SLV_CMD—Slave Command Register

Device:	31	**Function:**	3
Offset:	11h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2	SMBALERT_DIS	0 = Allows the generation of the interrupt or SMI#. 1 = Software sets this bit to block the generation of the interrupt or SMI# due to the SMBALERT# source. This bit is logically inverted and ANDed with the SMBALERT_STS bit. The resulting signal is distributed to the SMI# and/or interrupt generation logic. This bit does not affect the wake logic.	R/W
1	HOST_NOTIFY_WKEN	Software sets this bit to '1' to enable the reception of a Host Notify command as a wake event. When enabled, this event is 'OR'ed in with the other SMBus wake events and is reflected in the SMB_WAK_STS bit of the General Purpose Event 0 Status register.	R/W
0	HOST_NOTIFY_INTREN	Software sets this bit to '1' to enable the generation of interrupt or SMI# when HOST_NOTIFY_STS is 1. This enable does not affect the setting of the HOST_NOTIFY_STS bit. When the interrupt is generated, either PIRQ[B]# or SMI# is generated, depending on the value of the SMB_SMI_EN bit (D31, F3, Off40h, B1). If the HOST_NOTIFY_STS bit is set when this bit is written to a 1, then the interrupt (or SMI#) will be generated. The interrupt (or SMI#) is logically generated by 'AND'ing the STS and INTREN bits.	R/W



12.2.17 Offset 14h: NOTIFY_DADDR—Notify Device Address

Note: This register is in the resume well and is reset by RSMRST#.

Table 456. Offset 14h: NOTIFY_DADDR—Notify Device Address

Device:	31	***Function:***	3
Offset:	14h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:1	DEVICE_ADDRESS	This field contains the 7-bit device address received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to '1'.	RO
0	Reserved	Reserved.	

12.2.18 Offset 16h: NOTIFY_DLOW—Notify Data Low Byte Register

Note: This register is in the resume well and is reset by RSMRST#.

Table 457. Offset 16h: NOTIFY_DLOW—Notify Data Low Byte Register

Device:	31	***Function:***	3
Offset:	16h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	DATA_LOW_BYTE	This field contains the first (low) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to ‘1’.	RO

12.2.19 Offset 17h: NOTIFY_DHIGH—Notify Data High Byte Register

Note: This register is in the resume well and is reset by RSMRST#.

Table 458. Offset 17h: NOTIFY_DHIGH—Notify Data High Byte Register

Device:	31	***Function:***	3
Offset:	17h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	DATA_HIGH_BYTE	This field contains the second (high) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to ‘1’.	RO

AC'97 Audio Controller Registers (D31:F5) 13

13.1 AC'97 Audio PCI Configuration Space (D31:F5)

Note: Registers that are not shown should be treated as Reserved.

Table 459. PCI Configuration Map (Audio—D31:F5)

Offset	Mnemonic	Register	Default	Access
00-01h	VID	Vendor Identification	8086h	RO
02-03h	DID	Device Identification	25A6h	RO
04-05h	PCICMD	PCI Command	0000	R/W
06-07h	PCISTS	PCI Device Status	0290h	R/WC
08h	RID	Revision Identification	See Note 1	RO
09h	PI	Programming Interface	00	RO
0Ah	SCC	Sub Class Code	01h	RO
0Bh	BCC	Base Class Code	04h	RO
0Eh	HEDT	Header Type	00h	RO
10-13h	NAMBAR	Native Audio Mixer Base Address	00000000h	R/W
14-17h	NABMBAR	Native Audio Bus Mastering Base Address	00000000h	R/W
18-1Bh	MMBAR	Mixer Base Address (Mem)	00000000h	R/W
1C-1Fh	MBBAR	Bus Master Base Address (Mem)	00000000h	R/W
2C-2Dh	SVID	Subsystem Vendor ID	0000h	Write-Once
2E-2Fh	SID	Subsystem ID	0000h	Write-Once
34h	CAP_PTR	Capabilities Pointer	50h	RO
3Ch	INTR_LN	Interrupt Line	00h	R/W
3Dh	INTR_PN	Interrupt Pin	02h	RO
40h	PCID	Programmable Codec ID	09h	R/W
41h	CFG	Configuration	00h	R/W
50-51h	PID	PCI Power Management ID	0001h	RO
52-53h	PC	PC - Power Management Capabilities	C9C2h	RO
54-55h	PCS	Power Management Control and Status	0000h	R/W

NOTES:
1. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

Note: Internal reset as a result of $D3_{HOT}$ to D0 transition will reset all the core well registers except the following BIOS programmed registers as BIOS may not be invoked following the D3-to-D0 transition. Resume well registers will not be reset by the $D3_{HOT}$ to D0 transition.

Core Well registers **not** reset by the $D3_{HOT}$ to D0 transition:

- Offset 2Ch-2Dh – Subsystem Vendor ID (SVID)
- Offset 2Eh-2Fh – Subsystem ID (SID)
- Offset 40h – Programmable Codec ID (PCID)
- Offset 41h – Configuration (CFG)

Resume Well registers will **not** be reset by the $D3_{HOT}$ to D0 transition:

- Offset 54h-55h – Power Management Control and Status (PCS)
- Bus Mastering Register: Global Status Register, bit[17:16]
- Bus Mastering Register: SDATA_IN MAP register, bit[7:3]

13.1.1 Offset 00 - 01h: VID—Vendor Identification Register (Audio—D31:F5)

Table 460. Offset 00 - 01h: VID—Vendor Identification Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	00-01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel.	RO

13.1.2 Offset 02 - 03h: DID—Device Identification Register (Audio—D31:F5)

Table 461. Offset 02 - 03h: DID—Device Identification Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	02-03h	***Attribute:***	Read-Only
Default Value:	25A6h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Device ID Value		RO

13.1.3 Offset 04 - 05h: PCICMD—PCI Command Register (Audio—D31:F5)

Note: PCICMD is a 16-bit control register. Refer to the PCI 2.2 specification for complete details on each bit.

Table 462. Offset 04 - 05h: PCICMD—PCI Command Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	04-05h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:10	Reserved	Reserved. Read '0'.	
9	Fast Back-to-Back Enable (FBE)	Not implemented. Hardwired to '0'.	
8	SERR# Enable (SEN)	Not implemented. Hardwired to '0'.	
7	Wait Cycle Control (WCC)	Not implemented. Hardwired to '0'.	
6	Parity Error Response (PER)	Not implemented. Hardwired to '0'.	
5	VGA Palette Snoop (VPS)	Not implemented. Hardwired to '0'.	
4	Memory Write and Invalidate Enable (MWI)	Not implemented. Hardwired to '0'.	
3	Special Cycle Enable (SCE)	Not implemented. Hardwired to '0'.	
2	Bus Master Enable (BME)	Controls standard PCI bus mastering capabilities. 0 = Disable. 1 = Enable	R/W
1	Memory Space Enable (MSE)	Enables memory space addresses to the AC'97 Audio Controller.	R/W

13.1.4 Offset 06 - 07h: PCISTS—PCI Device Status Register (Audio—D31:F5)

Note: PCISTS is a 16-bit status register. Refer to the PCI 2.2 specification for complete details on each bit.

Note: When a master abort occurs on a memory read request performed by a particular channel, the run bit for that channel gets cleared immediately, and the corresponding DMA engine halts. Write requests are posted and hence aborts cannot be seen by the Intel® 6300ESB ICH AC'97 controller for write requests.

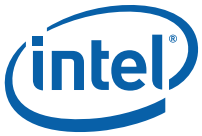


Table 463. Offset 06 - 07h: PCISTS—PCI Device Status Register (Audio—D31:F5)

Device:	31	**Function:**	5
Offset:	06-07h	**Attribute:**	Read/Write Clear
Default Value:	0290h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	Not implemented. Hardwired to ‘0’.	
14	SERR# Status (SERRS)	Not implemented. Hardwired to ‘0’.	
13	Master-Abort Status (MAS)	0 = Software clears this bit by writing a ‘1’ to the bit position. 1 = Bus Master AC '97 2.2 interface function, as a master, generates a master abort.	R/WC
12	Reserved	Reserved. Will always read as ‘0’.	
11	Signaled Target-Abort Status (STA)	Not implemented. Hardwired to ‘0’.	
10:9	DEVSEL# Timing Status (DEVT)	This 2-bit field reflects the Intel® 6300ESB ICH's DEVSEL# timing when performing a positive decode. 01b = Medium timing. Hardwired to ‘01’.	RO
8	Data Parity Detected (DPD)	Not implemented. Hardwired to ‘0’.	
7	Fast Back to back Capable (FBC)	Hardwired to ‘1’. This bit indicates that the Intel® 6300ESB ICH as a target is capable of fast back-to-back transactions.	
6	Reserved	Reserved. Hardwired to ‘0’.	
5	66 MHz Capable	Hardwired to ‘0’.	
4	Capabilities List Exists (CLIST)	Indicates that the controller contains a capabilities pointer list. The first item is pointed to by looking at configuration offset 34h.	
3:0	Reserved	Reserved.	

13.1.5 Offset 08h: RID—Revision Identification Register (Audio—D31:F5)

Table 464. Offset 08h: RID—Revision Identification Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

13.1.6 Offset 09h: PI—Programming Interface Register (Audio—D31:F5)

Table 465. Offset 09h: PI—Programming Interface Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	09h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Programming Interface		RO

13.1.7 Offset 0Ah: SCC—Sub Class Code Register (Audio—D31:F5)

Table 466. Offset 0Ah: SCC—Sub Class Code Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Sub Class Code	01h = Audio Device This indicates the device is an audio device, in the context of a multimedia device (Base Class Code = 04h).	RO

13.1.8 Offset 0Bh: BCC—Base Class Code Register (Audio—D31:F5)

Table 467. Offset 0Bh: BCC—Base Class Code Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	04H	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Base Class Code	04h = Multimedia device	RO

13.1.9 Offset 0Eh: HEDT—Header Type Register (Audio—D31:F5)

Table 468. Offset 0Eh: HEDT—Header Type Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	0Eh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Header Type Value	Hardwired to 00h.	

13.1.10 Offset 10 - 13h: NAMBAR—Native Audio Mixer Base Address Register (Audio—D31:F5)

The Native PCI Mode Audio function uses PCI Base Address register 1 to request a contiguous block of I/O space that is to be used for the Native Audio Mixer software interface. The mixer requires 256 bytes of I/O space. This 256 bytes space is divided into 128 bytes for the primary codec (offsets 00-7Fh) and 128 bytes for the secondary codec (offsets 80-FFh). Access to these registers will be decoded by the AC'97 controller and forwarded over the AC-link to the codec. The codec will then respond with the register value.

In the case of split codec implementation, accesses to the different codecs are differentiated by the controller by using address offsets 00h - 7Fh for the primary codec and address offsets 80h - FEh for the secondary codec.

Note: The tertiary codec cannot be addressed through this address space. The tertiary space is only available from the new MMBAR register. This register powers up as read only and only becomes writeable when the IOSE bit in offset 41h is set.

For descriptions of these I/O registers, refer to the AC'97 specification.

Table 469. Offset 10 - 13h: NAMBAR—Native Audio Mixer Base Address Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	10-13h	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved. All bits are hardwired to '0'.	
15:8	Base Address	These bits are used in the I/O space decode of the Native Audio Mixer interface registers. The number of upper bits that a device actually implements depends on how much of the address space the device will respond to. For the AC'97 mixer, the upper 16 bits are hardwired to '0', while bits 15:8 are programmable. This configuration yields a maximum I/O block size of 256 bytes for this base address.	R/W
7:1	Reserved	Reserved. Read as '0's.	
0	Resource Type Indicator (RTE)	This read-only bit defaults to '0' and flips to '1' if bit '0' of offset 41h is set. When this bit is set to '1', it indicates a request for I/O space.	RO

13.1.11 Offset 14 - 17h: NABMBAR—Native Audio Bus Mastering Base Address Register (Audio—D31:F5)

The Native PCI Mode Audio function uses PCI Base Address register 1 to request a contiguous block of I/O space that is to be used for the Native Mode Audio software interface. This BAR creates 64 bytes of I/O space to signify the base address of the bus master I/O space.

Note: The DMA registers for S/PDIF and Microphone In 2 cannot be addressed through this address space. These DMA functions are only available from the new MBBAR register. This register powers up as read only and only becomes writeable when the IOSE bit in offset 41h is set.

Table 470. Offset 14 - 17h: NABMBAR—Native Audio Bus Mastering Base Address Register (Audio—D31:F5)

Device:	31	**Function:**	5
Offset:	14-17h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved. Hardwired to '0'.	
15:6	Base Address	These bits are used in the I/O space decode of the Native Audio Bus Mastering interface registers. The number of upper bits that a device actually implements depends on how much of the address space the device will respond to. For AC'97 bus mastering, the upper 16 bits are hardwired to '0', while bits 15:6 are programmable. This configuration yields a maximum I/O block size of 64 bytes for this base address.	R/W
5:1	Reserved	Reserved. Read as '0's.	
0	Resource Type Indicator (RTE)	This read-only bit defaults to '0' and flips to '1' if bit '0' of offset 41h is set. When this bit is set to '1', it indicates a request for I/O space.	RO

13.1.12 Offset 18 - 1Bh: MMBAR—Mixer Base Address Register (Audio—D31:F5)

This BAR creates 512 bytes of memory space to signify the base address of the register space. The lower 256 bytes of this space map to the same registers as the 256-byte I/O space pointed to by NAMBAR. The lower 384 bytes are divided as follows:

- 128 bytes for the primary codec (offsets 00 – 7Fh)
- 128 bytes for the secondary codec (offsets 80h – FFh)
- 128 bytes for the tertiary codec (offsets 100h – 17Fh).
- 128 bytes of reserved space (offsets 180h – 1FFh), returning all '0'.

Table 471. Offset 18 - 1Bh: MMBAR—Mixer Base Address Register (Audio—D31:F5) (Sheet 1 of 2)

Device:	31	**Function:**	5
Offset:	18-1Bh	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:9	Base Address	Lower 32 bits of the 512-byte memory offset to use for decoding the primary, secondary, and tertiary codec's mixer spaces.	R/W

Table 471. Offset 18 - 1Bh: MMBAR—Mixer Base Address Register (Audio—D31:F5) (Sheet 2 of 2)

Device:	31	***Function:***	5
Offset:	18-1Bh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
8:3	Reserved	Reserved. Read as ‘0’s.	RO
2:1	Type	Indicates the base address exists in 32-bit address space	RO
0	Resource Type Indicator (RTE)	This bit is set to ’0’, indicating a request for memory space.	RO

13.1.13 Offset 1C - 1Fh: MBBAR—Bus Master Base Address Register (Audio—D31:F5)

Note: This BAR creates 256 bytes of memory space to signify the base address of the bus master memory space. The lower 64 bytes of the space pointed to by this register point to the same registers as the NABMBAR.

Table 472. Offset 1C - 1Fh: MBBAR—Bus Master Base Address Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	1C-1Fh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:8	Base Address	I/O offset to use for decoding the PCM In, PCM Out, and Microphone 1 DMA engines.	R/W
7:3	Reserved	Reserved. Read as ‘0’s.	RO
2:1	Type	Indicates the base address exists in 32-bit address space.	RO
0	Resource Type Indicator (RTE)	This bit is set to ’0’, indicating a request for memory space.	RO

13.1.14 Offset 2D - 2Ch: SVID—Subsystem Vendor ID Register (Audio—D31:F5)

The SVID register, in combination with the Subsystem ID register, enable the operating environment to distinguish one audio subsystem from the other(s).

This register is implemented as write-once register. Once a value is written to it, the value may be read back. Any subsequent writes will have no effect.

This register is not affected by the $D3_{HOT}$ to D0 transition.

Table 473. Offset 2D - 2Ch: SVID—Subsystem Vendor ID Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	2D-2Ch	***Attribute:***	Read/Write Once
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem Vendor ID Value		R/WO

13.1.15 Offset 2E - 2Fh: SID—Subsystem ID Register (Audio—D31:F5)

The SID register, in combination with the Subsystem Vendor ID register make it possible for the operating environment to distinguish one audio subsystem from the other(s).

This register is implemented as write-once register. Once a value is written to it, the value may be read back. Any subsequent writes will have no effect.

This register is not affected by the $D3_{HOT}$ to D0 transition.

Table 474. Offset 2E - 2Fh: SID—Subsystem ID Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	2E-2Fh	***Attribute:***	Read/Write Once
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem ID Value		R/WO

13.1.16 Offset 34h: CAP_PTR—Capabilities Pointer (Audio—D31:F5)

Note: This register indicates the offset for the capability pointer.

Table 475. Offset 34h: CAP_PTR—Capabilities Pointer (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	34h	***Attribute:***	Read-Only
Default Value:	50h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Capability Pointer (CP)	Indicates that the first capability pointer offset is offset 50h.	RO

13.1.17 Offset 3Ch: INTR_LN—Interrupt Line Register (Audio—D31:F5)

Note: This register indicates which PCI interrupt line is used for the AC'97 module interrupt. The AC'97 interrupt is internally ORed to the interrupt controller with the PIRQ[B]# signal.

Table 476. Offset 3Ch: INTR_LN—Interrupt Line Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	3Ch	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is used to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

13.1.18 Offset 3Dh: INTR_PN—Interrupt Pin Register (Audio—D31:F5)

Note: This register indicates which PCI interrupt pin is used for the AC'97 module interrupt. The AC'97 interrupt is internally OR'd to the interrupt controller with the PIRQB# signal.

Table 477. Offset 3Dh: INTR_PN—Interrupt Pin Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	3Dh	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2:0	AC'97 Interrupt Routing	Hardwired to 010b to select PIRQB#.	RO

13.1.19 Offset 40h: PCID—Programmable Codec ID Register (Audio—D31:F5)

Note: This register is used to specify the ID for the secondary and tertiary codecs for I/O accesses. This register is not affected by the $D3_{HOT}$ to D0 transition.

Note: The value in this register must only be modified prior to any AC'97 codec accesses.

Table 478. Offset 40h: PCID—Programmable Codec ID Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	40h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3:2	Tertiary Codec ID (TID)	These bits define the encoded ID that is used to address the tertiary codec I/O space. Bit '1' is the first bit sent and Bit '0' is the second bit sent on AC_SDATA_OUT during slot 0.	R/W
1:0	Secondary Codec ID (SCID)	These two bits define the encoded ID that is used to address the secondary codec I/O space. The two bits are the ID that will be placed on slot 0, bits '0' and '1', upon an I/O access to the secondary codec. Bit '1' is the first bit sent and bit '0' is the second bit sent on AC_SDATA_OUT during slot 0.	R/W

13.1.20 Offset 41h: CFG—Configuration Register (Audio—D31:F5)

Note: This register is used to specify the ID for the secondary and tertiary codecs for I/O accesses. This register is not affected by the $D3_{HOT}$ to D0 transition.

Table 479. Offset 41h: CFG—Configuration Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	41h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:1	Reserved	Reserved.	RO
0	I/O Space Enable (IOSE)	When cleared, the IOSE bit at offset 04h and the I/O space BARs at offset 10h and 14h become read-only registers. This is the default state for the I/O BARs. BIOS must explicitly set this bit to allow a legacy driver to work.	R/W

13.1.21 Offset 50h: PID—PCI Power Management Capability ID Register (Audio—D31:F5)

Table 480. Offset 50h: PID—PCI Power Management Capability ID Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	50h	***Attribute:***	Read-Only
Default Value:	0001h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:8	Next Capability (NEXT)	Indicates that the next item in the list is at offset 00h.	RO
7:0	Cap ID (CAP)	Indicates that this pointer is a message signaled interrupt capability.	RO

13.1.22 Offset 52h: PC—Power Management Capabilities Register (Audio—D31:F5)

Note: This register is not affected by the $D3_{HOT}$ to D0 transition.

Table 481. Offset 52h: PC—Power Management Capabilities Register (Audio—D31:F5)

Device:	31	***Function:***	5
Offset:	52h	***Attribute:***	Read-Only
Default Value:	C9C2h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:11	PME_Support	Indicates PME# may be generated from all D states.	RO
10:9	Reserved	Reserved.	RO
8:6	Aux_Current	Reports 375mA maximum Suspend well current required when in the D3 cold state.	RO
5	Device Specific Initial-ization (DSI)	Indicates that no device-specific initialization is required.	RO
4	Reserved	Reserved.	RO
3	PME Clock (PMEC)	Indicates that PCI clock is not required to generate PME#.	RO
2:0	Version (VS)	Indicates support for Revision 1.1 of the PCI Power Management Specification.	RO

13.1.23 Offset 54h: PCS—Power Management Control and Status Register (Audio—D31:F5)

Table 482. Offset 54h: PCS—Power Management Control and Status Register (Audio—D31:F5) (Sheet 1 of 2)

Device:	31	***Function:***	5
Offset:	54h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Resume

Bits	Name	Description	Access
15	PME Status (PMES)	This bit is set when the AC'97 controller would normally assert the PME# signal independent of the state of the PME_En bit. This bit resides in the resume well.	R/WC
14:9	Reserved	Reserved.	RO

Table 482. Offset 54h: PCS—Power Management Control and Status Register (Audio—D31:F5) (Sheet 2 of 2)

Device:	31	**Function:**	5
Offset:	54h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
8	Power Management Event Enable (PMEE)	When set, and if corresponding PMES is also set, the AC'97 controller sets the AC97_STS bit in the GPE0_STS register.	R/W
7:2	Reserved	Reserved.	RO
1:0	Power State (PS)	This field is used both to determine the current power state of the AC'97 controller and to set a new power state. The values are: 00 – D0 state 01 – not supported 10 – not supported 11 – $D3_{HOT}$ state When in the $D3_{HOT}$ state, the AC'97 controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. When software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally. However, the data is discarded and no state change occurs.	R/W

13.2 AC'97 Audio I/O Space (D31:F5)

The AC'97 I/O space includes Native Audio Bus Master Registers and Native Mixer Registers. For the Intel® 6300ESB ICH, the offsets are important as they will determine bits[1:0] of the TAG field (codec ID).

Audio Mixer I/O space may be accessed as a 16-bit field only since the data packet length on
AC-link is a word. Any S/W access to the codec will be done as a 16-bit access starting from the first active byte. In case no byte enables are active, the access will be done at the first word of the qWord that contains the address of this request.

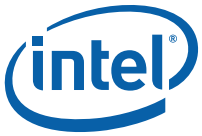


Table 483. Intel® 6300ESB I/O Controller Hub Audio Mixer Register Configuration (Sheet 1 of 2)

Primary Offset (Codec ID =00)	Secondary Offset (Codec ID =01)	Tertiary Offset (Codec ID =10)	NAMBAR Exposed Registers (D31:F5)
00h	80h	100h	Reset
02h	82h	102h	Master Volume
04h	84h	104h	Aux Out Volume
06h	86h	106h	Mono Volume
08h	88h	108h	Master Tone (R and L)
0Ah	8Ah	10Ah	PC_BEEP Volume
0Ch	8Ch	10Ch	Phone Volume
0Eh	8Eh	10Eh	Mic Volume
10h	90h	110h	Line In Volume
12h	92h	112h	CD Volume
14h	94h	114h	Video Volume
16h	96h	116h	Aux In Volume
18h	98h	118h	PCM Out Volume
1Ah	9Ah	11Ah	Record Select
1Ch	9Ch	11Ch	Record Gain
1Eh	9Eh	11Eh	Record Gain Mic
20h	A0h	120h	General Purpose
22h	A2h	122h	3D Control
24h	A4h	124h	AC'97 RESERVED
26h	A6h	126h	Powerdown Ctrl/Stat
28h	A8h	128h	Extended Audio
2Ah	AAh	12Ah	Extended Audio Ctrl/Stat
2Ch	ACh	12Ch	PCM Front DAC Rate
2Eh	AEh	12Eh	PCM Surround DAC Rate
30h	B0h	130h	PCM LFE DAC Rate
32h	B2h	132h	PCM LR ADC Rate
34h	B4h	134h	MIC ADC Rate
36h	B6h	136h	6Ch Vol: C, LFE
38h	B8h	138h	6Ch Vol: L, R Surround
3Ah	BAh	13Ah	S/PDIF Control
3C-56h	BC-D6h	13C-156h	Intel RESERVED
58h	D8h	158h	AC'97 Reserved

NOTES:
1. Software should not try to access reserved registers.
2. Primary Codec ID cannot be changed. Secondary codec ID may be changed through bits 1:0 of configuration register 40h. Tertiary codec ID may be changed through bits 3:2 of configuration register 40h.
3. The tertiary offset is only available through the memory space defined by the MMBAR register.

Table 483. Intel® 6300ESB I/O Controller Hub Audio Mixer Register Configuration (Sheet 2 of 2)

Primary Offset (Codec ID =00)	Secondary Offset (Codec ID =01)	Tertiary Offset (Codec ID =10)	NAMBAR Exposed Registers (D31:F5)
5Ah	DAh	15Ah	Vendor Reserved
7Ch	FCh	17Ch	Vendor ID1
7Eh	FEh	17Eh	Vendor ID2

NOTES:
1. Software should not try to access reserved registers.
2. Primary Codec ID cannot be changed. Secondary codec ID may be changed through bits 1:0 of configuration register 40h. Tertiary codec ID may be changed through bits 3:2 of configuration register 40h.
3. The tertiary offset is only available through the memory space defined by the MMBAR register.

The Bus Master registers are located from offset + 00h to offset + 51h and reside in the AC'97 controller. Accesses to these registers do NOT cause the cycle to be forwarded over the AC-link to the codec. S/W could access these registers as bytes, word, or dWord quantities, but reads must not cross dWord boundaries.

In the case of split codec implementation, accesses to the different codecs are differentiated by the controller by using address offsets 00h - 7Fh for the primary codec, address offsets 80h - FFh for the secondary codec and address offsets 100h - 17Fh for the tertiary codec.

The Global Control (GLOB_CNT) and Global Status (GLOB_STA) 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.

Bus Mastering registers exist in I/O space and reside in the AC'97 controller. The six channels, PCM in, PCM in 2, PCM out, Mic in, Mic 2, and S/PDIF out, each have their own set of Bus Mastering registers. The following register descriptions apply to all six channels. The register definition section titles use a generic "x_" in front of the register to indicate that the register applies to all six channels. The naming prefix convention used in Table 484 and in the register description I/O address is as follows:

PI = PCM in channel
PO = PCM out channel
MC = Mic in channel.
MC2 = Mic 2 channel
PI2 = PCM in 2 channel
SP = S/PDIF out channel.

Table 484. Native Audio Bus Master Control Registers (Sheet 1 of 3)

Offset	Mnemonic	Name	Default	Access
00h	PI_BDBAR	PCM in Buffer Descriptor list Base Address Register	00000000h	R/W
04h	PI_CIV	PCM in Current Index Value	00h	RO
05h	PI_LVI	PCM in Last Valid Index	00h	R/W
06h	PI_SR	PCM in Status Register	0003h	R/W
08h	PI_PICB	PCM in Position in Current Buffer	0000h	RO
0Ah	PI_PIV	PCM in Prefetched Index Value	00h	RO
0Bh	PI_CR	PCM in Control Register	00h	R/W

Table 484. Native Audio Bus Master Control Registers (Sheet 2 of 3)

Offset	Mnemonic	Name	Default	Access
10h	PO_BDBAR	PCM out Buffer Descriptor list Base Address Register	00000000h	R/W
14h	PO_CIV	PCM out Current Index Value	00h	RO
15h	PO_LVI	PCM out Last Valid Index	00h	R/W
16h	PO_SR	PCM out Status Register	0001h	R/W
18h	PO_PICB	PCM out Position In Current Buffer	0000h	RO
1Ah	PO_PIV	PCM out Prefetched Index Value	00h	RO
1Bh	PO_CR	PCM out Control Register	00h	R/W
20h	MC_BDBAR	Mic. in Buffer Descriptor List Base Address Register	00000000h	R/W
24h	MC_CIV	Mic. in Current Index Value	00h	RO
25h	MC_LVI	Mic. in Last Valid Index	00h	R/W
26h	MC_SR	Mic. In Status Register	0001h	R/W
28h	MC_PICB	Mic. In Position In Current Buffer	0000h	RO
2Ah	MC_PIV	Mic. in Prefetched Index Value	00h	RO
2Bh	MC_CR	Mic. in Control Register	00h	R/W
2Ch	GLOB_CNT	Global Control	00000000h	R/W
30h	GLOB_STA	Global Status	00700000h	RO
34h	ACC_SEMA	Codec Write Semaphore Register	00h	R/W
40-43h	MC2_BDBAR	Mic. 2 Buffer Descriptor List Base Address Register	00h	RO
44h	MC2_CIV	Mic. 2 Current Index Value	00h	R/W
45h	MC2_LVI	Mic. 2 Last Valid Index	0001h	R/W
46-47h	MC2_SR	Mic. 2 Status Register	0000h	RO
48-49h	MC2_PICB	Mic. 2 Position In Current Buffer	00h	RO
4Ah	MC2_PIV	Mic. 2 Prefetched Index Value	00h	R/W
4Bh	MC2_CR	Mic. 2 Control Register	00h	RO
50-53h	PI2_BDBAR	PCM in 2 Buffer Descriptor List Base Address Register	00000000h	R/W
54h	PI2_CIV	PCM in 2 Current Index Value	00h	RO
55h	PI2_LVI	PCM in 2 Last Valid Index	00h	R/W
56-57h	PI2_SR	PCM in 2 Status Register	0001h	R/W
58-59h	PI2_PICB	PCM in 2 Position in Current Buffer	0000h	RO
5Ah	PI2_PIV	PCM In 2 Prefetched Index Value	00h	RO
5Bh	PI2_CR	PCM In 2 Control Register	00h	R/W
60-63	SP_BAR	S/PDIF Buffer Descriptor List Base Address Register	00000000h	R/W
64h	SP_CIV	S/PDIF Current Index Value	00h	RO
65h	SP_LVI	S/PDIF Last Valid Index	00h	R/W
66-67h	SP_SR	S/PDIF Status Register	0001h	R/W

Table 484. Native Audio Bus Master Control Registers (Sheet 3 of 3)

Offset	Mnemonic	Name	Default	Access
68-69h	SP_PICB	S/PDIF Position In Current Buffer	0000h	RO
6Ah	SP_PIV	S/PDIF Prefetched Index Value	00h	RO
6Bh	SP_CR	S/PDIF Control Register	00h	R/W
80h	SDM	SData_IN Map	00h	R/W

Note: Internal reset as a result of $D3_{HOT}$ to D0 transition will reset all the core well registers except the registers shared with the AC'97 Modem (GCR, GSR, CASR). Resume well registers will not be reset by the $D3_{HOT}$ to D0 transition.

Core Well registers and bits NOT reset by the $D3_{HOT}$ to D0 transition:

- Offset 2Ch-2Fh – bits[15,6:0] Global Control (GLOB_CNT)
- Offset 30h-33h – bits[29,15,11:10,0] Global Status (GLOB_STA)
- Offset 34h – Codec Access Semaphore Register (CAS)

Resume Well registers and bits will NOT be reset by the $D3_{HOT}$ to D0 transition:

- Offset 30h-33h – bits[17:16] Global Status (GLOB_STA)

13.2.1 *x*_BDBAR—Buffer Descriptor Base Address Register

Note: Software may read the register at offset 00h by performing a single 32-bit read from address offset 00h. Reads across dWord boundaries are not supported.

Table 485. x_BDBAR—Buffer Descriptor Base Address Register

Device:	31	***Function:***	5
I/O Address:	NABMBAR + 00h (PIBDBAR), NABMBAR + 10h (POBDBAR), NABMBAR + 20h (MCBDBAR), MBBAR + 40h (MC2BDBAR), MBBAR + 50h (PI2BDBAR), MBBAR + 60h (SPBAR)	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:3	Buffer Descriptor Base Address[31:3]	These bits represent address bits 31:3. The data should be aligned on 8 byte boundaries. Each buffer descriptor is 8 bytes long, and the list may contain a maximum of 32 entries.	R/W
2:0		Hardwired to '0'.	

13.2.2 *x*_CIV—Current Index Value Register

Note: Software may read the registers at offsets 04h, 05h and 06h simultaneously by performing a single 32-bit read from address offset 04h. Software may also read this register individually by doing a single 8-bit read to offset 04h. Reads across dWord boundaries are not supported.

Table 486. x_CIV—Current Index Value Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 04h (PICIV), NABMBAR + 04h (PICIV), NABMBAR + 04h (PICIV), MBBAR + 44h (MC2CIV), MBBAR + 54h (PI2CIV), MBBAR + 64h (SPCIV)	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5		Hardwired to '0'.	
4:0	Current Index Value[4:0]	These bits represent which buffer descriptor within the list of 32 descriptors is currently being processed. As each descriptor is processed, this value is incremented. The value rolls over after it reaches 31.	RO

13.2.3 x_LVI—Last Valid Index Register

Table 487. x_LVI—Last Valid Index Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 05h (PILVI), NABMBAR + 15h (POLVI), NABMBAR + 15h (POLVI), NABMBAR + 15h (POLVI), MBBAR + 55h (PI2LVI), MBBAR + 65h (SPLVI)	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5		Hardwired to '0'.	
4:0	Last Valid Index[4:0]	This value represents the last valid descriptor in the list. This value is updated by the software each time it prepares a new buffer and adds it to the list.	R/W

13.2.4 x_SR—Status Register

Note: Software may read the registers at offsets 04h, 05h and 06h simultaneously by performing a single 32-bit read from address offset 04h. Software may also read this register individually by doing a single 8-bit read to offset 05h. Reads across dWord boundaries are not supported.

Table 488. x_SR—Status Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 06h (PISR), NABMBAR + 16h (POSR), NABMBAR + 26h (MCSR), MBBAR + 46h (MC2SR), MBBAR + 56h (PI2SR), MBBAR + 66h (SPSR)	**Attribute:**	Read/Write Clear, Read-Only
Default Value:	0001h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:5	Reserved	Reserved.	RO
4	FIFO Error (FIFOE)	0 = Cleared by writing a “1” to this bit position. 1 = FIFO error occurs. **PISR Register:** FIFO error indicates a FIFO overrun. The FIFO pointers don't increment; the incoming data is not written into the FIFO, thus is lost. **POSR Register:** FIFO error indicates a FIFO underrun. The sample transmitted in this case should be the last valid sample. The Intel® 6300ESB ICH will set the FIFOE bit if the underrun or overrun occurs when there are more valid buffers to process.	R/WC
3	Buffer Completion Interrupt Status (BCIS)	0 = Cleared by writing a “1” to this bit position. 1 = 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. It remains active until cleared by software.	R/WC
2	Last Valid Buffer Completion Interrupt (LVBCI)	0 = Cleared by writing a “1” to this bit position. 1 = 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 may be cleared by software once this event has been recognized. This event will cause an interrupt when the enable bit in the Control Register is set. The interrupt is cleared when the software clears this bit. In the case of *Transmits* (PCM out, Modem out) this bit is set after the last valid buffer has been fetched, not after transmitting it. In the case of *Receives*, this bit is set after the data for the last buffer has been written to memory.	R/WC
1	Current Equals Last Valid (CELV)	0 = Cleared by hardware when controller exits state (i.e., until a new value is written to the LVI register.) 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.	RO
0	DMA Controller Halted (DCH)	0 = Running. 1 = Halted. This could happen because of the Start/Stop bit being cleared and the DMA engines are idle, or it could happen once the controller has processed the last valid buffer.	RO

13.2.5 x_PICB—Position In Current Buffer Register

Table 489. x_PICB—Position In Current Buffer Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 08h (PIPICB), NABMBAR + 18h (POPICB), NABMBAR + 28h (MCPICB), MBBAR + 48h (MC2PICB), MBBAR + 58h (PI2PICB), MBBAR + 68h (SPPICB)	**Attribute:**	Read-Only
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Position In Current Buffer[15:0]	These bits represent the number of samples left to be processed in the current buffer. This means the number of samples not yet read from memory (in the case of reads from memory) or not yet written to memory (in the case of writes to memory) irrespective of the number of samples that have been transmitted/received across AC-link.	RO

13.2.6 x_PIV—Prefetched Index Value Register

Note: Software may read the registers at offsets 08h, 0Ah, and 0Bh by performing a 32-bit read from address offset 08h. Software may also read this register individually by doing a single 8-bit read to offset 0Ah. Reads across dWord boundaries are not supported.

Table 490. x_PIV—Prefetched Index Value Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 0Ah (PIPIV), NABMBAR + 1Ah (POPIV), NABMBAR + 2Ah (MCPIV), MBBAR + 4Ah (MC2PIV), MBBAR + 5Ah (PI2PIV), MBBAR + 6Ah (SPPIV)	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5		Hardwired to 0.	
4:0	Prefetched Index Value[4:0]	These bits represent which buffer descriptor in the list has been prefetched. The bits in this register are also modulo 32 and roll over after they reach 31.	RO

13.2.7 x_CR—Control Register

Note: Software may read the registers at offsets 08h, 0Ah, and 0Bh by performing a 32-bit read from address offset 08h. Software may also read this register individually by doing a single 8-bit read to offset 0Bh. Reads across dWord boundaries are not supported.

Table 491. x_CR—Control Register

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 0Bh (PICR), NABMBAR + 1Bh (POCR), NABMBAR + 2Bh (MCCR), MBBAR + 4Bh (MC2CR), MBBAR + 5Bh (PI2CR), MBBAR + 6Bh (SPCR)	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	Interrupt On Completion Enable (IOCE)	This bit controls whether or not an interrupt occurs when a buffer completes with the IOC bit set in its descriptor. 0 = Disable. Interrupt will not occur. 1 = Enable.	R/W
3	FIFO Error Interrupt Enable (FEIE)	This bit controls whether the occurrence of a FIFO error will cause an interrupt or not. 0 = Disable. Bit 4 in the Status Register will be set, but the interrupt will not occur. 1 = Enable. Interrupt will occur.	R/W
2	Last Valid Buffer Interrupt Enable (LVBIE)	This bit controls whether the completion of the last valid buffer will cause an interrupt or not. 0 = Disable. Bit 2 in the Status register will still be set, but the interrupt will not occur. 1 = Enable.	R/W
1	Reset Registers (RR)	0 = Removes reset condition. 1 = Contents of all Bus master related registers to be reset, except the interrupt enable bits (bit 4,3,2 of this register). Software needs to set this bit but need not clear it since the bit is self clearing. This bit must be set only when the Run/Pause bit is cleared. Setting it when the Run bit is set will cause undefined consequences.	R/W (special)
0	Run/Pause Bus master (RPBM)	0 = Pause bus master operation. This results in all state information being retained (i.e., master mode operation may be stopped and then resumed). 1 = Run. Bus master operation starts.	R/W

13.2.8 GLOB_CNT—Global Control Register

Note: Reads across dWord boundaries are not supported.

Table 492. GLOB_CNT—Global Control Register (Sheet 1 of 2)

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 2Ch	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:30	S/PDIF Slot Map (SSM)	When the run/pause bus master bit (bit '0' of offset 2Bh) is set, the value in these bits indicates which slots S/PDIF data is transmitted on. Software must ensure that the programming here does not conflict with the PCM channels being used. When there is a conflict, unpredictable behavior will result; the hardware will not check for a conflict. 00 - Reserved 01 - Slots 7 and 8 10 - Slots 6 and 9 11 - Slots 10 and 11	R/W
29:24	Reserved	Reserved.	
23:22	PCM Out Mode (POM)	Enables the PCM out channel to use 16 or 20-bit audio on PCM out. This does not affect the microphone of S/PDIF DMA. When greater than 16-bit audio is used, the data structures are aligned as 32 bits per sample, with the highest order bits representing the data and the lower order bits as "do not care". 00 = 16 bit audio (default) 01 = 20 bit audio 10 = Reserved. When set, indeterminate behavior will result. 11 = Reserved. When set, indeterminate behavior will result.	R/W
21:20	PCM 4/6 Enable	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	R/W
19:7	Reserved	Reserved.	
6	AC_SDIN2 Interrupt Enable (S2RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[2] causes a resume event on the AC-link.	R/W
5	AC_SDIN1 Resume Interrupt Enable (S1RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[1] causes a resume event on the AC-link.	R/W
4	AC_SDIN0 Resume Interrupt Enable (S0RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[0] causes a resume event on the AC-link.	R/W
3	ACLINK Shut Off (LSO)	0 = Normal operation. 1 = Controller disables all outputs which will be pulled low by internal pull down resistors.	R/W

Table 492. GLOB_CNT—Global Control Register (Sheet 2 of 2)

Device:	31	***Function:***	5
I/O Address:	NABMBAR + 2Ch	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
2	AC'97 Warm Reset	0 = Normal operation. 1 = Writing a '1' to this bit causes a warm reset to occur on the AC-link. The warm reset will awaken a suspended codec without clearing its internal registers. When software attempts to perform a warm reset while bit_clk is running, the write will be ignored and the bit will not change. This bit is self-clearing; it remains set until the reset completes and bit_clk is seen on the ACLink, after which it clears itself.	R/W (special)
1	AC'97 Cold Reset#	0 = Writing a '0' to this bit causes a cold reset to occur throughout the AC'97 circuitry. All data in the controller and the codec will be lost. Software must clear this bit no sooner than the minimum number of ms have elapsed. 1 = This bit defaults to '0' and hence, after reset, the driver needs to set this bit to a '1'. The value of this bit is retained after suspends; hence, when this bit is set to a '1' prior to suspending, a cold reset is not generated automatically upon resuming. **NOTE:** This bit is in the Core well.	R/W
0	GPI Interrupt Enable (GIE)	This bit controls whether the change in status of any GPI causes an interrupt. 0 = Bit '0' of the Global Status Register is set, but no interrupt is generated. 1 = The change on value of a GPI causes an interrupt and sets bit '0' of the Global Status Register.	R/W

13.2.9 GLOB_STA—Global Status Register

Note: Reads across dWord boundaries are not supported.

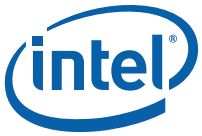


Table 493. GLOB_STA—Global Status Register (Sheet 1 of 3)

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 30h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00700000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:30	Reserved	Reserved.	
29	AC_SDIN2 Resume Interrupt (S2RI)	This bit indicates that a resume event occurred on AC_SDIN[2]. 0 = Cleared by writing a '1' to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
28	AC_SDIN2 Codec Ready (S2CR)	Reflects the state of the codec ready bit in AC_SDIN[2]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is "ready", it must never go "not ready" spontaneously. 0 = Not Ready. 1 = Ready.	RO
27	Bit Clock Stopped (BCS)	Indicates that the bit clock is not running. This bit is set when the Intel® 6300ESB ICH detects that there has been no transition on BIT_CLK for four consecutive PCI clocks. It is cleared when a transition is found on BIT_CLK.	RO
26	S/PDIF Interrupt (SPINT)	Indicates that the S/PDIF out channel interrupt status bits have been set. When the specific status bit is cleared, this bit will be cleared.	RO
25	PCM In 2 Interrupt (P2INT)	Indicates that one of the PCM In 2 channel status bits have been set. When the specific status bit is cleared, this bit will be cleared.	RO
24	Microphone 2 In Interrupt (M2INT)	Indicates that one of the Mic in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
23:22	Sample Capabilities	Indicates the capability to support greater than 16-bit audio. 00 = Reserved 01 = 16 and 20-bit Audio supported (Intel® 6300ESB ICH value) 10 = Reserved 11 = Reserved	RO
21:20	Multichannel Capabilities	Indicates the capability to support more 4 and 6 channels on PCM Out.	RO
19:18	Reserved	Reserved.	
17	MD3	Power down semaphore for Modem. This bit exists in the suspend well and maintains context across power states (except G3). The bit has no hardware function. It is used by software in conjunction with the AD3 bit to coordinate the entry of the two codecs into D3 state. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/W

Table 493. GLOB_STA—Global Status Register (Sheet 2 of 3)

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 30h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00700000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
16	AD3	Power down semaphore for Audio. This bit exists in the suspend well and maintains context across power states (except G3). The bit has no hardware function. It is used by software in conjunction with the MD3 bit to coordinate the entry of the two codecs into D3 state. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/W
15	Read Completion Status (RCS)	This bit indicates the status of codec read completions. 0 = A codec read completes normally. 1 = A codec read results in a time-out. The bit remains set until being cleared by software writing a “1” to the bit location. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
14	Bit 3 of slot 12	Display bit 3 of the most recent slot 12.	RO
13	Bit 2 of slot 12	Display bit 2 of the most recent slot 12.	RO
12	Bit 1 of slot 12	Display bit 1 of the most recent slot 12.	RO
11	AC_SDIN1 Resume Interrupt (S1RI)	This bit indicates that a resume event occurred on AC_SDIN[1]. 0 = Cleared by writing a ’1’ to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
10	AC_SDIN0 Resume Interrupt (S0RI)	This bit indicates that a resume event occurred on AC_SDIN[0]. 0 = Cleared by writing a ’1’ to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
9	AC_SDIN1 Codec Ready (S1CR)	Reflects the state of the codec ready bit in AC_SDIN[1]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is “ready”, it must never go “not ready” spontaneously. 0 = Not Ready. 1 = Ready.	RO
8	AC_SDIN0 Codec Ready (S0CR)	Reflects the state of the codec ready bit in AC_SDIN [0]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is “ready”, it must never go “not ready” spontaneously. 0 = Not Ready. 1 = Ready.	RO
7	Mic In Interrupt (MINT)	This bit indicates that one of the Mic in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO

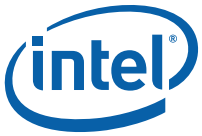


Table 493. GLOB_STA—Global Status Register (Sheet 3 of 3)

Device:	31	**Function:**	5
I/O Address:	NABMBAR + 30h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00700000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
6	PCM Out Interrupt (POINT)	This bit indicates that one of the PCM out channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
5	PCM In Interrupt (PIINT)	This bit indicates that one of the PCM in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
4:3	Reserved	Reserved.	
2	Modem Out Interrupt (MOINT)	This bit indicates that one of the modem out channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
1	Modem In Interrupt (MIINT)	This bit indicates that one of the modem interrupt status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
0	GPI Status Change Interrupt (GSCI)	This bit reflects the state of bit '0' in slot 12 and is set whenever bit '0' of slot 12 is set. This indicates that one of the GPIs changed state and that the new values are available in slot 12. The bit is cleared by software writing a '1' to this bit location. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC

13.2.10 CAS—Codec Access Semaphore Register

Note: Reads across dWord boundaries are not supported.

Table 494. CAS—Codec Access Semaphore Register

Device:	31	***Function:***	5
Offset:	NABMBAR + 34h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:1	Reserved	Reserved.	
0	Codec Access Semaphore (CAS)	This bit is read by software to check whether a codec access is currently in progress. 0 = No access in progress. 1 = The act of reading this register sets this bit to 1. The driver that read this bit may then perform an I/O access. Once the access is completed, hardware automatically clears this bit.	R/W (special)

13.2.11 SDM—SDATA_IN Map Register

Note: Reads across dWord boundaries are not supported.

Table 495. SDM—SDATA_IN Map Register (Sheet 1 of 2)

Device:	31	***Function:***	5
Offset:	NABMBAR + 80h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:6	PCM In 2, Microphone In 2 Data In Line (DI2L)	When the SE bit is set, these bits indicates which AC_SDIN line should be used by the hardware for decoding the input slots for PCM In 2 and Microphone In 2. When the SE bit is cleared, the value of these bits is irrelevant, and PCM In 2 and Mic In 2 DMA engines are not available. 00 AC_SDIN0 01 AC_SDIN1 10 AC_SDIN2 11 Reserved	R/W
5:4	PCM In 1, Microphone In 1 Data In Line (DI1L)	When the SE bit is set, these bits indicates which AC_SDIN line should be used by the hardware for decoding the input slots for PCM In 1 and Microphone In 1. When the SE bit is cleared, the value of these bits is irrelevant, and the PCM In 1 and Mic In 1 engines use the OR'd AC_SDIN lines. 00 AC_SDIN0 01 AC_SDIN1 10 AC_SDIN2 11 Reserved	R/W

Table 495. SDM—SDATA_IN Map Register (Sheet 2 of 2)

Device:	31	**Function:**	5
Offset:	NABMBAR + 80h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
3	Steer Enable (SE)	When set, the AC_SDIN lines are treated separately and not OR'd together before being sent to the DMA engines. When cleared, the AC_SDIN lines are OR'd together, and the Microphone In 2 and PCM In 2 DMA engines are not available.	R/W
2	Reserved	Reserved.	RO
1:0	Last Codec Read Data Input (LDI)	When a codec register is read, this indicates which AC_SDIN the read data returned on. Software may use this to determine how the codecs are mapped. The values are: 00 AC_SDIN0 01 AC_SDIN1 10 AC_SDIN2 11 Reserved	RO

November 2007
Order Number: 300641-004US

AC'97 Modem Controller Registers (D31:F6) 14

14.1 AC'97 Modem PCI Configuration Space (D31:F6)

Note: Registers that are not shown should be treated as Reserved.

Table 496. PCI Configuration Map (Modem—D31:F6)

Offset	Mnemonic	Register	Default	Access
00-01h	VID	Vendor Identification	8086h	RO
02-03h	DID	Device Identification	25A7h	RO
04-05h	PCICMD	PCI Command	0000h	R/W
06-07h	PCISTA	PCI Device Status	0290h	R/WC
08h	RID	Revision Identification	See Note 1	RO
09h	PI	Programming Interface	00h	RO
0Ah	SCC	Sub Class Code	03h	RO
0Bh	BCC	Base Class Code	07h	RO
0Eh	HEDT	Header Type	00h	RO
10-13h	MMBAR	Modem Mixer Base Address	00000001h	R/W
14-17h	MBAR	Modem Base Address	00000001h	R/W
2C-2Dh	SVID	Subsystem Vendor ID	0000h	Write-Once
2E-2Fh	SID	Subsystem ID	0000h	Write-Once
34h	CAP_PTR	Capabilities Pointer	50h	RO
3Ch	INTR_LN	Interrupt Line	00h	RO
3Dh	INT_PN	Interrupt Pin	02h	RO
50-51h	PID	PCI Power Management ID	0001h	RO
52-53h	PC	PC - Power Management Capabilities	C9C2h	RO
54-55h	PCS	Power Management Control and Status	0000h	R/W

NOTES:
1. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

Note: Internal reset as a result of $D3_{HOT}$ to D0 transition will reset all the core well registers except the following BIOS programmed registers as BIOS may not be invoked following the D3-to-D0 transition. Resume well registers will not be reset by the $D3_{HOT}$ to D0 transition.

Core Well registers **not** reset by the $D3_{HOT}$ to D0 transition:

- Offset 2Ch-2Dh – Subsystem Vendor ID (SVID)
- Offset 2Eh-2Fh – Subsystem ID (SID)

Resume Well registers will **not** be reset by the $D3_{HOT}$ to D0 transition:

- Offset 54h-55h – Power Management Control and Status (PCS)

14.1.1 Offset 00 - 01h: VID—Vendor Identification Register (Modem—D31:F6)

Table 497. Offset 00 - 01h: VID—Vendor Identification Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	01 - 01h	***Attribute:***	Read-Only
Default Value:	8086	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Vendor ID Value	16-bit field indicating the company vendor as Intel	RO

14.1.2 Offset 02 - 03h: DID—Device Identification Register (Modem—D31:F6)

Table 498. Offset 02 - 03h: DID—Device Identification Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	02 - 03h	***Attribute:***	Read-Only
Default Value:	25A7h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Device ID Value	Indicates the device number assigned by the SIG	RO

14.1.3 Offset 04 - 05h: PCICMD—PCI Command Register (Modem—D31:F6)

Note: PCICMD is a 16-bit control register. Refer to the PCI 2.2 specification for complete details on each bit.PCISTA—Device Status Register (Modem—D31:F6).

Table 499. Offset 04 - 05h: PCICMD—PCI Command Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	04 - 05h	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:10	Reserved	Reserved. Read 0.	
9	Fast Back-to-Back Enable (FBE)	Not implemented. Hardwired to ‘0’.	
8	SERR# Enable (SEN)	Not implemented. Hardwired to ‘0’.	
7	Wait Cycle Control (WCC)	Not implemented. Hardwired to ‘0’.	
6	Parity Error Response (PER)	Not implemented. Hardwired to ‘0’.	
5	VGA Palette Snoop (VPS)	Not implemented. Hardwired to ‘0’.	
4	Memory Write and Invalidate Enable (MWI)	Not implemented. Hardwired to ‘0’.	
3	Special Cycle Enable (SCE)	Not implemented. Hardwired to ‘0’.	
2	Bus Master Enable (BME)	Controls standard PCI bus mastering capabilities. 0 = Disable 1 = Enable	R/W
1	Memory Space (MS)	Hardwired to ‘0’; AC ‘97 does not respond to memory accesses.	
0	I/O Space (IOS)	This bit controls access to the I/O space registers. 0 = Disable access (default = 0). 1 = Enable access to I/O space. The Native PCI Mode Base Address register should be programmed prior to setting this bit.	R/W

14.1.4 Offset 06 - 07h: PCISTA—Device Status Register (Modem—D31:F6)

Note: PCISTA is a 16-bit status register. Refer to the PCI 2.2 specification for complete details on each bit.

Table 500. Offset 06 - 07h: PCISTA—Device Status Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	06 - 07h	***Attribute:***	Read/Write Clear
Default Value:	0290h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15	DPE (Detected Parity Error)	Not implemented. Hardwired to ‘0’.	RO
14	SERRS (SERR# Status)	Not implemented. Hardwired to ‘0’.	RO
13	MAS (Master-Abort Status)	0 = Software clears this bit by writing a ‘1’ to the bit position. 1 = Bus Master AC ‘97 interface function, as a master, generates a master abort.	R/WC
12	Reserved	Reserved. Read as ‘0’.	RO
11	STA (Signaled Target-Abort Status)	Not implemented. Hardwired to ‘0’.	RO
10:9	DEVT (DEVSEL# Timing Status)	This 2-bit field reflects the Intel® 6300ESB ICH's DEVSEL# timing parameter. These read-only bits indicate the Intel® 6300ESB ICH's DEVSEL# timing when performing a positive decode.	RO
8	DPD (Data Parity Detected)	Not implemented. Hardwired to ‘0’.	RO
7	FBC (Fast Back to back Capable)	Hardwired to ‘1’. This bit indicates that the Intel® 6300ESB ICH as a target is capable of fast back-to-back transactions.	RO
6	UDF Supported	Not implemented. Hardwired to ‘0’.	RO
5	66 MHz Capable	Hardwired to ‘0’.	RO
4	Capabilities List Exists (CLIST)	Indicates that the controller contains a capabilities pointer list. The first item is pointed to by looking at configuration offset 34h.	RO
3:0	Reserved	Reserved.	

14.1.5 Offset 08h: RID—Revision Identification Register (Modem—D31:F6)

Table 501. Offset 08h: RID—Revision Identification Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

14.1.6 Offset 09h: PI—Programming Interface Register (Modem—D31:F6)

Table 502. Offset 09h: PI—Programming Interface Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	09h	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Programming Interface Value		RO

14.1.7 Offset 0Ah: SCC—Sub Class Code Register (Modem—D31:F6)

Table 503. Offset 0Ah: SCC—Sub Class Code Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	03h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Sub Class Code Value	03h = Generic Modem.	RO

14.1.8 Offset 0Bh: BCC—Base Class Code Register (Modem—D31:F6)

Table 504. Offset 0Bh: BCC—Base Class Code Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	07h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Base Class Code Value	07h = Simple Communications Controller.	RO

14.1.9 Offset 0Eh: HEDT—Header Type Register (Modem—D31:F6)

Table 505. Offset 0Eh: HEDT—Header Type Register (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	0Eh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	Header Value		RO

14.1.10 Offset 10 - 13h: MMBAR—Modem Mixer Base Address Register (Modem—D31:F6)

Note: The Native PCI Mode Modem uses PCI Base Address register 1 to request a contiguous block of
I/O space that is to be used for the Modem Mixer software interface. The mixer requires 256 bytes of I/O space. All accesses to the mixer registers are forwarded over the AC-link to the codec where the registers reside.

In the case of split codec implementation, accesses to the different codecs are differentiated by the controller by using address offsets 00h - 7Fh for the primary codec and address offsets 80h - FEh for the secondary codec.

Table 506. Offset 10 - 13h: MMBAR—Modem Mixer Base Address Register (Modem—D31:F6)

Device:	31	**Function:**	6
Offset:	10 - 13h	**Attribute:**	Read/Write
Default Value:	00000001h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:16	Reserved	Hardwired to '0'.	
15:8	Base Address	These bits are used in the I/O space decode of the Modem interface registers. The number of upper bits that a device actually implements depends on how much of the address space the device will respond to. For the AC '97 Modem, the upper 16 bits are hardwired to 0, while bits 15:8 are programmable. This configuration yields a maximum I/O block size of 256 bytes for this base address.	R/W
7:1	Reserved	Read as 0.	
0	Resource Type Indicator (RTE)	This bit is set to one, indicating a request for I/O space.	RO

14.1.11 Offset 14 - 17h: MBAR—Modem Base Address Register (Modem—D31:F6)

Note: The Modem function uses PCI Base Address register 1 to request a contiguous block of I/O space that is to be used for the Modem software interface. The Modem Bus Mastering register space requires 128 bytes of I/O space. All Modem registers reside in the controller; therefore cycles are **not** forwarded over the AC-link to the codec.

Table 507. Offset 14 - 17h: MBAR—Modem Base Address Register (Modem—D31:F6)

Device:	31	**Function:**	6
Offset:	14 - 17h	**Attribute:**	Read/Write
Default Value:	00000001h	**Size:**	32-bit

Bits	Name	Description	Access
31:16	Reserved	Hardwired to '0'.	
15:7	Base Address	These bits are used in the I/O space decode of the Modem interface registers. The number of upper bits that a device actually implements depends on how much of the address space the device will respond to. For the AC '97 Modem, the upper 16 bits are hardwired to '0', while bits 15:7 are programmable. This configuration yields a maximum I/O block size of 128 bytes for this base address.	R/W
6:1	Reserved	Reserved. Read as '0'.	
0	Resource Type Indicator (RTE)	This bit is set to '1', indicating a request for I/O space.	RO

14.1.12 Offset 2C - 2Dh: SVID—Subsystem Vendor ID (Modem—D31:F6)

Note: This register should be implemented for any function that could be instantiated more than once in a given system. For example, a system with two audio subsystems, one on the motherboard and the other plugged into a PCI expansion slot, should have the SVID register implemented. The SVID register, in combination with the Subsystem ID register, enable the operating environment to distinguish one audio subsystem from the other(s).

Note: Software (BIOS) will write the value to this register. After that, the value may be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SID to create one 32-bit write. This register is not affected by D3HOT to D0 reset.

Table 508. Offset 2C - 2Dh: SVID—Subsystem Vendor ID (Modem—D31:F6)

Device:	31	***Function:***	6
Offset:	2C - 2Dh	***Attribute:***	Write-Once
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	Reserved	Hardwired to ‘0’s.	
15:7	Base Address	These bits are used in the I/O space decode of the Modem interface registers. The number of upper bits that a device actually implements depends on how much of the address space the device will respond to. For the AC ‘97 Modem, the upper 16 bits are hardwired to ‘0’, while bits 15:7 are programmable. This configuration yields a maximum I/O block size of 128 bytes for this base address.	R/W
6:1	Reserved	Reserved. Read as ‘0’.	RO
0	Resource Type Indicator (RTE)	This bit is set to ‘1’, indicating a request for I/O space.	RO

14.1.13 Offset 2E - 2Fh: SID—Subsystem ID (Modem—D31:F6)

Note: This register should be implemented for any function that could be instantiated more than once in a given system; for example, a system with two audio subsystems, one on the motherboard and the other plugged into a PCI expansion slot. The SID register, in combination with the Subsystem Vendor ID register, make it possible for the operating environment to distinguish one audio subsystem from the other(s).

Note: Software (BIOS) will write the value to this register. After that, the value may be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SVID to create one 32-bit write. This register is not affected by D3HOT to D0 reset.

Table 509. Offset 2E - 2Fh: SID—Subsystem ID (Modem—D31:F6)

Device:	31	**Function:**	6
Offset:	2E-2Fh	**Attribute:**	Write-Once
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Subsystem ID Value		R/WO

14.1.14 Offset 34h: CAP_PTR—Capabilities Pointer (Modem—D31:F6)

Note: This register indicates the offset for the capability pointer.

Table 510. Offset 34h: CAP_PTR—Capabilities Pointer (Modem—D31:F6)

Device:	31	**Function:**	6
Offset:	34h	**Attribute:**	Read-Only
Default Value:	50h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Capability Pointer (CP)	Indicates that the first capability pointer offset is offset 50h.	RO

14.1.15 Offset 3Ch: INTR_LN—Interrupt Line Register (Modem—D31:F6)

Note: This register indicates which PCI interrupt line is used for the AC'97 module interrupt.

Table 511. Offset 3Ch: INTR_LN—Interrupt Line Register (Modem—D31:F6)

Device:	31	**Function:**	6
Offset:	3Ch	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is used to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

14.1.16 Offset 3Dh: INT_PIN—Interrupt Pin (Modem—D31:F6)

Note: This register indicates which PCI interrupt pin is used for the AC'97 modem interrupt. The AC'97 interrupt is internally OR'd to the interrupt controller with the PIRQB# signal.

Table 512. Offset 3Dh: INT_PIN—Interrupt Pin (Modem—D31:F6)

Device:	29	***Function:***	5
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	02h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2:0	AC'97 Interrupt Routing	Hardwired to 010b to select PIRQB#.	RO

14.1.17 Offset 50h: PID—PCI Power Management Capability ID Register (Modem—D31:F6)

Table 513. Offset 50h: PID—PCI Power Management Capability ID Register (Modem—D31:F6)

Device:	29	***Function:***	5
Offset:	0001h	***Attribute:***	Read-Only
Default Value:	0001h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:8	Next Capability (NEXT)	Indicates that this is the last item in the list.	RO
7:0	Cap ID (CAP)	Indicates that this pointer is a message signaled interrupt capability.	RO

14.1.18 Offset 52h: PC—Power Management Capabilities Register (Modem—D31:F6)

Table 514. Offset 52h: PC—Power Management Capabilities Register (Modem—D31:F6)

Device:	29	**Function:**	5
Offset:	52h	**Attribute:**	Read-Only
Default Value:	C9C2h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:11	PME_Support	Indicates PME# may be generated from all D states.	RO
10:9	Reserved	Reserved.	RO
8:6	Aux_Current	Reports 375 mA maximum Suspend well current required when in the D3cold state.	RO
5	Device Specific Initialization (DSI)	Indicates that no device-specific initialization is required.	RO
4	Reserved	Reserved.	RO
3	PME Clock (PMEC)	Indicates that PCI clock is not required to generate PME#.	RO
2:0	Version (VS)	Indicates support for Revision 1.1 of the PCI Power Management Specification.	RO

14.1.19 Offset 54h: PCS—Power Management Control and Status Register (Modem—D31:F6)

Note: This register is not affected by the $D3_{HOT}$ to D0 transition.

Table 515. Offset 54h: PCS—Power Management Control and Status Register (Modem—D31:F6) (Sheet 1 of 2)

Device:	29	**Function:**	5
Offset:	54h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
15	PME Status (PMES)	This bit is set when the AC'97 controller would normally assert the PME# signal independent of the state of the PME_En bit. This bit resides in the resume well.	RW/C
14:9	Reserved	Reserved.	RO

Table 515. Offset 54h: PCS—Power Management Control and Status Register (Modem—D31:F6) (Sheet 2 of 2)

Device:	29	**Function:**	5
Offset:	54h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Resume

Bits	Name	Description	Access
8	Power Management Event Enable (PMEE)	When set, and if corresponding PMES is also set, the AC'97 controller sets the AC97_STS bit in the GPE0_STS register.	R/W
7:2	Reserved	Reserved.	RO
1:0	Power State (PS)	This field is used both to determine the current power state of the AC'97 controller and to set a new power state. The values are: 00 – D0 state 01 – not supported 10 – not supported 11 – $D3_{HOT}$ state When in the $D3_{HOT}$ state, the AC'97 controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. When software attempts to write a value of 10b or 01b to this field, the write operation must complete normally. However, the data is discarded and no state change occurs.	R/W

14.2 AC'97 Modem I/O Space (D31:F6)

In the case of split codec implementation, 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 - FEh for the secondary codec. Table 516 shows the register addresses for the modem mixer registers.

Table 516. Intel® 6300ESB I/O Controller Hub Modem Mixer Register Configuration

Register		MMBAR Exposed Registers (D31:F6)
Primary	Secondary	Name
00h:38h	80h:B8h	Intel 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
44h	C4h	*Handset DAC/ADC Rate*
46h	C6h	Line 1 DAC/ADC Level Mute
48h	C8h	*Line 2 DAC/ADC Level Mute*
4Ah	CAh	*Handset DAC/ADC Level Mute*
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	AC'97 Reserved
5Ah	DAh	Vendor Reserved
7Ch	FCh	Vendor ID1
7Eh	FEh	Vendor ID2

NOTES:
1. Registers in italics are for functions not supported by the Intel® 6300ESB ICH.
2. Software should not try to access reserved registers.
3. The Intel® 6300ESB ICH supports a modem codec connected to AC_SDIN[2:0] as long as the Codec ID is 00 or 01. However, the Intel® 6300ESB ICH does not support more than one modem codec. For a complete list of topologies, see the Intel® 875P/E7210/6300ESB Chipset Platform Design Guide.

The Global Control (GLOB_CNT) and Global Status (GLOB_STA) 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. S/W could access these registers as bytes, word, or dWord quantities, but reads must not cross dWord boundaries.

These registers exist 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 naming prefix convention

used is as follows:
MI = Modem in channel
MO = Modem out channel

Table 517 presents the modem registers.

Table 517. Modem Registers

Offset	Mnemonic	Name	Default	Access
00h	MI_BDBAR	Modem In Buffer Descriptor List Base Address Register	00000000 h	R/W
04h	MI_CIV	Modem In Current Index Value Register	00h	R
05h	MI_LVI	Modem In Last Valid Index Register	00h	R/W
06h	MI_SR	Modem In Status Register	0001h	R/W
08h	MI_PICB	Modem In Position In Current Buffer Register	00h	R
0Ah	MI_PIV	Modem In Prefetch Index Value Register	00h	RO
0Bh	MI_CR	Modem In Control Register	00h	R/W
10h	MO_BDBAR	Modem Out Buffer Descriptor List Base Address Register	00000000 h	R/W
14h	MO_CIV	Modem Out Current Index Value Register	00h	RO
15h	MO_LVI	Modem Out Last Valid Register	00h	R/W
16h	MO_SR	Modem Out Status Register	0001h	R/W
18h	MI_PICB	Modem In Position In Current Buffer Register	00h	RO
1Ah	MO_PIV	Modem Out Prefetched Index Register	00h	RO
1Bh	MO_CR	Modem Out Control Register	00h	R/W
3Ch	GLOB_CNT	Global Control	00000000 h	R/W
40h	GLOB_STA	Global Status	00000000 h	RO
44h	ACC_SEMA	Codec Write Semaphore Register	00h	R/W

NOTE: MI = Modem in channel; MO = Modem out channel

Note: Internal reset as a result of $D3_{HOT}$ to D0 transition will reset all the core well registers except the registers shared with the AC'97 Audio Controller (GCR, GSR, CASR). Resume well registers will not be reset by the $D3_{HOT}$ to D0 transition.

Core Well registers and bits **not** reset by the $D3_{HOT}$ to D0 transition:

- Offset 3Ch-3Fh – bits[6:0] Global Control (GLOB_CNT)
- Offset 40h-43h – bits[29,15,11:10] Global Status (GLOB_STA)
- Offset 44h – Codec Access Semaphore Register (CAS)

Resume Well registers and bits will **not** be reset by the $D3_{HOT}$ to D0 transition:

- Offset 40h-43h – bits[17:16] Global Status (GLOB_STA)

14.2.1 x_BDBAR—Buffer Descriptor List Base Address Register

Note: Software may read the register at offset 00h by performing a single 32-bit read from address offset 00h. Reads across dWord boundaries are not supported.

Table 518. x_BDBAR—Buffer Descriptor List Base Address Register

Device:	29	**Function:**	5
I/O Address:	MBAR + 00h (MIBDBAR), MBAR + 10h (MOBDBAR)	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:3	Buffer Descriptor List Base Address[31:3]	These bits represent address bits 31:3. The entries should be aligned on 8-byte boundaries.	R/W
2:0		Hardwired to '0'.	

14.2.2 x_CIV—Current Index Value Register

Note: Software may read the registers at offsets 04h, 05h and 06h simultaneously by performing a single 32-bit read from address offset 04h. Software may also read this register individually by doing a single 8-bit read to offset 04h. Reads across dWord boundaries are not supported.

Table 519. x_CIV—Current Index Value Register

Device:	29	**Function:**	5
I/O Address:	MBAR + 04h (MICIV), MBAR + 14h (MOCIV)	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5		Hardwired to '0'.	
4:0	Current Index Value [4:0]	These bits represent which buffer descriptor within the list of 16 descriptors is being processed currently. As each descriptor is processed, this value is incremented.	RO

14.2.3 x_LVI—Last Valid Index Register

Note: Software may read the registers at offsets 04h, 05h and 06h simultaneously by performing a single, 32-bit read from address offset 04h. Software may also read this register individually by doing a single, 8-bit read to offset 05h. Reads across dWord boundaries are not supported.

Table 520. x_LVI—Last Valid Index Register

Device:	29	***Function:***	5
I/O Address:	MBAR + 05h (MILVI), MBAR + 15h (MOLVI)	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:5	Hardwired to 0	Hardwired to 0.	
4:0	Last Valid Index [4:0]	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.	R/W

14.2.4 x_SR—Status Register

Note: Software may read the registers at offsets 04h, 05h and 06h simultaneously by performing a single, 32-bit read from address offset 04h. Software may also read this register individually by doing a single, 16-bit read to offset 06h. Reads across dWord boundaries are not supported.

Table 521. x_SR—Status Register (Sheet 1 of 2)

Device:	29	***Function:***	5
I/O Address:	MBAR + 06h (MISR), MBAR + 16h (MOSR)	***Attribute:***	Read/Write Clear
Default Value:	0001h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:5	Reserved	Reserved.	
4	FIFO error (FIFOE)	0 = Cleared by writing a '1' to this bit position. 1 = FIFO error occurs. **Modem in:** FIFO error indicates a FIFO overrun. The FIFO pointers don't increment, the incoming data is not written into the FIFO, thereby being lost. **Modem out:** FIFO error indicates a FIFO underrun. The sample transmitted in this case should be the last valid sample. The Intel® 6300ESB ICH will set the FIFOE bit if the under-run or overrun occurs when there are more valid buffers to process.	R/WC
3	Buffer Completion Interrupt Status (BCIS)	0 = Cleared by writing a '1' to this bit position. 1 = 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.	R/WC

Table 521. x_SR—Status Register (Sheet 2 of 2)

Device:	29	**Function:**	5
I/O Address:	MBAR + 06h (MISR), MBAR + 16h (MOSR)	**Attribute:**	Read/Write Clear
Default Value:	0001h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	Last Valid Buffer Completion Interrupt (LVBCI)	0 = Cleared by writing a '1' to this bit position 1 = Set 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 may be cleared by software once this event has been recognized. This event will cause an interrupt if the enable bit in the Control Register is set. The interrupt is cleared when the software clears this bit. In the case of transmits (PCM out, Modem out), this bit is set after the last valid buffer has been fetched (not after transmitting it). In the case of Receives, this bit is set after the data for the last buffer has been written to memory.	R/WC
1	Current Equals Last Valid (CELV)	0 = Hardware clears when controller exits state (i.e., until a new value is written to the LVI register). 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.	RO
0	DMA Controller Halted (DCH)	0 = Running. 1 = Halted. This could happen because of the Start/Stop bit being cleared and the DMA engines are idle, or it could happen once the controller has processed the last valid buffer.	RO

14.2.5 x_PICB—Position in Current Buffer Register

Note: Software may read the registers at the offsets 08h, 0Ah, and 0Bh by performing a 32-bit read from address offset 08h. Software may also read this register individually by doing a single 16-bit read to offset 08h. Reads across dWord boundaries are not supported.

Table 522. x_PICB—Position in Current Buffer Register

Device:	29	**Function:**	5
I/O Address:	MBAR + 08h (MIPICB), MBAR + 18h (MOPICB)	**Attribute:**	Read-Only
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Position In Current Buffer[15:0]	These bits represent the number of samples left to be processed in the current buffer.	RO

November 2007
Order Number: 300641-004US

14.2.6 x_PIV—Prefetch Index Value Register

Note: Software may read the registers at offsets 08h, 0Ah, and 0Bh by performing a 32-bit read from address offset 08h. Software may also read this register individually by doing a single 8-bit read to offset 0Ah. Reads across dWord boundaries are not supported.

Table 523. x_PIV—Prefetch Index Value Register

Device:	29	**Function:**	5
I/O Address:	MBAR + 0Ah (MIPIV), MBAR + 1Ah (MOPIV)	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Position In Current Buffer[15:0]	These bits represent the number of samples left to be processed in the current buffer.	RO

14.2.7 x_CR—Control Register

Note: Software may read the registers at offsets 08h, 0Ah, and 0Bh by performing a 32-bit read from address offset 08h. Software may also read this register individually by doing a single 8-bit read to offset 0Bh. Reads across dWord boundaries are not supported.

Table 524. x_CR—Control Register (Sheet 1 of 2)

Device:	29	**Function:**	5
I/O Address:	MBAR + 0Bh (MICR), MBAR + 1Bh (MOCR)	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:5	Reserved	Reserved.	
4	Interrupt On Completion Enable (IOCE)	This bit controls whether or not an interrupt occurs when a buffer completes with the IOC bit set in its descriptor. 0 = Disable 1 = Enable	R/W
3	FIFO Error Interrupt Enable (FEIE)	This bit controls whether the occurrence of a FIFO error will cause an interrupt or not. 0 = Disable. Bit 4 in the Status Register will be set, but the interrupt will not occur. 1 = Enable. Interrupt will occur.	R/W

Table 524. x_CR—Control Register (Sheet 2 of 2)

Device:	29	**Function:**	5
I/O Address:	MBAR + 0Bh (MICR), MBAR + 1Bh (MOCR)	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	Last Valid Buffer Interrupt Enable (LVBIE)	This bit controls whether the completion of the last valid buffer will cause an interrupt or not. 0 = Disable. Bit 2 in the Status register will still be set, but the interrupt will not occur. 1 = Enable	R/W
1	Reset Registers(RR)	0 = Removes reset condition. 1 = Contents of all registers to be reset, except the interrupt enable bits (bit 4,3,2 of this register). Software must set this bit. It must be set only when the Run/Pause bit is cleared. Setting it when the Run bit is set will cause undefined consequences. This bit is self-clearing (software needs not clear it).	R/W (special)
0	Run/Pause Bus Master (RPBM)	0 = Pause bus master operation. This results in all state information being retained (i.e., master mode operation may be stopped and then resumed). 1 = Run. Bus master operation starts.	R/W

14.2.8 GLOB_CNT—Global Control Register

Note: Reads across dWord boundaries are not supported.

Table 525. GLOB_CNT—Global Control Register (Sheet 1 of 2)

Device:	29	**Function:**	5
I/O Address:	MBAR + 3Ch	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:7	Reserved	Reserved	
6	AC_SDIN2 Interrupt Enable (S2RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[2] causes a resume event on the AC-link.	R/W
5	AC_SDIN1 Resume Interrupt Enable (S1RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[1] causes a resume event on the AC-link.	R/W
4	AC_SDIN0 Resume Interrupt Enable (S0RE)	0 = Disable. 1 = Enable an interrupt to occur when the codec on AC_SDIN[0] causes a resume event on the AC-link.	R/W
3	ACLINK Shut Off (LSO)	0 = Normal operation. 1 = Controller disables all outputs which will be pulled low by internal pull down resistors.	R/W

Table 525. GLOB_CNT—Global Control Register (Sheet 2 of 2)

Device:	29	**Function:**	5
I/O Address:	MBAR + 3Ch	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
2	AC'97 Warm Reset	0 = Normal operation. 1 = Writing a '1' to this bit causes a warm reset to occur on the AC-link. The warm reset will awaken a suspended codec without clearing its internal registers. When software attempts to perform a warm reset while bit_clk is running, the write will be ignored and the bit will not change. This bit is self-clearing; it remains set until the reset completes and bit_clk is seen on the ACLink, after which it clears itself.	R/W (special)
1	AC'97 Cold Reset#	0 = Writing a '0' to this bit causes a cold reset to occur throughout the AC '97 circuitry. All data in the controller and the codec will be lost. Software must clear this bit no sooner than the minimum number of ms have elapsed. 1 = This bit defaults to '0' and hence after reset, the driver needs to set this bit to a '1'. The value of this bit is retained after suspends; hence, when this bit is set to a '1' prior to suspending, a cold reset is not generated automatically upon resuming. **NOTE:** This bit is in the Core well.	R/W
0	GPI Interrupt Enable (GIE)	This bit controls whether the change in status of any GPI causes an interrupt. 0 = Bit '0' of the Global Status Register is set, but no interrupt is generated. 1 = The change on value of a GPI causes an interrupt and sets bit '0' of the Global Status Register.	R/W

14.2.9 GLOB_STA—Global Status Register

Note: On reads from a codec, the controller will give the codec a maximum of four frames to respond, after which, if no response is received, it will return a dummy read completion to the processor (with all 'F's on the data) and also set the Read Completion Status bit in the Global Status Register.

Reads across dWord boundaries are not supported.

Table 526. GLOB_STA—Global Status Register (Sheet 1 of 3)

Device:	29	**Function:**	5
I/O Address:	MBAR + 40h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00300000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:30	Reserved	Reserved.	
29	AC_SDIN2 Resume Interrupt (S2RI)	This bit indicates that a resume event occurred on AC_SDIN[2]. 0 = Cleared by writing a '1' to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
28	AC_SDIN2 Codec Ready (S2CR)	Reflects the state of the codec ready bit in AC_SDIN[2]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is "ready", it must never go "not ready" spontaneously. 0 = Not Ready. 1 = Ready.	RO
27	Bit Clock Stopped (BCS)	Indicates that the bit clock is not running. This bit is set when the Intel® 6300ESB ICH detects that there has been no transition on BIT_CLK for four consecutive PCI clocks. It is cleared when a transition is found on BIT_CLK.	RO
26	S/PDIF Interrupt (SPINT)	Indicates that the S/PDIF out channel interrupt status bits have been set. When the specific status bit is cleared, this bit will be cleared.	RO
25	PCM In 2 Interrupt (P2INT)	Indicates that one of the PCM In 2 channel status bits have been set. When the specific status bit is cleared, this bit will be cleared.	RO
24	Microphone 2 In Interrupt (M2INT)	Indicates that one of the Mic in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
23:22	Sample Capabilities	Indicates the capability to support more greater than 16-bit audio. 00 = Reserved 01 = 16 and 20-bit Audio supported (Intel® 6300ESB ICH value) 10 = Reserved 11 = Reserved	RO
21:20	Multichannel Capabilities	Indicates the capability to support more 4 and 6 channels on PCM Out.	RO
19:18	Reserved	Reserved.	

Table 526. GLOB_STA—Global Status Register (Sheet 2 of 3)

Device:	29	**Function:**	5
I/O Address:	MBAR + 40h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00300000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
17	MD3	Power down semaphore for Modem. This bit exists in the suspend well and maintains context across power states (except G3). The bit has no hardware function. It is used by software in conjunction with the AD3 bit to coordinate the entry of the two codecs into D3 state. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/W
16	AD3	Power down semaphore for Audio. This bit exists in the suspend well and maintains context across power states (except G3). The bit has no hardware function. It is used by software in conjunction with the MD3 bit to coordinate the entry of the two codecs into D3 state. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/W
15	Read Completion Status (RCS)	This bit indicates the status of codec read completions. 0 = A codec read completes normally. 1 = A codec read results in a time-out. The bit remains set until being cleared by software writing a “1” to the bit location. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
14	Bit 3 of slot 12	Display bit 3 of the most recent slot 12.	RO
13	Bit 2 of slot 12	Display bit 2 of the most recent slot 12.	RO
12	Bit 1 of slot 12	Display bit 1 of the most recent slot 12.	RO
11	AC_SDIN1 Resume Interrupt (S1RI)	This bit indicates that a resume event occurred on AC_SDIN[1]. 0 = Cleared by writing a ’1’ to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
10	AC_SDIN0 Resume Interrupt (S0RI)	This bit indicates that a resume event occurred on AC_SDIN[0]. 0 = Cleared by writing a ’1’ to this bit position. 1 = Resume event occurred. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC
9	AC_SDIN1 Codec Ready (S1CR)	Reflects the state of the codec ready bit in AC_SDIN[1]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is “ready”, it must never go “not ready” spontaneously. 0 = Not Ready. 1 = Ready.	RO
8	AC_SDIN0 Codec Ready (S0CR)	Reflects the state of the codec ready bit in AC_SDIN [0]. Bus masters ignore the condition of the codec ready bits, so software must check this bit before starting the bus masters. Once the codec is “ready”, it must never go “not ready” spontaneously. 0 = Not Ready. 1 = Ready.	RO

Table 526. GLOB_STA—Global Status Register (Sheet 3 of 3)

Device:	29	**Function:**	5
I/O Address:	MBAR + 40h	**Attribute:**	Read-Only, Read/Write, Read/Write Clear
Default Value:	00300000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7	Mic In Interrupt (MINT)	This bit indicates that one of the Mic in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
6	PCM Out Interrupt (POINT)	This bit indicates that one of the PCM out channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
5	PCM In Interrupt (PIINT)	This bit indicates that one of the PCM in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
4:3	Reserved	Reserved.	RO
2	Modem Out Interrupt (MOINT)	This bit indicates that one of the modem out channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
1	Modem In Interrupt (MIINT)	This bit indicates that one of the modem in channel interrupts status bits has been set. When the specific status bit is cleared, this bit will be cleared.	RO
0	GPI Status Change Interrupt (GSCI)	This bit reflects the state of bit '0' in slot 12, and is set whenever bit '0' of slot 12 is set. This indicates that one of the GPIs changed state, and that the new values are available in slot 12. The bit is cleared by software writing a '1' to this bit location. This bit is not affected by $D3_{HOT}$ to D0 Reset.	R/WC

14.2.10 CAS—Codec Access Semaphore Register

Note: Reads across dWord boundaries are not supported.

Table 527. CAS—Codec Access Semaphore Register

Device:	29	**Function:**	5
I/O Address:	NABMBAR + 44h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:1	Reserved	Reserved.	
0	Codec Access Semaphore (CAS)	This bit is read by software to check whether a codec access is currently in progress. 0 = No access in progress. 1 = The act of reading this register sets this bit to 1. The driver that read this bit may then perform an I/O access. Once the access is completed, hardware automatically clears this bit.	R/W (special)

Multimedia Timer Registers 15

15.1 Memory Mapped Registers

The timer registers are memory mapped in a non-indexed scheme. This allows the processor to directly access each register without having to use an index register. The timer register space is 1024 bytes. The registers are generally aligned on 64-bit boundaries to simplify implementation with IA64 processors. There are four possible memory address ranges beginning at
1) FED0_0000h, 2) FED0_1000h, 3) FED0_2000h., 4) FED0_3000h. The choice of address range will be selected by configuration bits in General Control register (offset D0h) in Device 31, Function 0.

15.1.1 Behavioral Rules

1. Software must not attempt to read or write across register boundaries. For example, a 32-bit access should be to offset x0h, x4h, x8h, or xCh. 32-bit accesses should not be to 01h, 02h, 03h, 05h, 06h, 07h, 09h, 0Ah, 0Bh, 0Dh, 0Eh, or 0Fh. Any accesses to these offsets will result in unexpected behavior and may result in a master abort. However, these accesses will not result in system hangs. 64-bit accesses may only be to x0h and must not cross 64-bit boundaries.
2. Software should not write to read-only registers.
3. Reads or writes to unimplemented timers should not be attempted. Timers 3:31 are not implemented.
4. All registers are implemented in the Core Well, and all bits are reset by PXPCIRST#.
5. Reads to reserved registers or bits will return a value of ‘0’.
6. Software must not attempt locks to the memory mapped I/O ranges for Multimedia Timers. When attempted, the lock is not honored, which means potential deadlock conditions may occur.

Table 528. Memory-Mapped Registers (Sheet 1 of 2)

Offset	Register	Type
000-007h	General Capabilities and ID	Read Only
008-00Fh	Reserved	
010-017h	General Config	Read-Write
018-01Fh	Reserved	
020-027h	General Interrupt Status	Read/Write Clear
028-0EFh	Reserved	
0F0-0F7h	Main Counter Value	Read/Write
0F8-0FFh	Reserved	
100-107h	Timer 0 Config and Capabilities	Read/Write
108-10Fh	Timer 0 Comparator Value	Read/Write
110-11Fh	Reserved	
120-127h	Timer 1 Config and Capabilities	Read/Write
128-12Fh	Timer 1 Comparator Value	Read/Write

Table 528. Memory-Mapped Registers (Sheet 2 of 2)

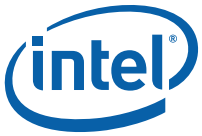
Offset	Register	Type
130-13Fh	Reserved	
140-147h	Timer 2 Config and Capabilities	Read/Write
148-14Fh	Timer 2 Comparator Value	Read/Write
150-15Fh	Reserved	
160-3FFh	Reserved	

15.1.2 Offset 000-007h: General Capabilities and ID Register

Table 529. Offset 000-007h: General Capabilities and ID Register

Offset:	000-007h	***Attribute:***	Read-Only
Default Value:	0429 B17F 8086 A201h	***Size:***	64-bit

Bits	Name	Description	Access
63:32	COUNTER_CLK_PER_CAP	Main Counter Tick Period: This read-only field indicates the period at which the counter increments in femptoseconds (10^-15 seconds). This will return 0429B17F when read. This indicates a period of 69841279 fs (69.841279 ns).	RO
31:16	VENDOR_ID_CAP	This is a 16-bit value assigned to Intel. These bits will return 8086h when read.	RO
15	LEG_RT_CAP	Legacy Rout Capable: This bit will return a '1' when read, indicating that the Legacy Interrupt Rout option is supported.	RO
14	Reserved	Reserved. This bit will return '0' when read.	
13	COUNT_SIZE_CAP	Counter Size: This bit will return a '1' when read, indicating that the counter is 64-bit wide.	RO
12:8	NUM_TIM_CAP	Number of Timers: This indicates the number of timers in this block. The value in this field is 02h, indicating that there are three timers.	RO
7:0	REV_ID	This indicates which revision of the function is implemented. Default value will be 01h.	RO



15.1.3 Offset 010-017h: General Config Register

Table 530. Offset 010-017h: General Config Register

Offset:	010-017h	***Attribute:***	Read/Write
Default Value:	0000 0000 0000 0000h	***Size:***	64-bit

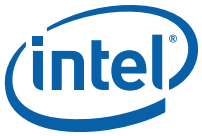
Bits	Name	Description	Access
63:2	Reserved	Reserved. These bits will return '0' when read.	
1	LEG_RT_CNF	Legacy Rout: When the ENABLE_CNF bit and the LEG_RT_CNF bit are both set, the interrupts will be routed as follows: • Timer 0 will be routed to IRQ0 in 8259 or IRQ2 in the I/O APIC. • Timer 1 will be routed to IRQ8 in 8259 or IRQ8 in the I/O APIC. • Timer 2-n will be routed as per the routing in the timer n config registers. • When the Legacy Rout bit is set, the individual routing bits for Timers 0 and 1 (APIC) will have no impact. • When the Legacy Rout bit is not set, the individual routing bits for each of the timers are used. • This bit will default to '0'. BIOS may set it to '1' to enable the legacy routing or '0' to disable the legacy routing.	R/W
0	ENABLE_CNF: Overall Enable	This bit must be set to enable any of the timers to generate interrupts. When this bit is '0', the main counter will halt (will not increment) and no interrupts will be caused by any of these timers. For level-triggered interrupts, if an interrupt is pending when the ENABLE_CNF bit is changed from '1' to '0', the interrupt status indications in the various Txx_INT_STS bits will not be cleared. Software must write to the Txx_INT_STS bits to clear the interrupts. **NOTE:** This bit will default to '0'. BIOS may set it to '1' or '0'.	R/W

15.1.4 Offset 020-027h: General Interrupt Status Register

Table 531. Offset 020-027h: General Interrupt Status Register

Offset:	020-027h	***Attribute:***	Read/Write
Default Value:	0000 0000 0000 0000h	***Size:***	64-bit

Bits	Name	Description	Access
63:3	Reserved	Reserved. These bits will return '0' when read.	
2	T02_INT_STS: Timer 2 Interrupt Active	Same functionality as Timer 0.	R/W
1	T01_INT_STS: Timer 1 Interrupt Active	Same functionality as Timer 0.	R/W
0	T00_INT_STS: Timer 0 Interrupt Active	The functionality of this bit depends on whether the edge or level-triggered mode is used for this timer: **When set to level-triggered mode:** This bit defaults to '0'. This bit will be set by hardware when the corresponding timer interrupt is active. Once the bit is set, it may be cleared by software writing a '1' to the same bit position. Writes of '0' to this bit will have no effect. For example, if the bit is already set, a write of '0' will not clear the bit. **When set to edge-triggered mode:** This bit should be ignored by software. Software should always write '0' to this bit. **NOTE:** Defaults to '0'. In edge-triggered mode, this bit will always read as '0' and writes will have no effect.	R/W



15.1.5 Offset 0F0 - 0f7h: Main Counter Value

Table 532. Offset 0F0 - 0f7h: Main Counter Value

Offset:	0F0-0f7h	***Attribute:***	Read/Write
Default Value:	N/A	***Size:***	64-bit

Bits	Name	Description	Access
63:0	COUNTER_VAL[63:0]	Reads return the current value of the counter. Writes load the new value to the counter. **NOTES:** 1. Software can access the various bytes in this register using 32-bit or 64-bit accesses. 32-bit accesses can be done to offset 0F0h or 0F4h. 64-bit accesses can be done to 0F0h. 32-bit accesses must not be done starting at: 0F1h, 0F2h, 0F3h, 0F5h, 0F6h, or 0F7h. 2. Writes to this register should only be done while the counter is halted. 3. Reads to this register return the current value of the main counter. 4. 32-bit counters will always return '0' for the upper 32 bits of this register. 5. If 32-bit software attempts to read a 64-bit counter, it should first halt the counter. Since this will delay the interrupts for all of the timers, this should be done only if the consequences are understood. It is strongly recommended that 32-bit software only operate the timer in 32-bit mode. 6. Reads to this register are monotonic. No two consecutive reads will return the same value. The second of two reads will always return a larger value (unless the timer has rolled over to '0').	R/W

15.1.6 Timer *n* Config and Capabilities

Note: The letter n may be 0, 1, or 2, referring to Timer 0, 1 or 2.

Table 533. Timer n Config and Capabilities (Sheet 1 of 3)

Offset:	Timer 0: 100-107h, Timer 1: 120-127h, Timer 2: 140-147h	***Attribute:***	Read/Write
Default Value:	N/A	***Size:***	64-bit

Bits	Name	Description	Access
64:56	Reserved	Reserved. These bits will return '0' when read.	
63:32	TIMERn_INT_ROUT_CAP : TIMERn_INT_ROUT[31:0]_CAP	This 32-bit read-only field indicates which interrupts in the I/O (x) APIC this timer's interrupt may be routed to. This is used in conjunction with the TIMERn_INT_ROUT_CNF field. Each bit in this field corresponds to a particular interrupt. For example, when this timer's interrupt may be mapped to interrupts 16, 18, 20, 22, or 24, bits 16, 18, 20, 22, and 24 in this field will be set to '1'. All other bits will be '0'. Intel® 6300ESB ICH and Timer 0, 1 Specific: Bits 20, 21, 22, and 23 in this field (corresponding to bits 52, 53, 54, and 55 in this register) will have a value of '1'. All other bits will be '0'. Writes will have no effect. Intel® 6300ESB ICH and Timer 2 Specific: Bits 11, 20, 21, 22, and 23 in this field (corresponding to bits 43, 52, 53, 54, and 55 in this register) will have a value of '1'. All other bits will be '0'. Writes will have no effect. When IRQ 11 is used for MMT#2, software should ensure IRQ11 is not shared with any other devices to ensure the proper operation of MMT#2.	
31:16	Reserved	Reserved. These bits will return '0' when read.	
15	TIMERn_FSB_INT_DEL_CAP: FSB Interrupt Delivery	(where n is the timer number: 00 to 31) If this read-only bit is 1, then the hardware supports a direct processor side bus delivery of this timer's interrupt. **NOTE:** This bit will always read as 0, since the Intel® 6300ESB ICH Multimedia Timer implementation does not support the direct FSB interrupt delivery.	
14	TIMERn_FSB_EN_CNF	(where n is the timer number: 00 to 31). If the TIMERn_FSB_INT_DEL_CAP bit is set for this timer, then the software can set the TIMERn_FSB_EN_CNF bit to force the interrupts to be delivered directly as FSB messages, rather than using the I/O (x) APIC. In this case, the TIMERn_INT_ROUT_CNF field in this register will be ignored. The TIMERn_FSB_ROUT register will be used instead.	

Table 533. Timer n Config and Capabilities (Sheet 2 of 3)

Offset:	Timer 0: 100-107h, Timer 1: 120-127h, Timer 2: 140-147h	***Attribute:***	Read/Write
Default Value:	N/A	***Size:***	64-bit

Bits	Name	Description	Access
13:9	TIMERn_INT_ROUT_CNF	Interrupt Rout: This 5-bit field indicates the routing for the interrupt to the I/O (x) APIC. A maximum value of 32 interrupts is supported; default is 00h. Software writes to this field to select which interrupt in the I/O (x) will be used for this timer's interrupt. When the value is not supported by this particular timer, the value read back will not match what is written. The software must only write valid values. **NOTES:** 1. When the Legacy Rout bit is set, Timers 0 and 1 will have a different routing, and this bit field has no effect for those two timers. 2. Timer 0,1 Specific: Software is responsible to make sure it programs a valid value (20, 21, 22, or 23) for this field. The Intel® 6300ESB ICH logic does not check the validity of the value written. 3. Timer 2 Specific: Software is responsible to make sure it programs a valid value (11, 20, 21, 22, or 23) for this field. The Intel® 6300ESB ICH logic does not check the validity of the value written.	
8	TIMERn_32MODE_CNF	Timer n 32-bit Mode: Software may set this bit to force a 64-bit timer to behave as a 32-bit timer. For Timer 0, this bit will be read/write and default to '0'. For Timers 1 and 2, this bit will always read as '0', and writes will have no effect, since these two timers are 32 bits.	
7	Reserved	Reserved. This bit will return '0' when read.	
6	TIMERn_VAL_SET_CNF	Timer n Value Set: Software uses this bit only for Timer 0 when it has been set to periodic mode. By writing this bit to a '1', the software is then allowed to directly set the timer's accumulator. Software does NOT have to write this bit back to '0'; it automatically clears. Software should not write a '1' to this bit position when the timer is set to non-periodic mode. **NOTE:** This bit will return '0' when read. Writes will only have an effect for Timer 0 when it is set to periodic mode. Writes will have no effect for Timers 1 and 2.	
5	TIMERn_SIZE_CAP	Timer n Size: This read-only field indicates the size of the timer. Value is '1' (64 bits) for Timer 0. Value is '0' (32 bits) for Timers 1 and 2.	
4	TIMERn_PER_INT_CAP	Periodic Interrupt Capable: When this read-only bit is '1', the hardware supports a periodic mode for this timer's interrupt. Timer 0 will support the periodic interrupt, so the bit will always read as a '1'. Timers 1 and 2 will not support periodic interrupts, so the bit will always read as '0'.	

Table 533. Timer n Config and Capabilities (Sheet 3 of 3)

Offset:	Timer 0: 100-107h, Timer 1: 120-127h, Timer 2: 140-147h	***Attribute:***	Read/Write
Default Value:	N/A	***Size:***	64-bit

Bits	Name	Description	Access
3	TIMERn_TYPE_CNF	Timer n Type: For Timers 1 and 2, this bit will always return '0' when read and writes will have no impact. For Timer 0, this bit is read/write, and may be used to enable the timer to generate a periodic interrupt. Writing a 1 to this bit enables the timer to generate a periodic interrupt. **NOTE:** For timer 0, this bit will be read/write, with default of '0'. For timers 1:2, this bit will be read-only, with a fixed value of '0'.	
2	TIMERn_INT_ENB_CNF	Timer n Interrupt Enable: This bit must be set to enable timer n to cause an interrupt when it times out. When this bit is '0', the timer may still count and generate appropriate status bits but will not cause an interrupt. Default value is '0'.	
1	TIMERn_INT_TYPE_CNF	Timer Interrupt Type: 0 = The timer interrupt is edge triggered. This means that an edge-type interrupt is generated. When another interrupt occurs, another edge will be generated. 1 = The timer interrupt is level triggered. This means that a level-triggered interrupt is generated. The interrupt will be held active until it is cleared by writing to the bit in the General Interrupt Status Register. When another interrupt occurs before the interrupt is cleared, the interrupt will remain active. **NOTE:** The default value is '0', edge-triggered. The interrupt type is not expected to be changed dynamically. The interrupt type for any timer should be set before any interrupts are generated by that timer. If the interrupt type is changed dynamically, there will be some delay before the new type takes effect. That delay is not known at this time. Supports edge and level triggered modes for all three timers.	
0	Reserved	Reserved. These bits will return '0' when read.	

15.1.7 Timer *n* Comparator Value

Table 534. Timer n Comparator Value

Offset:	Timer 0: 108h – 10Fh, Timer 1: 128h – 12Fh, Timer 2: 148h – 14Fh	**Attribute:**	Read/Write
Default Value:	N/A	**Size:**	64-bit

Bits	Name	Description	Access
63:0		1. Reads to this register return the current value of the comparator. 2. When Timers 0,1, or 2 are configured to non-periodic mode: A. Writes to this register load the value against which the main counter will be compared for this timer. B. When the main counter equals the value last written to this register, the corresponding interrupt will be generated (when enabled). C. The value in this register does not change based on the interrupt being generated. 3. When Timer 0 is configured to periodic mode: A. When the main counter equals the value last written to this register, the corresponding interrupt will be generated (when enabled). B. After the main counter equals the value in this register, the value in this register is increased by the value last written to the register. For example, if the value written to the register is 00000123h, then: 1. An interrupt will be generated when the main counter reaches 00000123h. 2. The value in this register will be adjusted by the hardware to 00000246h. 3. Another interrupt will be generated when the main counter reaches 00000246h. 4. The value in this register will be adjusted by the hardware to 00000369h. C. As each periodic interrupt occurs, the value in this register will increment. When the incremented value is greater than the maximum value possible for this register (FFFFFFFFh for a 32-bit timer or FFFFFFFFFFFFFFFFh for a 64-bit timer), the value will wrap around through 0. For example, if the current value in a 32-bit timer is FFFF0000h and the last value written to this register is 20000, then after the next interrupt the value will change to 00010000h. 4. Default value for each timer is all ones for the bits that are implemented. For example, a 32-bit timer will have a default value of 00000000FFFFFFFFh. A 64-bit timer will have a default value of FFFFFFFFFFFFFFFFh.	

November 2007
Order Number: 300641-004US

WatchdogTimer(WDT)(D29:F4)16

16.1 Product Features

The Watchdog Timer (WDT) supports the following features and functions:

- Selectable prescaler – approximately 1 MHz and approximately 1 KHz
- 33 MHz clock (30 ns clock ticks)
- Multiple modes: WDT and free-running
  - Free-running mode:
    - One stage timer.
    - Toggles WDT_OUT# after programmable time.
  - WDT Mode:
    - Two stage timer:
      1. First stage generates IRQ and SMI interrupt after programmable time.
      2. Second stage drives WDT_OUT# low or inverts the previous value.

      *Used only after first timeout occurs.

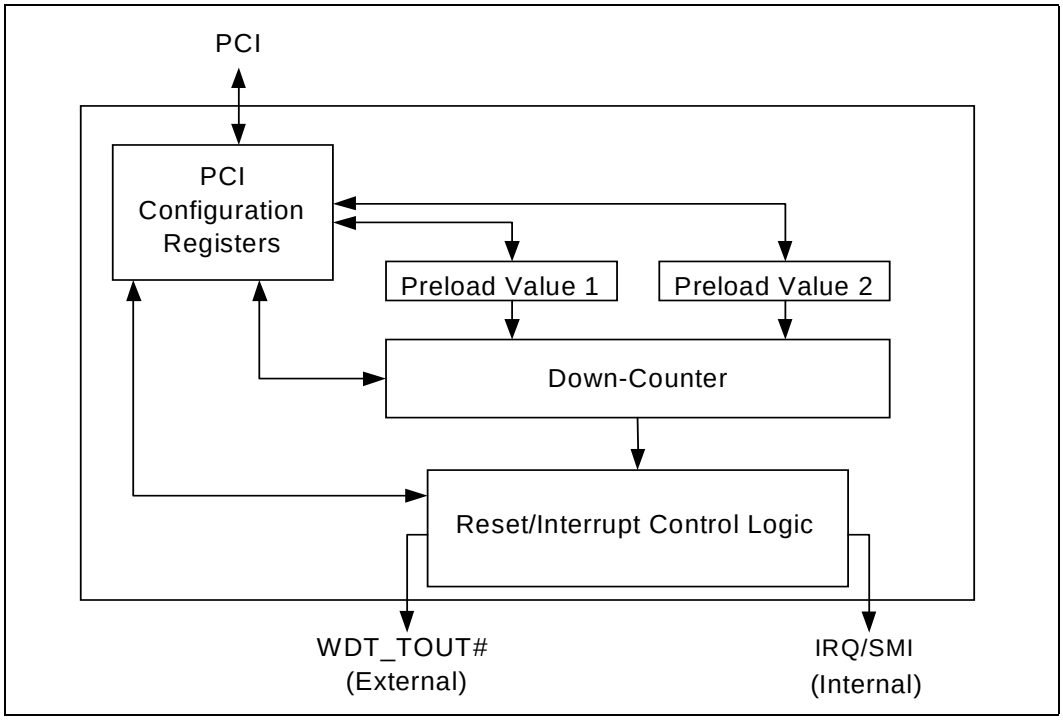
      *Status bit preserved in RTC well for possible error detection and correction.

      *Drives WDT_TOUT# when OUTPUT is enabled.
    - Timer may be disabled (default state) or locked (hard reset required to disable WDT)
    - WDT automatic reload of preload value when WDT reload sequence is performed

Note: The WDT device (Dev 29:F4) cannot be hidden by using bit 12 of the D31:F0 FUNC_DIS Register. The WDT will always be present as a PCI device in PCI Config Space.

16.2 Product Overview

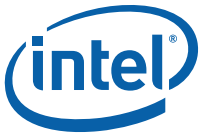
Figure 29. WDT Block Diagram



The timer uses a 35-bit down-counter. The counter is loaded with the value from the first Preload register. The timer is then enabled and starts counting down. The time at which the WDT first starts counting down is called the first stage. When the host fails to reload the WDT before the 35-bit down-counter reaches zero, the WDT generates an internal interrupt. After the interrupt is generated, the WDT loads the value from the second Preload register into the WDT's 35-bit down-counter and starts counting down. The WDT is now in the second stage. When the host still fails to reload the WDT before the second timeout, the WDT drives the WDT_TOUT# pin low and sets the timeout bit (WDT_TIMEOUT). This bit indicates that the System has become unstable. The WDT_TOUT# pin is held low until the system is reset or the WDT times out again (depending on TOUT_CNF). The process of reloading the WDT involves the following sequence of writes:

1. Write 80 to offset BAR + 0Ch.
2. Write 86 to offset BAR + 0Ch.
3. Write 1 to WDT_RELOAD in Reload Register.

The same process is used for setting the values in the preload registers. The only difference exists in step 3. Instead of writing a '1' to the WDT_RELOAD, write the desired preload value into the corresponding Preload register. This value is not loaded into the 35-bit down-counter until the next time the WDT reenters the stage. For example, when Preload Value 2 is changed, it is not loaded into the 35-bit down-counter until the next time the WDT enters the second stage.



16.3 Signal Descriptions

The following signals are driven from the WDT.

Table 535. WDT Interface

Signal	Type	Name and Description
WDT_TOUT# / GPIO[32]	O	**Watchdog Timer Timeout:** The WDT_TOUT# signal is driven low from the Intel® 6300ESB ICH to an external pin. The signal is driven low when the main 35-bit down-counter reaches zero during the second stage. The WDT_TOUT_CNF bit in the Configuration register determines if the output is to change from the previous state when another timeout occurs, or WDT_OUT# is driven low until the system is reset or power is cycled. Driven active to indicate the second stage of the WDT has overflowed. This signal will toggle states for each overflow in periodic mode. In non-periodic mode, this signal will go active low and remain in this state until a system reset or power cycle. This signal is muxed with GPIO[32].
WDT_INT (Internal only signal)	O	**Interrupt:** The WDT_INT# is an internal signal that is used to generate an interrupt when the first stage has been allowed to reach zero. The WDT is capable of generating SCI, SMI, and IRQ (APIC 1, INT 10) based interrupts. Interrupts are not generated when WDT_TOUT_CNF is set to change output after every timeout (See Configuration Register). The WDT INT# signal is an active low interrupt.

16.4 Device 29: Function 4 Configuration Registers

16.4.1 Configuration Registers

Table 536. Configuration Registers (Sheet 1 of 2)

Offset	Register	Default	Type
00-01h	Vendor ID	8086h	Read Only
02-03h	Device ID	25ABh	Read Only
04-05h	Command Register (COM)	0000h	Read/Write
06-07h	Device Status Register (DS)	0280h	Read/Write Clear
08h	Revision ID Register (RID)	See NOTE:	Read Only
09h	Programming Interface Register (PI)	00h	Read Only
0Ah	Sub Class Code Register (SCC)	80h	Read Only
0Bh	Base Class Code Register (BCC)	08h	Read Only
0Eh	Header Type Register (HEDT)	00h	Read Only
10-13h	Base Address Register (BAR)	00000000h	Read/Write

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

Table 536. Configuration Registers (Sheet 2 of 2)

Offset	Register	Default	Type
14-2Bh	Reserved	00h	
2C-2Dh	Subsystem Vendor ID	00h	Read/Write Once
2E-2F	Subsystem ID	00h	Read/Write Once
30-5F	Reserved	00h	
60-61h	WDT Configuration	00h	Read/Write
64-67h	Reserved	0000h	
68h	WDT Lock Register	00h	Read/Write
6C-F7	Reserved	00h	
F8-FBh	Manufacturer's ID	00000F66h	Read Only
FC-FFh	Reserved	00h	

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

16.4.2 Memory Mapped Registers

Table 537. Memory Mapped Registers

Offset	Register	Default	Type
Base + 00h	Preload Value 1	FFFFFh	Read/Write
Base + 04h	Preload Value 2	FFFFFh	Read/Write
Base + 08h	General Interrupt Status	00h	Read/Write/Clear
Base + 0Ch	Reload Register	0000h	Write

16.4.3 Offset 00h: VID—Vendor Identification Register

Table 538. Offset 00h: VID—Vendor Identification Register

Device:	29	***Function:***	4
Offset:	00h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit

Bits	Name	Description	Access
15:0	Vendor ID		

16.4.4 Offset 02h: DID—Device Identification Register

Table 539. Offset 02h: DID—Device Identification Register

Device:	29	***Function:***	4
Offset:	02h	***Attribute:***	Read-Only
Default Value:	25ABh	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Device ID		

16.4.5 Offset 04 - 05h: COM—Command Register

Table 540. Offset 04 - 05h: COM—Command Register

Device:	29	***Function:***	4
Offset:	04 - 05h	***Attribute:***	Read-Only
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

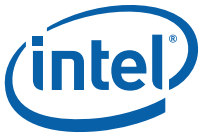
Bits	Name	Description	Access
15:10	Reserved	Reserved.	NA
9	Fast Back-to-Back Enable (FBE)	Reserved as ‘0’.	RO
8	SERR# Enable	Reserved as ‘0’.	RO
7	Wait Cycle Control	Reserved as ‘0’.	RO
6	Parity Error Response	Reserved as ‘0’.	RO
5	VGA Palette Snoop	Reserved as ‘0’.	RO
4	PMWE - Postable Memory Write Enable	Reserved as ‘0’.	RO
3	SCE - Special Cycle Enable	Reserved as ‘0’.	RO
2	BME - Bus Master Enable	Reserved as ‘0’.	RO
1	MSE - Memory Space Enable	This bit controls access to the WDT’s Memory Mapped registers. If this bit is set, accesses to the WDT’s Memory Mapped registers are enabled. The Base Address register for WDT should be programmed before this bit is set.	R/W
0	IOSE - I/O Space Enable	Reserved as ‘0’.	RO

16.4.6 Offset 06h - 07h: DS—Device Status Register

Table 541. Offset 06h - 07h: DS—Device Status Register

Device:	29	***Function:***	4
Offset:	06h - 07h	***Attribute:***	Read/Write Clear
Default Value:	0280h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15	DPE - Detected Parity Error	Reserved as ‘0’.	RO
14	SSE - Signaled System Error	Reserved as ‘0’.	RO
13	RMA - Received Master Abort	Reserved as ‘0’.	RO
12	RTA - Received Target Abort	Reserved as ‘0’.	RO
11	STA - Signaled Target-Abort Status	This bit is set when the function is targeted with a transaction that the Intel® 6300ESB ICH terminates with a target abort. Software resets STA to ’0’ by writing a ’1’ to this bit location.	R/WC
10:9	DEVT - DEVSEL# Timing Status	This 2-bit field defines the timing for DEVSEL# assertion. These read only bits indicate Intel® 6300ESB ICH’s DEVSEL# timing when performing a positive decode. The Intel® 6300ESB ICH generates DEVSEL# with medium time.	RO
8	Data Parity Error Detected	Reserved as ‘0’.	RO
7	Fast Back-to-Back Capable	Reserved as ‘1’.	RO
6	UDF - User Definable Features	Reserved as ‘0’.	RO
5	66 MHz Capable	Reserved as ‘0’.	RO
4:0	Reserved	Reserved.	RO



16.4.7 Offset 08h: RID—Revision Identification Register

Note: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID.

Table 542. Offset 08h: RID—Revision Identification Register

Device:	29	**Function:**	4
Offset:	08h	**Attribute:**	Read-Only
Default Value:	See Note	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15	DPE - Detected Parity Error	Reserved as ‘0’.	RO
14	SSE - Signaled System Error	Reserved as ‘0’.	RO
13	RMA - Received Master Abort	Reserved as ‘0’.	RO
12	RTA - Received Target Abort	Reserved as ‘0’.	RO
11	STA - Signaled Target-Abort Status	This bit is set when the function is targeted with a transaction that Intel® 6300ESB ICH terminates with a target abort. Software resets STA to ’0’ by writing a ’1’ to this bit location.	R/WC
10:9	DEVT - DEVSEL# Timing Status	This two-bit field defines the timing for DEVSEL# assertion. These read-only bits indicate the Intel® 6300ESB ICH’s DEVSEL# timing when performing a positive decode. The Intel® 6300ESB ICH generates DEVSEL# with medium time.	RO
8	Data Parity Error Detected	Reserved as ‘0’.	RO
7	Fast Back-to-Back Capable	Reserved as ‘1’.	RO
6	UDF - User Definable Features	Reserved as ‘0’.	RO
5	66 MHz Capable	Reserved as ‘0’.	RO
4:0	Reserved	Reserved.	RO

16.4.8 Offset 09h: PI—Programming Interface Register

Table 543. Offset 09h: PI—Programming Interface Register

Device:	29	**Function:**	4
Offset:	09h	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Programming Interface		

16.4.9 Offset 0Ah: SCC—Sub Class Code Register

Table 544. Offset 0Ah: SCC—Sub Class Code Register

Device:	29	**Function:**	4
Offset:	0Ah	**Attribute:**	Read-Only
Default Value:	80h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Sub Class Code		

16.4.10 Offset 0Bh: BCC—Base Code Class Register

Table 545. Offset 0Bh: BCC—Base Code Class Register

Device:	29	**Function:**	4
Offset:	0Bh	**Attribute:**	Read-Only
Default Value:	08h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Base Code Class		

16.4.11 Offset 0Eh: HEDT—Header Type Register

Note: The Base Address Register points to several memory mapped registers for the Watchdog Timer. It decodes the smallest possible region of 16 Bytes.

Table 546. Offset 0Eh: HEDT—Header Type Register

Device:	29	**Function:**	4
Offset:	0Eh	**Attribute:**	Read-Only
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:0	Header Type		

16.4.12 Offset 10h: BAR—Base Address Register

Table 547. Offset 10h: BAR—Base Address Register

Device:	29	**Function:**	4
Offset:	10h	**Attribute:**	Read-Write
Default Value:	00000000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:4	Base Address	These bits are used to determine the size of the memory-mapped region being requested.	R/W
3	Prefetchable	Hard-wired to ‘0’, indicating that this range is not pre-fetchable.	RO
2:1	Type	Hard-wired to ‘00’, indicating that this range can be mapped anywhere in 32-bit address space.	RO
0	RTE - Resource Type Indicator	Hard-wired to ‘0’, indicating a request for memory space.	RO

16.4.13 Offset 2Dh - 2Ch: SVID—Subsystem Vendor ID

Note: Software (BIOS) will write the value to this register. After that, the value may be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SID to create one 32-bit write.

Table 548. Offset 2Dh - 2Ch: SVID—Subsystem Vendor ID

Device:	29	**Function:**	4
Offset:	2Dh - 2Ch	**Attribute:**	*Read, Write Once*
Default Value:	00h	**Size:**	*16-bit*
Lockable:	No	**Power Well:**	*Core*

Bits	Name	Description	Access
31:4	Base Address	These bits are used to determine the size of the memory-mapped region being requested.	R/W
3	Prefetchable	Hard-wired to '0', indicating that this range is not pre-fetchable.	RO
2:1	Type	Hard-wired to '00', indicating that this range can be mapped anywhere in 32-bit address space.	RO
0	RTE - Resource Type Indicator	Hard-wired to '0', indicating a request for memory space.	RO

16.4.14 Offset 2Eh - 2Fh: SID—Subsystem ID

Note: Software (BIOS) will write the value to this register. After that, the value may be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SVID to create one 32-bit write.

Table 549. Offset 2Eh - 2Fh: SID—Subsystem ID

Device:	29	**Function:**	4
Offset:	2Eh-2Fh	**Attribute:**	Read, Write Once
Default Value:	00h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Subsystem ID		

16.4.15 Offset 60 - 61h: WDT Configuration Register

Table 550. Offset 60 - 61h: WDT Configuration Register

Device:	29	**Function:**	4
Offset:	60 - 61h	**Attribute:**	Read, Write
Default Value:	00h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:6	Reserved	Reserved.	RO
5	WDT_OUTPUT: Output Enable	This bit indicates whether or not the WDT will toggle the external WDT_TOUT# pin when the WDT times out. 0 = Enabled (Default) 1 = Disabled This signal is muxed with GPIO32.	R/W
4:3	Reserved	**Reserved.**	RO
2	WDT_PRE_SEL: Prescaler Select	The WDT provides two options for prescaling the main down-counter. The preload values are loaded into the main down-counter right justified. The prescaler adjusts the starting point of the 35-bit down counter. 0 = The 20-bit Preload Value is loaded into bits 34:15 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^{15}. The approximate clock generated is 1 KHz, (Default) 1 = The 20-bit Preload Value is loaded into bits 24:5 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^{5}. The approximate clock generated is 1 MHz. **NOTE:** Timeout value is determined by the preload value multiplied by the clock period.	R/W
1:0	WDT_INT_TYPE	The WDT timer supports programmable routing of interrupts. The set of bits allows the user to choose the type of interrupt desired when the WDT reached the end of the first stage without being reset. 00 = IRQ (APIC 1, INT 10) (Default) 01 = Reserved 10 = SMI 11 = Disabled IRQ is Active low, level triggered	R/W

16.4.16 Offset 68h: WDT Lock Register

Table 551. Offset 68h: WDT Lock Register

Device:	29	**Function:**	4
Offset:	68h	**Attribute:**	Read-Write/Write Once
Default Value:	00h	**Size:**	8-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
7:3	Reserved	Reserved.	RO
2	WDT_TOUT_CNF	Timeout Configuration: This register is used to choose the functionality of the timer. 0 = Watchdog Timer Mode: When enabled (i.e., WDT_ENABLE goes from '0' to '1') the timer will reload Preload Value 1 and start decrementing. (Default) Upon reaching the second stage timeout, the WDT_TOUT# is driven low once and will not change again until Power is cycled or a hard reset occurs. 1 = Free Running Mode: WDT_TOUT# will change from previous state when the next timeout occurs. The timer ignores the first stage. The timer only uses Preload Value 2. In this mode the timer is restarted whenever WDT_ENABLE goes from a '0' to a 1. This means that the timer will reload Preload Value 2 and start decrementing every time it is enabled. In free running mode it is not necessary to reload the timer as it is done automatically every time the decrementer reaches zero.	R/W
1	WDT_ENABLE	The following bit enables or disables the WDT. 0 = Disabled (Default) 1 = Enabled **NOTE:** This bit cannot be modified if WDT_LOCK has been set. **NOTE:** In free-running mode, Preload Value 2 is reloaded into the down-counter every time WDT_ENABLE goes from '0' to '1'. In WDT mode, Preload Value 1 is reloaded every time WDT_ENABLE goes from '0' to '1' or the WDT_RELOAD bit is written using the proper sequence of writes (see Register Unlocking Sequence). **WARNING:** Software should ensure that a timeout is not about to occur before disabling the timer. A reload sequence is suggested.	R/W
0	WDT_LOCK	Setting this bit will lock the values of this register until a hard reset occurs or power is cycled. 0 = Unlocked (Default) 1 = Locked This is a Write-Once bit. It cannot be changed until either power is cycled or a hard reset occurs.	R/WO

16.4.17 Offset F8 - FBh: Manufacturer's ID

Table 552. Offset F8 - FBh: Manufacturer's ID

Device:	29	**Function:**	4
Offset:	F8 - FBh	**Attribute:**	Read-Only
Default Value:	00000F66h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:16	Reserved	Reserved.	RO
15:8	Manufacturer	0Fh = Intel	RO
7:0	Process/Dot	66h	RO

16.4.18 Offset Base + 00h: Preload Value 1 Register

Table 553. Offset Base + 00h: Preload Value 1 Register

Device:	29	**Function:**	4
Offset:	Base + 00h	**Attribute:**	Read-Write
Default Value:	FFFFFh	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:20	Reserved	Reserved.	RO
19:0	Preload_Value_1 [19:0]	Use this register to hold the preload value for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down-counter every time the WDT enters the first stage. **NOTE:** The value loaded into the preload register needs to be one less than the intended period, as the timer makes use of zero-based counting (i.e., zero is counted as part of the decrement). Please refer to Section 16.5.2, "Register Unlocking Sequence" for details on how to change the value of this register.	R/W

16.4.19 Offset Base + 04h: Preload Value 2 Register

Preload Value 2 Register

Table 554. Offset Base + 04h: Preload Value 2 Register

Device:	29	***Function:***	4
Offset:	Base + 04h	***Attribute:***	Read-Write
Default Value:	FFFFFh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:20	Reserved	Reserved.	RO
19:0	Preload_Value_2 [19:0]	Use this register to hold the preload value for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down-counter every time the WDT enters the second stage. **NOTE:** The value loaded into the preload register needs to be one less than the intended period, as the timer makes use of zero-based counting (i.e., zero is counted as part of the decrement). Please refer to Section 16.5.2, “Register Unlocking Sequence” for details on how to change the value of this register.	R/W

16.4.20 Offset Base + 08h: General Interrupt Status Register

Preload Value 2 Register

Table 555. Offset Base + 08h: General Interrupt Status Register

Device:	29	***Function:***	4
Offset:	Base + 08h	***Attribute:***	Read-Write Clear
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:1	Reserved	Reserved.	RO
0	Watchdog Timer Interrupt Active:	(1st Stage) This bit is set when the first stage of the 35-bit down-counter reaches zero. An interrupt will be generated if WDT_INT_TYPE is configured to do so (See WDT Configuration Register). This is a sticky bit and is only cleared by writing a 1. 0 = No Interrupt 1 = Interrupt Active **NOTE:** This bit is not set in free-running mode.	RWC

16.4.21 Offset Base + 0Ch: Reload Register

Table 556. Offset Base + 0Ch: Reload Register

Device:	29	***Function:***	4
Offset:	Base + 0Ch	***Attribute:***	Read-Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:10	Reserved	Reserved.	RO
9	WDT_TIMEOUT	This bit resides in the RTC Well and its value is not lost if the host resets the system. It is set to '1' if the host fails to reset the WDT before the 35-bit Down-Counter reaches zero for the second time in a row. This bit is cleared by performing the Register Unlocking Sequence followed by a '1' to this bit. 0 = Normal (Default). 1 = System has become unstable. **NOTE:** In free running mode this bit is set every time the down counter reaches zero.	R/W
8	WDT_RELOAD	To prevent a timeout, the host must perform the Register Unlocking Sequence followed by a '1' to this bit (See Register Unlocking Sequence). **NOTE:** Refer to Register Unlocking Sequence for details on how to write to this bit.	R/W
7:0	Reserved	Reserved.	W

NOTE: The reload sequence is only necessary for the Reload register and Preload_Value registers and is not used in Free Running mode.

16.5 Theory Of Operation

16.5.1 RTC Well and WDT_TOUT# Functionality

The WDT_TIMEOUT bit is set to a ‘1’ when the WDT 35-bit down counter reaches zero for the second time in a row. The WDT_TOUT# pin is then toggled LOW by the WDT from the Intel® 6300ESB ICH. The board designer should attach the WDT_TOUT# to the appropriate external signal. When WDT_TOUT_CNF is a ’1’ the WDT toggles WDT_TOUT# again when the next timeout occurs. Otherwise, WDT_TOUT# is driven low until the system is reset or power is cycled.

16.5.2 Register Unlocking Sequence

The register unlocking sequence is necessary whenever writing to the RELOAD register or either PRELOAD_VALUE registers. The host must write a sequence of two writes to offset BAR + 0Ch before attempting to write to either the WDT_RELOAD and WDT_TIMEOUT bits of the RELOAD register or the PRELOAD_VALUE registers. The first writes are 80 and 86, in that order, to offset BAR + 0Ch. The next write will be to the proper register (e.g., RELOAD, PRELOAD_VALUE_1, PRELOAD_VALUE_2)

The following is an example of how to prevent a timeout:

1. Write 80 to offset BAR + 0Ch.
2. Write 86 to offset BAR + 0Ch.
3. Write a ‘1’ to RELOAD [8] (WDT_RELOAD) of the Reload Register.

Note: Any subsequent writes require that this sequence be performed again.

16.5.3 Reload Sequence

To keep the timer from causing an interrupt or driving WDT TOUT#, the timer must be updated periodically. Other timers refer to updating the timer as “kicking” the timer. The frequency of updates required is dependent on the value of the Preload values. To update the timer, the Register Unlocking Sequence must be performed followed by writing a ‘1’ to bit 8 at offset BAR+ 0Ch within the watchdog timer memory mapped space. This sequence of events is referred to as the “Reload Sequence”.

16.5.4 Low Power State

The Watchdog Timer does not operate when PCICLK is stopped.

APIC1 Configuration Registers (D29:F5) 17

APIC1's direct registers are assigned with base address FEC1xxxxH. To support legacy device/driver on external PCI bus used with the Intel ICHx, APIC1 has an alternate base address FEC0xxxxH. This means external PCI devices may write to IRQ pin assertion register (either FEC0_0020H or FEC1_0020H) to generate interrupt from APIC1. Devices on the primary PCI bus can write to IRQ Pin Assertion Register FEC0_0020H to generate an APIC0 interrupt. Devices/drivers on the PCI-X segment have write access only on the APIC1 IRQ Pin Assertion Register. Devices/drivers on the PCI segment can access only APIC0 registers.

Since the Intel® 6300ESB ICH does not implement Hub Interface EOI special cycle, MCH will translate EOI special cycle to a memory write cycle to EOI register at address FEC0_0040H and pass it to the Intel® 6300ESB ICH. This memory write cycle will be passed to both APIC0 and APIC1 internally.

From CPU/MCH point of view, it should always use address FEC0xxxxH to access APIC0 registers and address FEC1xxxxH to access APIC1 registers. APIC1 will not respond to CPU/MCU's access to address FEC0xxxxH other than the EOI cycle stated above.

APIC1 also includes an XAPIC_EN config bit. This bit must be set to enable the I/O (x) APIC extension to the I/O APIC. For APIC1, this extension is always enabled.

17.1 APIC1 Configuration Registers (D29:F5)

Note: Registers that are not shown should be treated as Reserved. See "PCI Configuration Map" on page 277 for details.

Table 557. APIC1 Configuration Map (D29:F5) (Sheet 1 of 2)

Offset	Mnemonic	Register Name	Default	Type
00-03h	VID_DID	Vendor ID/Device ID	25ACH/8086h	RO
04-05h	APIC1CMD	APIC1 Command Register	0000h	R/W
06-07h	APIC1STA	APIC1 Device Status Register	0010h	RO
08h	RID	Revision ID	See Note 1	RO
09-0Bh	CC	Class Code	080020h	RO
0C-0Fh	HEADTYP	Header Type	00000000h	RO
2C-2F	SS	Subsystem Identifiers	00000000h	RWO
34h	CAP	Capabilities Pointer	50h	RO
3Ch	ILINE	Interrupt Line	00h	R/W
3Dh	IPIN	Interrupt Pin	00h	RO
40-41h	ABAR	Alternate Base Address Register	8000h	R/W

NOTES:
1. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

Table 557. APIC1 Configuration Map (D29:F5) (Sheet 2 of 2)

Offset	Mnemonic	Register Name	Default	Type
44-47h	MBAR	Memory Base Register	FEC10000h	RO
50-51h	XID	PCI-X Identifiers	0007h	RO
54-57h	XSR	PCI-X Status	000100EDh	RO

NOTES:
1. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.

17.1.1 Offset 00 - 03h: VID_DID—Vendor/ID Register (APIC1—D29:F5)

Table 558. Offset 00 - 03h: VID_DID—Vendor/ID Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	00 - 03h	***Attribute:***	Read-Only
Default Value:	25ACh-8086h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Device ID Value	This is a 16-bit value assigned to the APIC1. DID = 25ACh	
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel. Intel VID = 8086h	

17.1.2 Offset 04 - 05h: APIC1CMD—APIC1 COMMAND Register (APIC1—D29:F5)

Table 559. Offset 04 - 05h: APIC1CMD—APIC1 COMMAND Register (APIC1—D29:F5) (Sheet 1 of 2)

Device:	29	***Function:***	5
Offset:	04-05h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:9	Reserved	Reserved.	
8	SERR_EN: SERR# Enable	SERR# Enable controls the enable for the DO_SERR special cycle on the hub interface. 0 = Disable special cycle. 1 = Enable special cycle.	R/W
7	Reserved	Reserved.	

Table 559. Offset 04 - 05h: APIC1CMD—APIC1 COMMAND Register (APIC1—D29:F5) (Sheet 2 of 2)

Device:	29	**Function:**	5
Offset:	04-05h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
6	PERE: Parity Error Response Enable	0 = No action is taken when detecting a parity error. 1 = The Intel® 6300ESB ICH will take normal action when a parity error is detected. **NOTE:** D30FO offset F4h, bit #2 must be set to a '1' AND D28FO offset F4h, bit #2 must be set to a '1' in order for PERE to have any effect.	R/W
5:3	Reserved	Reserved.	
2	BME: Bus Master Enable	Controls the I/O APIC1's ability to act as a master on Hub Interface when forwarding processor side bus interrupt messages.	R/W
1	MSE: Memory Space Enable	Controls the I/O APIC1's response as a target to memory accesses that address the I/O APIC1.	R/W
0	Reserved	Reserved.	

17.1.3 Offset 06 - 07h: APIC1STA—APIC1 Device Status (APIC1—D29:F5)

Table 560. Offset 06 - 07h: APIC1STA—APIC1 Device Status (APIC1—D29:F5)

Device:	29	**Function:**	5
Offset:	06 - 07h	**Attribute:**	Read-Only
Default Value:	0010h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10:9	DEV_STS: DEVSEL# Timing Status	00 = Fast Decode. **NOTE:** These bits are set for fast decode '00', but a true device select does not exist, so they have no effect.	RO
8:6	Reserved	Reserved.	
5	66MHZ_CAP: 66 MHz capable	Hardwired to 1. Not 66 MHz capable.	RO
4	Capabilities List	This bit is hardwired to '1', indicating the presence of a valid capabilities pointer at offset 34h.	
3:0	Reserved	Reserved.	

17.1.4 Offset 08h: RID—Revision ID Register (APIC1—D29:F5)

Table 561. Offset 08h: RID—Revision ID Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	08h	***Attribute:***	Read-Only
Default Value:	See bit description	***Size:***	8-bit

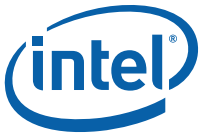
Bits	Name	Description	Access
7:0	Revision ID Value	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID Register.	RO

17.1.5 Offset 09 - 0Bh: CC—Class Code Register (APIC1—D29:F5)

Table 562. Offset 09 - 0Bh: CC—Class Code Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	09-0Bh	***Attribute:***	Read-Only
Default Value:	080020h	***Size:***	24-bit

Bits	Name	Description	Access
23:16	BCC: Base Class Code	The value of 08h indicates that this is a generic system peripheral.	RO
15:8	SCC: Sub Class Code	The value of 00h indicates that this generic peripheral is an interrupt controller.	RO
7:0	PIF: Programming Interface	The value of 20h indicates that this interrupt peripheral is an I/OxAPIC.	RO



17.1.6 Offset 0C - 0Fh: HEADTYP—Header Type Register (APIC1—D29:F5)

Table 563. Offset 0C - 0Fh: HEADTYP—Header Type Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	0C - 0Fh	***Attribute:***	Read-Only
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23:16	Header Type	This indicates that it is a type ‘00’ header (normal PCI device) and that it is a single function device.	RO
15:0	Reserved	Reserved.	

17.1.7 Offset 2C - 2Fh: SS—APIC1 Subsystem Identifiers (APIC1—D29:F5)

Note: This register is initialized to logic ‘0’ by the assertion of PXPCIRST#. This register may be written only once after PXPCIRST# deassertion.

Table 564. Offset 2C - 2Fh: SS—APIC1 Subsystem Identifiers (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	2C - 2Fh	***Attribute:***	Read/Write Once
Default Value:	0000000h	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:16	SSID: Subsystem ID	Write once register for subsystem ID.	RWO
15:0	SSVID: Subsystem Vendor ID	Write once register for holding the subsystem vendor ID.	RWO

17.1.8 Offset 34h: CAP_PTR—APIC1 Capabilities Pointer (APIC1—D29:F5)

Table 565. Offset 34h: CAP_PTR—APIC1 Capabilities Pointer (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	34h	***Attribute:***	Read-Only
Default Value:	50h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	CAP: Capabilities Pointer	This register points to the starting offset (50h) of the I/O APIC1 capabilities range.	RO

17.1.9 Offset 3Ch: ILINE—Interrupt Line (APIC1—D29:F5)

Table 566. Offset 3Ch: ILINE—Interrupt Line (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	3Ch	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
17:0	ILINE: Interrupt Line	This data is not used by the Intel® 6300ESB ICH. It is used as a scratchpad register to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

17.1.10 Offset 3Dh: IPIN—Interrupt Pin (APIC1—D29:F5)

Table 567. Offset 3Dh: IPIN—Interrupt Pin (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
7:0	IPIN: Interrupt pin	The value of 00h indicates that I/O APIC1 does not connect to PIRQ#.	RO

17.1.11 Offset 40 - 41h: ABAR—APIC1 Alternate Base Address Register (APIC1—D29:F5)

Note: This register contains an alternate base address in the legacy APIC range. This range may coexist with the BAR register range. This range is needed for OSs that support the APIC but do not yet support remapping the APIC anywhere in the 4 Gbyte address space.

Note: On downstream writes, only ABAR + Offset 40h (EOI) is claimed. On upstream cycles, only ABAR + offset 20h are claimed.

Table 568. Offset 40 - 41h: ABAR—APIC1 Alternate Base Address Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	40-41h	***Attribute:***	Read/Write
Default Value:	8000h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15	EN: Enable	When set, the range FECX_YZ00 to FECX_YZFF is enabled as an alternate access method to the IOxAPIC registers. Bits 'XYZ' are defined below.	RO
14	BIE: Boot Interrupt Enable	0 = Boot interrupt is enabled. 1 = Boot interrupt is disabled. **NOTE:** For details on the Boot interrupt, see Section 5.7.3, "Boot Interrupt".	RW
13:12	Reserved	Reserved.	
11:8	Base Address [19:16] (XBAD)	These bits determine the high order bits of the I/O APIC address map. When a memory address is recognized by the Intel® 6300ESB ICH that matches FECX_YZ00 or FECX_YZ10, the Intel® 6300ESB ICH will respond to the cycle and access the internal I/O APIC1.	
7:4	Base Address [15:12] (YBAD)	These bits determine the low order bits of the I/O APIC address map. When a memory address is recognized by the Intel® 6300ESB ICH that matches FECX_YZ00 or FECX_YZ10, the Intel® 6300ESB ICH will respond to the cycle and access the internal I/O APIC1.	
3:0	Base Address [11:8] (ZBAD)	These bits determine the low order bits of the I/O APIC address map. When a memory address is recognized by the Intel® 6300ESB ICH that matches FECX_YZ00 or FECX_YZ10, the Intel® 6300ESB ICH will respond to the cycle and access the internal I/O APIC1	

17.1.12 Offset 44 - 47h: MBAR—APIC1 Memory Base Register (APIC1—D29:F5)

Note: This register contains the APIC1 Base Address for the memory space.

Table 569. Offset 44 - 47h: MBAR—APIC1 Memory Base Register (APIC1—D29:F5)

Device:	29	**Function:**	5
Offset:	44-47h	**Attribute:**	Read-Only
Default Value:	FEC10000h	**Size:**	32-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
31:12	ADDR: Address	These bits determine the base address of the I/O APIC1.	RO
11:4	Reserved	Reserved.	
3	PF: Prefetchable	Indicates that the BAR is not pre-fetchable.	RO
2:1	LOC: Location	'00' indicates that the address may be located anywhere in the 32-bit address space.	RO
0	SI: Space Indicator	Indicates that the BAR is in memory space.	RO

17.1.13 Offset 50 - 51h: XID—PCI-X Identifiers Register (APIC1—D29:F5)

Table 570. Offset 50 - 51h: XID—PCI-X Identifiers Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	50-51h	***Attribute:***	Read-Only
Default Value:	0007h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:8	XNPTR: Next Pointer	Points to the next capabilities list pointer (empty).	RO
7:0	XCID: Capability ID	Capabilities ID indicates PCI-X (07h).	RO

17.1.14 Offset 52h: XSR—PCI-X Status Register (APIC1—D29:F5)

Table 571. Offset 52h: XSR—PCI-X Status Register (APIC1—D29:F5)

Device:	29	***Function:***	5
Offset:	52h	***Attribute:***	Read-Only
Default Value:	000300EDh	***Size:***	32-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
31:21	Reserved	Reserved.	
20	Device Complexity	Hardwired to logic '0' to indicate that this is a simple device.	RO
19	Unexpected Split Completion	This device will never see an unexpected split completion, as it never generates any master cycles besides posted writes for MSI.	RO
18	Split Completion Discarded	This device does not support Split Completion.	RO
17	133 MHz Capable	Hardwired to logic '0' to indicate this device is not 133 MHz capable.	RO
16	64-bit Device	Hardwired to logic '1' to indicate that this is a 64-bit device.	RO
15:8	Bus Number	Indicates the bus number of the bus segment for this device. This value will match the primary bus number field from the attached bridge.	RO
7:3	Device Number	Reflects the device number that has been hard-coded for the device. This number will be 1Dh (29) for APIC1.	RO
2:0	Function Number	Reflects the function number for the device.	RO

17.2 Advanced Interrupt Controller (APIC)

There are two APICs in the Intel® 6300ESB ICH: APIC0 and APIC1 (in device 29, function 5). APIC0's direct registers are assigned with base address FEC0xxxxH; however, no external PCI device may write to these registers. APIC1's direct registers are assigned with base address FEC1xxxxH. To support legacy device/ drivers on external PCI bus used with the Intel ICHx, APIC1 has an alternate base address, FEC0xxxxH. This means external PCI devices may write to IRQ pin assertion register (either FEC0_0020H or FEC1_0020H) to generate interrupt from APIC1.

APIC0 also includes an XAPIC_EN config bit. This bit must be set to enable the I/O (x) APIC extension to the I/O APIC. For APIC1, this extension is always enabled.

17.2.1 APIC1 Direct Register Map

The APIC is accessed through an indirect addressing scheme. Two registers are visible by software for manipulation of most of the APIC registers. These registers are mapped into memory space. The registers are shown in Table 572.

Table 572. APIC1 Direct Registers

Address	Register	Size	Type
FEC1_0000h	Index Register	8 bits	R/W
FEC1_0010h	Data Register	32 bits	R/W
FEC1_0020h	IRQ Pin Assertion Register	32 bits	R/W
FEC1_0040h	EOI Register	32 bits	R/W

Table 573 lists the registers which may be accessed within the APIC through the Index Register. When accessing these registers, accesses must be done a DWORD at a time. For example, software should never access byte 2 from the Data register before accessing bytes 0 and 1. The hardware will not attempt to recover from a bad programming model in this case.

Table 573. APIC Indirect Registers

Index	Register	Size	Type
00	ID	32 bits	R/W
01	Version	32 bits	RO
02	Arbitration ID	32 bits	RO
03	Boot Configuration	32 bits	R/W
04-0F	Reserved		RO
10 -11	Redirection Table 0	64 bits	R/W
12 - 13	Redirection Table 1	64 bits	R/W
3E-3F	Redirection Table 23	64 bits	R/W
40-FF	Reserved		RO

17.2.2 IND—Index Register

Note: The Index Register will select which APIC indirect register to be manipulated by software. The selector values for the indirect registers are listed in Table 573. Software will program this register to select the desired APIC internal register.

Table 574. IND—Index Register

Device:	29	**Function:**	5
Memory Address:	FEC0_0000h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:0	APIC Index	This is an 8-bit pointer into the I/O APIC register table.	R/W

17.2.3 DAT—Data Register

Note: This is a 32-bit register specifying the data to be read or written to the register pointed to by the Index register. This register may only be accessed in DWORD quantities.

Table 575. DAT—Data Register

Device:	29	**Function:**	5
Memory Address:	FEC0_0010h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
7:0	APIC Data	This is a 32-bit register for the data to be read or written to the APIC indirect register pointed to by the Index register.	R/W

17.2.4 IRQPA—IRQ Pin Assertion Register

The IRQ Pin Assertion Register is present to provide a mechanism to scale the number of interrupt inputs into the I/O APIC without increasing the number of dedicated input pins. When a device that supports this interrupt assertion protocol requires interrupt service, that device will issue a write to this register. Bits 4:0 written to this register contain the IRQ number for this interrupt. The only valid values are 0-23. Bits 31:5 are ignored. To provide for future expansion, peripherals should always write a value of 0 for Bits 31:5.

Note: Writes to this register are only allowed by the processor and by masters on the Intel® 6300ESB ICH's PCI bus. Writes by devices on PCI buses above the Intel® 6300ESB ICH (e.g., a PCI segment on a P64H) are not supported.

Table 576. IRQPA—IRQ Pin Assertion Register

Device:	29	**Function:**	5
Memory Address:	FEC0_0020h	**Attribute:**	Write-Only
Default Value:	N/A	**Size:**	32-bit

Bits	Name	Description	Access
31:5	Reserved	Reserved. To provide for future expansion, the processor should always write a value of '0' to Bits 31:5.	
4:0	IRQ Number	Bits 4:0 written to this register contain the IRQ number for this interrupt. The only valid values are 0-23.	WO

17.2.5 EOIR—EOI Register

The EOI register is present to provide a mechanism to maintain the level-triggered semantics for level-triggered interrupts issued on the parallel bus.

When a write is issued to this register, the I/O APIC will check the lower 8 bits written to this register and compare them with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared.

Note: When there are multiple I/O Redirection entries, assign the same vector for more than one interrupt input. Each of those entries will have the Remote_IRR bit reset to '0'. The interrupt that was prematurely reset will not be lost because if its input remained active when the Remote_IRR bit is cleared, the interrupt will be reissued and serviced at a later time. Only bits 7:0 are actually used. Bits 31:8 are ignored.

Table 577. EOIR—EOI Register

Device:	29	**Function:**	5
Memory Address:	FEC0_0040h	**Attribute:**	Write-Only
Default Value:	N/A	**Size:**	32-bit

Bits	Name	Description	Access
31:8	Reserved	Reserved. To provide for future expansion, the processor should always write a value of '0' to Bits 31:8.	
7:0	End of Interrupt (EOI)	Vector to be compared with vector field in the I/O redirection table when an EOI is issued.	WO

17.2.6 Offset 00h: ID—Identification Register

Note: The APIC ID serves as a physical name of the APIC1. This register is reset to '0' on power-up reset.

Table 578. Offset 00h: ID—Identification Register

Device:	29	**Function:**	5
Offset:	00h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:28	Reserved	Reserved.	
27:24	APIC ID	Software must program this value before using the APIC.	R/W
23:0	Reserved	Reserved.	

17.2.7 Offset 01h: VER—Version Register

Note: Each I/O APIC contains a hardwired Version Register that identifies different implementations of APIC and their versions. The maximum redirection entry information also is in this register to let software know how many interrupts are supported by this APIC.

Table 579. Offset 01h: VER—Version Register

Device:	29	**Function:**	5
Offset:	01h	**Attribute:**	Read-Only
Default Value:	00178020h	**Size:**	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23:16	Maximum Redirection Entries	This is the entry number (0 being the lowest entry) of the highest entry in the redirection table. It is equal to the number of interrupt input pins minus one and is in the range 0 through 239. In the Intel® 6300ESB ICH, this field is hardwired to 17h to indicate 24 interrupts.	RO
15	PRQ	This bit is set to '1' to indicate that this version of the I/O APIC implements the IRQ Assertion register and allows PCI devices to write to it to cause interrupts.	RO
14:8	Reserved	Reserved.	
7:0	Version	This is a version number that identifies the implementation version. The version number assigned to the Intel® 6300ESB ICH for the I/O (x) APIC is 20h.	RO

17.2.8 Offset 03h: BOOT_CONFIG—Boot Configuration Register

Note: This register is used to control the interrupt delivery mechanism for the APIC.

Table 580. Offset 03h: BOOT_CONFIG—Boot Configuration Register

Device:	29	***Function:***	5
Offset:	03h	***Attribute:***	Read-Only
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:1	Reserved	Reserved.	
0	DT: Delivery Type	Hardwire to 1. Interrupt delivery mechanism is always a Processor System Bus message.	RO

17.2.9 Redirection Table

The Redirection Table has a dedicated entry for each interrupt input pin. The information in the Redirection Table is used to translate the interrupt manifestation on the corresponding interrupt pin into an APIC message.

Note: The APIC will respond to an edge-triggered interrupt as long as the interrupt is held until after the acknowledge cycle has begun. Once the interrupt is detected, a delivery status bit internally to the I/O APIC is set. The state machine will step ahead and wait for an acknowledgment from the APIC bus unit that the interrupt message was sent over the APIC bus. Only then will the I/O APIC be able to recognize a new edge on that interrupt pin. That new edge will only result in a new invocation of the handler if its acceptance by the destination APIC causes the Interrupt Request Register bit to go from '0' to '1'. (i.e., if the interrupt was not already pending at the destination.)

Table 581. Redirection Table (Sheet 1 of 3)

Device:	29	***Function:***	5
Offset:	10h-11h (vector 0) through 3E-3Fh (vector 23)	***Attribute:***	Read/Write
Default Value:	Bit 16-1, Bits[15:12]=0. All other bits undefined	***Size:***	64 bits each, accessed as two 32 bit quantities

Bits	Name	Description	Access
63:56	Destination	When bit 11 of this entry is 0 [Physical], bits [59:56] specify an APIC ID. In this case, bits 63:59 should be programmed by software to 0. When bit 11 of this entry is '1' [Logical], bits [63:56] specify the logical destination address of a set of processors.	R/W
55:48	Extended Destination ID (EDID)	These bits are only sent to a local APIC when in Processor System Bus mode. They become bits [11:4] of the address.	
47:18	Reserved	Reserved. Software should program these bits to 0.	
17	Disable Flushing (DFLSH)	This bit is maintained for any potential software compatibility, but the Intel® 6300ESB ICH performs no flushing action, regardless of the setting of this bit.	

Table 581. Redirection Table (Sheet 2 of 3)

Device:	29	**Function:**	5
Offset:	10h-11h (vector 0) through 3E-3Fh (vector 23)	**Attribute:**	Read/Write
Default Value:	Bit 16-1, Bits[15:12]=0. All other bits undefined	**Size:**	64 bits each, accessed as two 32 bit quantities

Bits	Name	Description	Access
16	Mask	0 = Not masked: An edge or level on this interrupt pin results in the delivery of the interrupt to the destination. 1 = Masked: Interrupts are not delivered nor held pending. Setting this bit after the interrupt is accepted by a local APIC has no effect on that interrupt. This behavior is identical to the device withdrawing the interrupt before it is posted to the processor. It is software's responsibility to deal with the case where the mask bit is set after the interrupt message has been accepted by a local APIC unit but before the interrupt is dispensed to the processor.	R/W
15	Trigger Mode	This field indicates the type of signal on the interrupt pin that triggers an interrupt. 0 = Edge triggered. 1 = Level triggered.	R/W
14	Remote IRR	This bit is used for level-triggered interrupts in Fixed or Lowest Priority Delivery Modes only; its meaning is undefined for edge triggered interrupts. For level-triggered interrupts, this bit is set if the I/O APIC successfully sends the level interrupt message. Remote IRR bit is reset when an EOI message is received that matches the interrupt vector in this entry. This bit is never set for SMI, NMI, INIT, or ExtINT delivery modes.	R/W
13	Interrupt Input Pin Polarity	This bit specifies the polarity of each interrupt signal connected to the interrupt pins. 0 = Active high. 1 = Active low.	R/W
12	Delivery Status	This field contains the current status of the delivery of this interrupt. Writes to this bit have no effect. 0 = Idle. No activity for this interrupt. 1 = Pending. Interrupt has been injected, but delivery is held up due to the APIC bus being busy or the inability of the receiving APIC unit to accept the interrupt at this time.	RO
11	Destination Mode	This field determines the interpretation of the Destination field. 0 = Physical. Destination APIC ID is identified by bits [59:56]. 1 = Logical. Destinations are identified by matching bit [63:56] with the Logical Destination in the Destination Format Register and Logical Destination Register in each Local APIC.	R/W

Table 581. Redirection Table (Sheet 3 of 3)

Device:	29	**Function:**	5
Offset:	10h-11h (vector 0) through 3E-3Fh (vector 23)	**Attribute:**	Read/Write
Default Value:	Bit 16-1, Bits[15:12]=0. All other bits undefined	**Size:**	64 bits each, accessed as two 32 bit quantities

Bits	Name	Description	Access
10:8	Delivery Mode	This field specifies how the APICs listed in the destination field should act upon reception of this signal. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. These encodings are listed in the note below:	R/W
7:0	Vector	This field contains the interrupt vector for this interrupt. Values range between 10h and FEh.	R/W

NOTE: Delivery Mode encoding:
000 = Fixed. Deliver the signal on the INTR signal of all processor cores listed in the destination. Trigger Mode can be edge or level.
001 = Lowest Priority. Deliver the signal on the INTR signal of the processor core that is executing at the lowest priority among all the processors listed in the specified destination. Trigger Mode can be edge or level.
010 = SMI. This delivery mode is not supported.
011 = Reserved.
100 = NMI. This delivery mode is not supported.
101 = INIT. This delivery mode is not supported.
110 = Reserved
111 = ExtINT. Deliver the signal to the INTR signal of all processor cores listed in the destination as an interrupt that originated in an externally connected 8259A compatible interrupt controller. The INTA cycle that corresponds to this ExtINT delivery will be routed to the external controller that is expected to supply the vector. Requires the interrupt to be programmed as edge triggered. The Remote IRR bit is never set if programmed for ExtINT level-triggered operation; as a result a continuous stream of interrupts will be generated as long as the INTR input is asserted.

PCI-X Overview (D28:F0) 18

Note: Since the Intel® 6300ESB ICH supports a PCI interface and a PCI-X interface, the notation PCI-X will be used to refer to the PCI-X interface. Since the PCI-X interface can support the PCI-X protocol as well as the PCI protocol, the PCI-X terminology is intended to refer to the interface being described and not to the protocol.

18.1 I/O Window Addressing

This section describes the I/O window that may be set up in the bridge. Refer to Section 18.3, "VGA Addressing" to see how I/O cycles in the VGA range are handled.

The register bits listed below also modify the response by the Intel® 6300ESB ICH to I/O transactions:

- I/O Base and Limit Registers
- I/O Enable bit in the Command Register
- Master enable bit in the Command Register
- Enable 1K granularity in the Intel® 6300ESB ICH Configuration Register

To enable outbound I/O transactions, the I/O enable bit must be set in the command register in the Intel® 6300ESB ICH configuration space (bit '0' at offset 04-05h). When the I/O enable bit is not set, all I/O transactions initiated on the Hub Interface receive a master abort completion. No inbound I/O transactions may cross the bridge and are therefore master aborted.

The Intel® 6300ESB ICH implements one set of I/O base and limit address registers in configuration space that define an I/O address range for the bridge. Hub interface I/O transactions with addresses that fall inside the range defined by the I/O base and limit registers are forwarded to PCI-X, and PCI-X I/O transactions with addresses that fall outside this range are master aborted.

Setting the base address to a value greater than that of the limit address turns off the I/O range. When the I/O range is turned off, no I/O transactions are forwarded to PCI even when the I/O enable bit is set. The I/O range has a minimum granularity of 4 Kbytes and is aligned on a 4 Kbyte boundary. The maximum I/O range is 64 Kbytes. This range may be lowered to 1K granularity by setting the EN1K bit in the Intel® 6300ESB ICH Configuration register at offset 40h.

The base register consists of an 8-bit field at configuration address 1Ch, and a 16-bit field at address 30h. The top four bits of the 8-bit field define bits [15:12] of the I/O base address. The bottom four bits read only as 0h to indicate that the Intel® 6300ESB ICH supports 16-bit I/O addressing. Bits [11:0] of the base address are assumed to be '0', which naturally aligns the base address to a 4 Kbyte boundary. The I/O base upper 16 bits register at offset 30h is reserved. After chip reset, the value of the I/O base address is initialized to 0000h.

The I/O limit register consists of an 8-bit field at offset 1Dh and a 16-bit field at offset 32h. The top four bits of the 8-bit field define bits [15:12] of the I/O limit address. The bottom four bits read only as 0h to indicate that 16-bit I/O addressing is supported. Bits [11:0] of the limit address are assumed to be FFFh, which naturally aligns the limit address to the top of a 4 Kbyte I/O address block. The 16 bits contained in the I/O limit upper 16 bits register at offset 32h are reserved. After chip reset, the value of the I/O limit address is reset to 0FFFh.

Note: When the EN1K bit is set in the Intel® 6300ESB ICH Configuration register, the base and limit registers are changed so that the top six bits of the 8-bit field define bits [15:10] of the I/O base/limit address, and the bottom two bits read only as 0h to indicate support for 16-bit I/O addressing. Bits [9:0] are assumed to be '0' for the base register and '1' for the limit register, which naturally aligns the address to a 1 Kbyte boundary.

18.2 Memory Window Addressing

This section describes the memory windows that may be set up in the bridge. Refer to Section 18.2.2, "Prefetchable Memory Base and Limit Address Registers, Upper 32-Bit Registers" to see how memory cycles in the VGA range are handled.

- Memory-mapped I/O Base and Limit registers
- Prefetchable Memory Base and Limit registers
- Prefetchable Memory Base and Limit upper 32 bits register
- Memory Enable bit in the Command register
- Master Enable bit in the Command register

To enable outbound memory transactions, the memory space enable bit in the command register must be set (bit 1 of offset 04-05h). To enable inbound memory transactions, the master enable bit in the command register must be set (bit 2 of offset 04-05h). The Intel® 6300ESB ICH does not prefetch data from PCI devices. The Intel® 6300ESB ICH supports 64 bits of addressing (DAC cycles) on both interfaces.

18.2.1 Memory Base and Limit Address Registers

The memory base address and memory limit address registers define an address range that the Intel® 6300ESB ICH uses to determine when to forward memory commands. The Intel® 6300ESB ICH forwards a memory transaction from the Hub Interface to PCI-X when the address falls within the range, and forwards it from PCI-X to the Hub Interface when the address is outside the range, provided that they do not fall into the prefetchable memory range (see Section 18.2.2, "Prefetchable Memory Base and Limit Address Registers, Upper 32-Bit Registers"). This memory range supports 32-bit addressing only (addresses 4 Gbytes). It has a granularity and alignment of 1 Mbyte.

This range is defined by a 16-bit base address register at offset 20h in configuration space and a 16-bit limit address register at offset 22h. The top 12 bits of each of these registers correspond to bits [31:20] of the memory address. The low four bits are hardwired to '0'. The low 20 bits of the base address are assumed to be all '0', which results in a natural alignment to a 1 Mbyte boundary. The low 20 bits of the limit address are assumed to be all '1's, which results in an alignment to the top of a 1 Mbyte block.

Setting the base to a value greater than that of the limit turns off the memory range.

18.2.2 Prefetchable Memory Base and Limit Address Registers, Upper 32-Bit Registers

The prefetchable memory base and address registers, along with their upper 32-bit counterparts, define an additional address range that the Intel® 6300ESB ICH uses to forward accesses. The Intel® 6300ESB ICH forwards a memory transaction from the Hub Interface to PCI-X when the address falls within the range, and forwards

transactions from PCI-X to the Hub Interface when the address is outside the range and do not fall into the regular memory range (see Section 18.2.1, “Memory Base and Limit Address Registers”). This memory range supports 64-bit addressing and has a granularity and alignment of 1 Mbyte.

This lower 32 bits of the range are defined by a 16-bit base register at offset 24h in configuration space and a 16-bit limit register at offset 26h. The top 12 bits of each of these registers correspond to bits [31:20] of the memory address. The low four bits are hardwired to 1h, indicating 64-bit address support. The low 20 bits of the base address are assumed to be all ‘0’s, which results in a natural alignment to a 1 Mbyte boundary. The low 20 bits of the limit address are assumed to be all ‘1’s, which results in an alignment to the top of a 1 Mbyte block.

The upper 32 bits of the range are defined by a 32-bit base register at offset 28h in configuration space and a 32-bit limit register at offset 2Ch.

Setting the entire base (with upper 32 bits) to a value greater than that of the limit turns off the memory range.

18.3 VGA Addressing

When a VGA-compatible device exists behind an Intel® 6300ESB ICH bridge, the VGA enable bit in the bridge control register is set (offset 3 at 3E-3Fh). When set, the Intel® 6300ESB ICH forwards all transactions addressing the VGA frame buffer memory and VGA I/O registers from the Hub Interface to PCI-X, regardless of the values of the Intel® 6300ESB ICH base and limit address registers. When set, the Intel® 6300ESB ICH does not forward VGA frame buffer memory accesses to the Hub Interface regardless of the values of the memory address ranges. However, the I/O enable and memory enable bit in the command register must still be set. When cleared, the Intel® 6300ESB ICH forwards transactions addressing the VGA frame buffer memory and VGA I/O registers from the Hub Interface to PCI-X when the defined memory address ranges enable forwarding. When cleared, accesses to the VGA frame buffer memory are forwarded from PCI-X to the Hub Interface when the defined memory address ranges enable forwarding. However, the master enable bit must still be set. The VGA I/O addresses are never forwarded to the Hub Interface.

The VGA frame buffer consists of the following memory address range: 000A 0000h–00B FFFFh.

The VGA I/O addresses consist of the I/O addresses 3B0h–3BBh and 3C0h–3DFh. These I/O addresses are aliased every 1 Kbyte throughout the first 64 Kbyte of I/O space. This means that address bits [9:0] (3B0h-3BBh and 3C0h-3DFh) are decoded, [15:10] are not decoded and may be any value, and address bits [31:16] must be all ‘0’s.

18.4 Configuration Addressing

Figure 30 shows how the Intel® 6300ESB ICH appears to configuration software.

Figure 30. Intel® 6300ESB I/O Controller Hub Appearance to Software

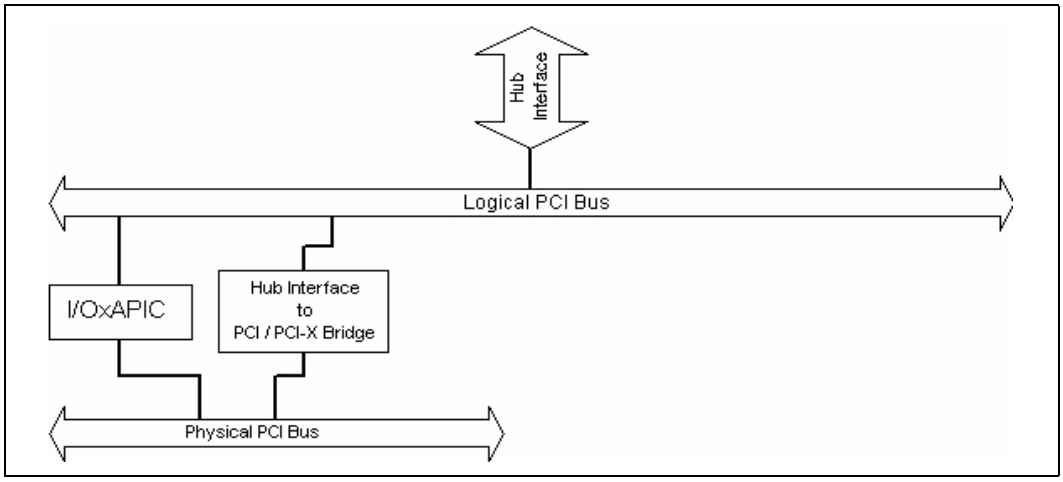


Table 582. Configuration Addressing

Function	Hub Interface ID	PCI-X Bus	PCI-X Dev	PCI-X Func
PCI/PCI-XBus	6	0 (Hub Interface)	28 h	0

As seen in Figure 30, the SM Bus controller does not appear to software. This function does not have a space visible to software.

Configuration cycles on the Hub Interface have the same address format, with the bus number, device number, function number and register number present in the address. Refer to the Hub Interface specification for details of the address.

18.4.1 Type 0 Accesses to the Intel® 6300ESB ICH

The configuration space of the bridge in the Intel® 6300ESB ICH is accessed by a Type 0 configuration transaction on the Hub Interface. The bridge configuration space (the Intel® 6300ESB ICH) responds to a Type 0 configuration transaction when the following conditions are met by the Hub Interface address:

- The bus command is a configuration read or configuration write transaction.
- Low 2 address bits AD[1:0] must be 00b.
- The device number matches one of the Intel® 6300ESB ICH devices (28).

18.4.2 Type 1 to Type 0 Translation

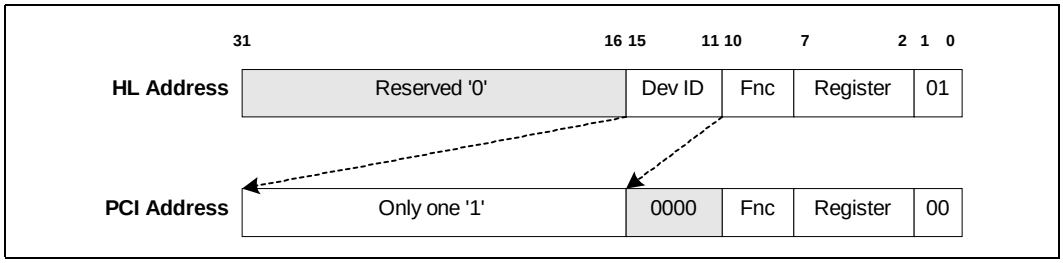
The Intel® 6300ESB ICH performs a Type 1 to Type 0 translation when the Type 1 transaction is generated on the Hub Interface and is intended for a device attached directly to the secondary bus. The Intel® 6300ESB ICH must convert the configuration command to a Type 0 format so that the secondary bus device may respond to it. This

translation is done for cycles that originate on the Hub Interface and target PCI/PCI-X. The Intel® 6300ESB ICH translates a Type 1 configuration transaction into a Type 0 transaction under the following conditions:

- The bus command is a Configuration read or write transaction.
- The low 2 address bits on AD [1:0] are 01b.
- The bus number in address field AD [23:16] is equal to the value in the secondary bus number register in the Intel® 6300ESB ICH configuration space.

The resulting Type 0 address to be driven on PCI-X is shown in Figure 31. Device numbers are decoded to generate a single '1' in address bits 31:16. When the device number is greater than 16, then all bits are '0'.

Figure 31. Type '1' to Type '0' Translation



18.4.3 Type 1 to Type 1 Forwarding

The Intel® 6300ESB ICH forwards a type 1 configuration cycle unchanged to the PCI-X bus under the following conditions.

- The bus command is a configuration read or write transaction.
- The low two address bits are equal to 01b.
- The bus number falls in the range defined by the lower limit (exclusive) in the secondary bus number register and the upper limit (inclusive) in the subordinate bus number register.

Type 1 to type 1 forwarding is only done for cycles from the Hub Interface to PCI-X.

18.4.4 Type 1 to Special Cycle Forwarding

The Intel® 6300ESB ICH translates a type 1 configuration write transaction on the Hub Interface into a special cycle on PCI-X, but does not translate a type 1 configuration access on PCI-X to a special cycle on the Hub Interface. A cycle to be translated has the following attributes in the address:

- The low two address bits on AD[1:0] are equal to 01b.
- The device number in address bits AD[15:11] is equal to 11111b.
- The function number in address bits AD[10:8] is equal to 111b.
- The register number in address bits AD[7:2] is equal to 000000b.
- The bus number is equal to the value in the secondary bus number register in configuration space.
- The bus command is a Configuration Write command.

The address and data are forwarded unchanged. Devices ignore the address and decode only the bus command. The data phase contains the special cycle message. The transaction does a master abort but results in a normal completion on the opposite bus

(normal completion status on the Hub Interface, TRDY# on PCI-X). When more than one data transfer is requested, the Intel® 6300ESB ICH responds with a target disconnect operation during the first data phase.

18.5 Transaction Ordering

18.5.1 Comparison of Rules vs. a PCI – PCI Bridge

When a PCI segment is in PCI (PCI-X) mode, the Intel® 6300ESB ICH follows the producer-consumer model of a PCI – PCI bridge. Table 583 is taken from Appendix E of the *PCI Specification,* Rev 2.2 for PCI, and Section 8.4.4 of the *PCI-X Addendum for PCI-X*. The shaded entries represent differences from that table, and an explanation of the differences:

Table 583. Comparison of Rules vs. A PCI – PCI Bridge

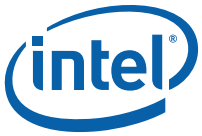
Row Pass Col?	Posted Write	Delayed (Split) Read Request		Delayed (Split) Write Request[3]		Delayed (Split) Read Completion		Delayed (Split) Write Completion[3]	
Posted Write	No	Yes		Yes		Yes		Yes	
Delayed (Split) Read Request	No	Yes[1]	No[2]	Yes[1]	No[2]	No[2]	Yes[2]	Yes	Yes[2]
Delayed (Split) Write Request	No	No[2]		No[2]		No[3]	Yes[2]	No[3]	Yes[2]
Delayed (Split) Read Completion	No	Yes		Yes		No[2]	Yes[2]	No[2]	Yes[2]
Delayed (Split) Write Completion	No[2]	Yes		Yes		No[2]	Yes[2]	No[2]	Yes[2]

NOTES:
1. Subsequent requests only (prefetches). All inbound initial requests are in order.
2. In a bridge, these are allowed to be yes/no.
3. In a bridge, these are allowed to be yes/no. These particular entries are “No” because the Intel® 6300ESB ICH does not accept inbound write requests that are not posted (I/O writes, configuration writes).

18.5.2 Other Notes

Ordering relationships are established for the following classes of transactions crossing the Intel® 6300ESB ICH:

- The Intel® 6300ESB ICH does not combine separate write transactions into a single write transaction.
- The Intel® 6300ESB ICH does not merge bytes on separate write transactions to the same dWord address.
- The Intel® 6300ESB ICH does not collapse sequential write transactions to the same address into a single write transaction – the *PCI Local Bus Specification* does not permit this.



18.6 Device 28 – Hub Interface to PCI-X Bridge

18.6.1 Configuration Space Registers

18.6.1.1 Register Summary

Table 584. Configuration Space Register Summary (Sheet 1 of 2)

Start	End	Symbol	Full Name	Default
00	03	ID	Identifiers	25AE8086h
04	05	CMD	Command	0000h
06	07	PSTS	Primary Status	0030h
08	08	RID	Revision ID	See NOTE:
09	0B	CC	Class Code	060400h
0C	0C	CLS	Cache Line Size	00h
0D	0D	PLT	Primary Latency Timer	00h
0E	0E	HTYPE	Header Type	01h
18	1A	BNUM	Bus Numbers	000000h
1B	1B	SLT	Secondary Latency Timer	00h
1C	1D	IOBL	I/O Base and Limit	0000h
1E	1F	SSTS	Secondary Status	02A0h
20	23	MBL	Memory Base and Limit	00000000h
24	27	PMBL	Prefetchable Memory Base and Limit	00010001h
28	2B	PMBU32	Prefetchable Memory Base Upper 32 Bits	00000000h
2C	2F	PMLU32	Prefetchable Memory Limit Upper 32 Bits	00000000h
30	33	IOBLU16	I/O Base and Limit Upper 16 Bits	00000000h
34	34	CAPP	Capabilities List Pointer	50h
3C	3D	INTR	Interrupt Information	0000h
3E	3F	BCTRL	Bridge Control	0000h
40	41	CNF	Intel® 6300ESB ICH Configuration	000SSh
42	42	MTT	Multi-Transaction Timer	00h
44	47	STRP	PCI Strap Status	00h
50	50	PX_CAPID	PCI-X Capabilities Identifier	07h
51	51	PX_NXTP	Next Item Pointer	00h
52	53	PX_SSTS	PCI-X Secondary Status	0001h
54	57	PX_BSTS	PCI-X Bridge Status	000100D0h
58	5B	PX_USTC	PCI-X Upstream Split Transaction Control	0000FFFFh

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

Table 584. Configuration Space Register Summary (Sheet 2 of 2)

Start	End	Symbol	Full Name	Default
5C	5F	PX_DSTC	PCI-X Downstream Split Transaction Control	0000FFFFh
60	9F		Reserved	
E0	E3	ACNF	Additional Intel® 6300ESB ICH Configuration	0000000Fh
E4	E5	PCR	PCI Compensation Register	0002h
F0	F3	HCCR	Hub Interface Command/Control Register	00000000h
F4	F7		Reserved	
F8	F9	PC33	Prefetch Control – 33 MHz	1111h
FA	FB	PC66	Prefetch Control – 66 MHz	3121h
FC	FF		Reserved	7B7BBFFFh

NOTE: Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

18.6.1.2 Offset 00: ID—Identifiers

Note: Contains the vendor and device identifiers for software.

Table 585. Offset 00: ID—Identifiers

Device 28 ***Function*** 0

Offset 00 ***Attribute:*** Read-Only

Size: 32-bit

Bits	Name	Description	Reset Value	Access
31:16	Device ID (DID)	Indicates what device number was assigned by the PCI SIG.	25AEh	RO
15:00	Vendor ID (VID)	16-bit field which indicates that Intel is the vendor.	8086h	RO

18.6.1.3 Offset 04: CMD—Command

This controls how the device behaves on the primary interface and is the same as all other devices, with the exception of the VGA Palette Snoop bit. As this component is a bridge, additional command information is located in a separate register called “Bridge Control” located at offset 3E.

Table 586. Offset 04: CMD—Command (Sheet 1 of 2)

Device 28	***Function*** 0
Offset 04	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:10	Reserved	Reserved.	00h	RO
09	Fast Back-to-back enable (FBE)	This bit has no meaning on the Hub Interface. It is hardwired to '0'.	0	RO
08	SERR# Enable (SEE)	Controls the enable for assertion of SERR# (via NMI/SMI#) when the SSE bit (D28:F0:Offset 06h, bit 14) is set. See Section 5.1.4 for more details on this bit. 0 = SERR# disabled 1 = SERR# enabled	0	R/W
07	Wait Cycle Control (WCC)	Reserved.	0	RO
06	Parity Error Response Enable (PERE)	Controls the Intel® 6300ESB ICH's response when a parity error is detected on the Hub Interface. 0 = The Intel® 6300ESB ICH ignores these errors on the Hub Interface. 1 = The Intel® 6300ESB ICH reports these errors on the Hub Interface and sets the DPD bit in the status register. **NOTE:** The Hub Interface Parity Unsupported bit (D30:F0:40h:bit 20) must be cleared for the PER bit to have any effect.	0	R/W
05	VGA Palette Snoop Enable (VGA_PSE)	Reserved.	0	RO
04	Memory Write and Invalidate Enable (MWIE)	The Intel® 6300ESB ICH does not generate memory write and invalidate transactions, as the Hub Interface does not have a corresponding transfer type.	0	RO
03	Special Cycle Enable (SCE)	Reserved.	0	RO

Table 586. Offset 04: CMD—Command (Sheet 2 of 2)

Device 28	***Function*** 0
Offset 04	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
02	Bus Master Enable (BME)	Controls the ability of the Intel® 6300ESB ICH to act as a master on the Hub Interface when forwarding memory transactions from PCI-X. When '0': the Intel® 6300ESB ICH does not respond to any memory transactions on the PCI-X interface that target Hub Interface.	0	R/W
01	Memory Space Enable (MSE)	Controls the ability of the Intel® 6300ESB ICH to respond as a target to memory accesses on the Hub Interface that address a device behind the Intel® 6300ESB ICH.	0	R/W
0	I/O Space Enable (IOSE)	Controls the ability of the Intel® 6300ESB ICH to respond as a target to I/O transactions on the primary interface that address a device that resides behind the Intel® 6300ESB ICH.	0	R/W

18.6.1.4 Offset 06: PSTS—Primary Status

Note: For the writable bits in this register, writing a '1' clears the bit. Writing a '0' has no effect.

Note: RASERR# will be asserted based on activity of bits 15:11, 8.

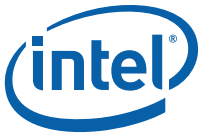


Table 587. Offset 06: PSTS—Primary Status

Device 28	**Function** 0
Offset 06	**Attribute:** Read/Write Clear
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15	Detected Parity Error (DPE)	When set to 1, this bit indicates that the Intel® 6300ESB ICH detected an address parity, data parity, error on the Hub Interface. This bit gets set even when the Parity Error Response bit (bit 6 of the command register) is not set. Note that each bridge sets this bit, regardless of address. **NOTE:** The Hub Interface Parity Unsupported bit (D30:F0:40h:bit 20) must be cleared for the PER bit to have any effect.	0	R/WC
14	Signaled System Error (SSE)	This bit is set to '1' when the SERR# is reported to the Hub Interface through the NMI/SMI# assertion when enabled.	0	R/WC
13	Received Master Abort (RMA)	This bit is set whenever the Intel® 6300ESB ICH is acting as master on the Hub Interface and receives a completion packet with master abort status.	0	R/WC
12	Received Target Abort (RTA)	This bit is set whenever the Intel® 6300ESB ICH is acting as master on the Hub Interface and receives a completion packet with target abort status.	0	R/WC
11	Signaled Target Abort (STA)	This bit is set whenever the Intel® 6300ESB ICH generates a completion packet with target abort status.	0	R/WC
10:09	DEVSEL# Timing (DVT)	These bits have no meaning on the Hub Interface. Fast decode timing is reported.	00	RO
08	Data Parity Error Detected (DPD)	This bit is set when the Intel® 6300ESB ICH receives a completion packet from the Hub Interface from a previous request and detects a parity error, and the Parity Error Response bit in the Command Register (offset 04h, bit 6) is set.	0	R/WC
07	Fast Back-to-Back Capable (FBC)	This bit has no meaning on the Hub Interface.	0	RO
06	Reserved	Reserved.	0	RO
05	66 MHz Capable (C66)	This bit has no meaning on the Hub Interface but is set to be true in case of any software dependencies on bandwidth calculations.	1	RO
04	Capabil-ities List Enable (CAPE)	Indicates that the Intel® 6300ESB ICH contains the capabilities pointer in the bridge. Offset 34h indicates the offset for the first entry in the linked list of capabilities.	1	RO
03:0	Reserved	Reserved.	0h	RO

18.6.1.5 Offset 08: RID—Revision ID

Table 588. Offset 08: RID—Revision ID

Device 28		***Function*** 0	
Offset 08		***Attribute:*** Read-Only	
		Size: 8-bit	

Bits	Name	Description	Reset Value	Access
07:0	Revision ID (RID)	Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.	00h	RO

18.6.1.6 Offset 09: CC—Class Code

Note: This contains the class code, sub class code, and programming interface for the device.

Table 589. Offset 09: CC—Class Code

Device	28	***Function***	0
Offset	09	***Attribute:***	Read-Only
		Size:	24-bit

Bits	Name	Description	Reset Value	Access
23:16	Base Class Code (BCC)	The value of 06h indicates that this is a bridge device.	06h	RO
15:08	Sub Class Code (SCC)	8-bit value that indicates this is of type PCI-PCI bridge.	04h	RO
07:0	Programming Interface (PIF)	Indicates that this is standard (non-subtractive) PCI-PCI bridge.	00h	RO

18.6.1.7 Offset 0C: CLS—Cache Line Size

Note: This indicates the cache line size of the system.

Table 590. Offset 0C: CLS—Cache Line Size

Device	28	***Function***	0
Offset	0C	***Attribute:***	Read/Write
		Size:	8-bit

Bits	Name	Description	Reset Value	Access
07:0	Cache Line Size (CLS)	The value in this register is used by the Intel® 6300ESB ICH to determine the size of packets on the Hub Interface. This read/write register specifies the system cache line size in units of dWords. When the value is '08h', represents a 32-byte line (8 dWords). A value of '10h' represents a 64-byte line, and a value of '20h' represents a 128-byte line. Any value outside this range defaults to a 64-byte line. When the Intel® 6300ESB ICH is creating read and write requests to the Hub Interface, this value is used to partition the requests such that multiple snoops for the same line are avoided in the memory subsystem.	00h	R/W

18.6.1.8 Offset 0D: PLT—Primary Latency Timer

Note: This register does not apply to Hub Interface and is maintained as R/W for software compatibility.

Table 591. Offset 0D: PLT—Primary Latency Timer

Device	28	***Function***	0
Offset	0D	***Attribute:***	Read/Write
		Size:	8-bit

Bits	Name	Description	Reset Value	Access
07:03	Time Value (TV)	Read/write for software compatibility only.	00h	R/W
02:0	Reserved	Reserved.	000	RO

18.6.1.9 Offset 0E: HTYPE—Header Type

Note: This register determines how the rest of the configuration space is laid out.

Table 592. Offset 0E: HTYPE—Header Type

Device	28	***Function***	0
Offset	0E	***Attribute:***	Read-Only
		Size:	8-bit

Bits	Name	Description	Reset Value	Access
07	Multi-function device (MFD)	Reserved as '0' to indicate the bridge is a single function device.	0	RO
06:0	Header Type (HTYPE)	Defines the layout of addresses 10h through 3Fh in configuration space. Reads as '01h' to indicate that the register layout conforms to the standard PCI-to-PCI bridge layout.	01h	RO

18.6.1.10 Offset 18: BNUM—Bus Numbers

Note: This contains the primary, secondary, and maximum subordinate bus number registers.

Table 593. Offset 18: BNUM—Bus Numbers

Device 28	***Function*** 0
Offset 18	***Attribute:*** Read/Write
	Size: 24-bit

Bits	Name	Description	Reset Value	Access
23:16	Subordinate Bus Number (SBBN)	Indicates the highest PCI bus number below this bridge. Any type one configuration cycle on the Hub Interface whose bus number is greater than the secondary bus number and less than or equal to the subordinate bus number is run as a type one configuration cycle on the PCI bus.	00h	R/W
15:08	Secondary Bus Number (SCBN)	Indicates the bus number of PCI to which the secondary interface is connected. Any type one configuration cycle matching this bus number is translated to a type 0 configuration cycle and run on the PCI bus.	00h	R/W
07:00	Primary Bus Number (PBN)	Indicates the bus number of the Hub Interface. Any type 1 configuration cycle with a bus number less than this number is not accepted by this portion of the Intel® 6300ESB ICH (i.e., it still may match the other bridge).	00h	R/W

18.6.1.11 Offset 1B: SLT—Secondary Latency Timer

Note: This timer controls the amount of time that the Intel® 6300ESB ICH continues to burst data on its secondary interface. The counter starts counting down from the assertion of PXFRAME#. When the grant is removed, the expiration of this counter results in the deassertion of PXFRAME#. When the grant has not been removed, the Intel® 6300ESB ICH may continue ownership of the bus. The secondary latency timer's default value should be 64 in PCI-X mode (Section 8.6.1 of the *PCI-X 1.0 Specification*).

Table 594. Offset 1B: SLT—Secondary Latency Timer

Device 28	***Function*** 0
Offset 1B	***Attribute:*** Read/Write
	Size: 8-bit

Bits	Name	Description	Reset Value	Access
07:03	Secondary Latency Timer (TV)	A five-bit value that indicates the number of PCI clocks, in 8-clock increments, that the Intel® 6300ESB ICH remains as a master of the PCI bus when another master is requesting use of the PCI bus.	00h	R/W
02:00	Reserved	Reserved.	000	RO

18.6.1.12 Offset 1C: IOBL—I/O Base and Limit

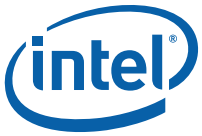
Note: Defines the base and limit, aligned to a 4 Kbyte boundary, of the I/O area of the bridge. Accesses from the Hub Interface that are within the ranges specified in this register are sent to PCI-X when the I/O space enable bit is set.

Note: Accesses from PCI-X that are outside the ranges specified result in a master abort.

Table 595. Offset 1C: IOBL—I/O Base and Limit

Device 28	***Function*** 0
Offset 1C	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:12	I/O Limit Address Bits [15:12] (IOLA)	Defines the top address of an address range to determine when to forward I/O transactions from one interface to the other. These bits correspond to address lines 15:12 for 4 Kbyte alignment. Bits 11:0 are assumed to be FFFh.	0h	R/W
11:10	I/O Limit Address Bits [11:10] (IOLA1K)	When the EN1K bit is set in the Intel® 6300ESB ICH Configuration register (CNF), these bits become read/write and are compared with I/O address bits 11:10 to determine the 1K limit address. When the EN1K bit is cleared, this field becomes **Read Only**.	00	R/W
09:08	I/O Limit Addressing Capability (IOLC)	This is hard-wired to 0h, indicating support for only 16-bit I/O addressing.	0h	RO
07:04	I/O Base Address Bits [15:12] (IOBA)	Defines the bottom address of an address range to determine when to forward I/O transactions from one interface to the other. These bits correspond to address lines 15:12 for 4 Kbyte alignment. Bits 11:0 are assumed to be 000h.	0h	R/W
03:02	I/O Base Address Bits [11:10] (IOBA1K)	When the EN1K bit is set in the Intel® 6300ESB ICH Configuration register (CNF), these bits become read/write and are compared with I/O address bits 11:10 to determine the 1K base address. When the EN1K bit is cleared, this field becomes **Read Only**.	00	R/W
01:00	I/O Base Addressing Capability (IOBC)	This is hard-wired to 0h, indicating support for only 16-bit I/O addressing.	0h	RO



18.6.1.13 Offset 1E: SSTS—Secondary Status

Note: For the writable bits in this register, writing a '1' clears the bit. Writing a '0' to the bit has no effect.

Note: RASERR# will be asserted based on activity of bits 15:11, 8.

Table 596. Offset 1E: SSTS—Secondary Status (Sheet 1 of 2)

Device 28	***Function*** 0
Offset 1E	***Attribute:*** Read/Write Clear
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15	Detected Parity Error (DPE)	This bit is set to a '1' whenever the Intel® 6300ESB ICH detects a address or data parity error on the PCI-X bus. This bit gets set even when the Parity Error Response bit (bit '0' of offset 3E-3F) is not set.	0	R/WC
14	Received System Error (RSE)	The Intel® 6300ESB ICH sets this bit when a SERR# assertion is received on PCI-X.	0	R/WC
13	Received Master Abort (RMA)	This bit is set whenever the Intel® 6300ESB ICH is acting as an initiator on the PCI-X bus and the cycle is master-aborted. For Hub Interface packets that have completion required, this should also cause a target abort completion status to be returned and set the Signaled Target Abort bit in the primary status register.	0	R/WC
12	Received Target Abort (RTA)	This bit is set whenever the Intel® 6300ESB ICH is acting as an initiator on PCI-X and a cycle is target-aborted on PCI-X. For "completion required" Hub Interface packets, this event should force a completion status of "target abort" on the Hub Interface and set the Signaled Target Abort in the Primary Status Register.	0	R/WC
11	Signaled Target Abort (STA)	This bit is set to '1' when the Intel® 6300ESB ICH is acting as a target on the PCI-X Bus and signals a target abort.	0	R/WC
10:9	DEVSEL# Timing (DVT)	Indicates that the Intel® 6300ESB ICH responds in medium decode time to all cycles targeting the Hub Interface.	01	RO
8	Data Parity Error Detected (DPD)	The Intel® 6300ESB ICH sets this bit when all of the following are true: • The Intel® 6300ESB ICH is the initiator on PCI-X. • PERR# is detected asserted or a parity error is detected internally. • The Parity Error Response Enable bit in the Bridge Control Register (bit 0, offset 3Eh) is set.	0	R/WC
7	Fast Back-to-Back Capable (FBC)	Indicates that the secondary interface of the Intel® 6300ESB ICH may receive fast back-to-back cycles.	1	RO
6	Reserved	Reserved.	0	RO

Table 596. Offset 1E: SSTS—Secondary Status (Sheet 2 of 2)

Device	28	**Function**	0
Offset	1E	**Attribute:**	Read/Write Clear
		Size:	16-bit

Bits	Name	Description	Reset Value	Access
5	66 MHz Capable (C66)	Indicates the secondary interface of the bridge is 66 MHz capable.	1	RO
4	PERR# Assertion Detect	This bit is set by hardware whenever the PERR# pin is asserted on the rising edge of PCI-X clock. This includes cases in which the chipset is the agent driving PERR#. It remains asserted until cleared by software writing a '1' to this location. When enabled by the PERR#-to-SERR# Enable bit (in the Bridge Control register), a '1' in this bit can generate an internal SERR# and be a source for the NMI logic.		R/WC
3:0	Reserved	Reserved.	00h	RO

18.6.1.14 Offset 20: MBL—Memory Base and Limit

Note: Defines the base and limit, aligned to a 1 Mbyte boundary, of the memory area of the bridge. Accesses from the Hub Interface that are within the ranges specified in this register are sent to PCI-X when the memory space enable bit is set.

Note: Accesses from PCI-X that are outside the ranges specified are forwarded to the Hub Interface when the bus master enable bit is set.

Table 597. Offset 20: MBL—Memory Base and Limit

Device	28	**Function**	0
Offset	20	**Attribute:**	Read/Write
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:20	Memory Limit (ML)	These bits are compared with bits 31:20 of the incoming address to determine the upper 1 Mbyte aligned value (exclusive) of the range. The incoming address must be less than or equal to this value.	000h	R/W
19:16	Reserved	Reserved.	0h	RO
15:04	Memory Base (MB)	These bits are compared with bits 31:20 of the incoming address to determine the lower 1 Mbyte aligned value (inclusive) of the range. The incoming address must be greater than or equal to this value.	000h	R/W
03:00	Reserved	Reserved.	0h	RO

18.6.1.15 Offset 24: PMBL—Prefetchable Memory Base and Limit

Note: Defines the base and limit, aligned to a 1 Mbyte boundary, of the prefetchable memory area of the bridge. Accesses from the Hub Interface that are within the ranges specified in this register are sent to PCI-X when the memory space enable bit is set.

Note: Accesses from PCI-X that are outside the ranges specified are forwarded to the Hub Interface when the bus master enable bit is set.

Table 598. Offset 24: PMBL—Prefetchable Memory Base and Limit

Device	28	***Function***	0
Offset	24	***Attribute:***	Read/Write
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:20	Prefetchable Memory Limit (PML)	These bits are compared with bits 31:20 of the incoming address to determine the upper 1 Mbyte aligned value (exclusive) of the range. The incoming address must be less than this value.	000h	R/W
19:16	64-bit Indicator (IS64L)	Indicates that 32-bit addressing is supported for the limit. This value must be in agreement with the IS64B field.	1h	RO
15:04	Prefetchable Memory Base (PMB)	These bits are compared with bits 31:20 of the incoming address to determine the lower 1 Mbyte aligned value (inclusive) of the range. The incoming address must be greater than or equal to this value.	000h	R/W
03:00	64-bit Indicator (IS64B)	Indicates that 32-bit addressing is supported for the limit. This value must be in agreement with the IS64L field.	1h	RO

18.6.1.16 Offset 28: PMBU32—Prefetchable Memory Base Upper 32 Bits

Note: This defines the upper 32 bits of the prefetchable address base register.

Table 599. Offset 28: PMBU32—Prefetchable Memory Base Upper 32 Bits

Device	28	***Function***	0
Offset	28	***Attribute:***	Read/Write
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:00	Prefetchable Memory Base Upper Portion (PMBU)	All bits are read/writable. This register should always be programmed to 00000000h since the Intel® 6300ESB ICH only supports 32-bit downstream addressing and 64-bit upstream addressing	00000000h	R/W

18.6.1.17 Offset 2C: PMLU32—Prefetchable Memory Limit Upper 32 Bits

Note: This defines the upper 32 bits of the prefetchable address limit register.

Table 600. Offset 2C: PMLU32—Prefetchable Memory Limit Upper 32 Bits

Device	28	***Function***	0
Offset	2C	***Attribute:***	Read/Write
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:00	Prefetchable Memory Limit Upper Portion (PMLU)	All bits are read/writable - the Intel® 6300ESB ICH supports full 64-bit addressing. **NOTE:** The upper 32 bits should not be used to determine the prefetch region. The Intel® 6300ESB ICH supports only 32-bit downstream cycles, so the upper 32 bits of the prefetch region are ignored. Prefetch regions are limited to a single 4 Gbyte boundary. The upper 32 bits of the prefetch region cannot be used to extend this region. The Intel® 6300ESB ICH supports 64-bit upstream cycles, although the upper 32 bits are not used to determined the destination if the target lies within the Intel® 6300ESB ICH.	00000000h	R/W

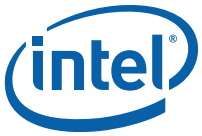
18.6.1.18 Offset 30: IOBLU16—I/O Base and Limit Upper 16 Bits

Note: Since I/O is limited to 64 Kbytes, this register is reserved and not used.

Table 601. Offset 30: IOBLU16—I/O Base and Limit Upper 16 Bits

Device	28	***Function***	0
Offset	30	***Attribute:***	Read-Only
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:16	I/O Base High 16 Bits (IOBH)	Reserved.	0000h	RO
15:00	I/O Limit High 16 Bits (IOLH)	Reserved.	0000h	RO



18.6.1.19 Offset 34: CAPP—Capabilities List Pointer

Note: Contains the pointer for the first entry in the capabilities list.

Table 602. Offset 34: CAPP—Capabilities List Pointer

Device 28		***Function*** 0	
Offset 34		***Attribute:*** Read-Only	
		Size: 8-bit	

Bits	Name	Description	Reset Value	Access
07:00	Capabil-ities Pointer (PTR)	Indicates that the pointer for the first entry in the capabilities list is at 50h in configuration space.	50h	RO

18.6.1.20 Offset 3C: INTR—Interrupt Information

Note: This register contains information on interrupts on the bridge.

Table 603. Offset 3C: INTR—Interrupt Information

Device 28		***Function*** 0	
Offset 3C		***Attribute:*** Read-Only	
		Size: 16-bit	

Bits	Name	Description	Reset Value	Access
15:08	Interrupt Pin (PIN)	Bridges do not support the generation of interrupts.	00h	RO
07:00	Interrupt Line (LINE)	The Intel® 6300ESB ICH Bridge does not generate interrupts, so this is reserved as 00h.	00h	RO

18.6.1.21 Offset 3E: BCTRL—Bridge Control

Note: This register provides extensions to the Command register that are specific to a bridge. The Bridge Control register provides many of the same controls for the secondary interface that are provided by the Command register for the primary interface. Some bits affect operation of both interfaces of the bridge.

Table 604. Offset 3E: BCTRL—Bridge Control (Sheet 1 of 3)

Device 28	**Function** 0
Offset 3E	**Attribute:** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:11	Reserved	Reserved.	0h	RO
12	PERR# to SERR# Enable	When this bit is set to '1', PCI-X PERR NMI reporting is enabled. In addition to setting this bit, you also must set bit '1' of D30_F0 PNE Register. Section 7.1.28 When this bit is set to a '1' and PERR# is asserted on PCI-X, the PERR# Assertion detect status bit in the Secondary Status Register will indicate a PERR# internal SERR# assertion. The SERR# can be a s source on NMI.		
11	Discard Timer SERR# Enable (DTSE)	Controls the generation of SERR# on the primary interface in response to a timer discard on the secondary interface. When 0: Do not generate SERR# on a secondary timer discard When 1: Generate SERR# in response to a secondary timer discard	0	R/W
10	Discard Timer Status (DTSb)	This bit is set to a '1' when the secondary discard timer expires (there is no discard timer for the primary interface).	0	R/WC
09	Secondary Discard Timer (SDT)	Sets the maximum number of PCI clock cycles that the Intel® 6300ESB ICH waits for an initiator on PCI to repeat a delayed transaction request. The counter starts once the delayed transaction completion is at the head of the queue. If the master has not repeated the transaction at least once before the counter expires, the Intel® 6300ESB ICH discards the transaction from its queues. When 0: The PCI master timeout value is between *2^15* and *2^16* PCI clocks. When 1: The PCI master timeout value is between *2^10* and *2^11* PCI clocks	0	R/W
08	Primary Discard Timer (PDT)	Not relevant to Hub Interface. This bit is R/W for software compatibility only.	0	R/W
07	Fast Back-to-Back Enable (FBE)	The Intel® 6300ESB ICH cannot generate fast back-to-back cycles on the PCI-X bus from Hub Interface initiated transactions.	0	RO

Table 604. Offset 3E: BCTRL—Bridge Control (Sheet 2 of 3)

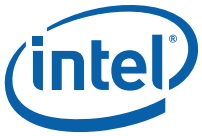
Device 28	**Function** 0
Offset 3E	**Attribute:** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
06	Secondary Bus Reset (SBR)	Controls PXPCIRST# assertion on PCI-X bus when SBRE is set. See Section 18.6.1.32, "Offset E4: PCR - PCI Compensation Register" for SBRE details. 1 = The Intel® 6300ESB ICH asserts PCIXSBRST#. When PCIXSBRST# is asserted, the data buffers between the Hub Interface and PCI-X and the PCI-X bus are initialized back to reset conditions. The Hub Interface and the configuration registers are not affected. To be effective, software must keep asserted for at least 100 µsecs. 0 = The Intel® 6300ESB ICH de-asserts PCIXSBRST#	0	R/W
05	Master Abort Mode (MAM)	Controls the Intel® 6300ESB ICH's behavior when a master abort occurs on either interface. **Master Abort on Hub Interface:** When 0: The Intel® 6300ESB ICH asserts TRDY# on PCI-X. It drives all '1's for reads and discards data on writes. When 1: The Intel® 6300ESB ICH returns a target abort on PCI-X. **Master Abort PCI/PCI-X:** *(Completion required packets only)* When 0: Normal completion status is returned on the Hub Interface. When 1: Target abort completion status is returned on the Hub Interface.	0	R/W
04	VGA 16-bit Decode	Enables the bridge to provide 16 bits decoding of VGA I/O address precluding the decode of VGA alias addresses every 1 KB. This bit requires the VGA enable bit (bit 3 of this register) to be set 1.	0	R/W
03	VGA Enable (VGAE)	Modifies the Intel® 6300ESB ICH's response to VGA compatible address. When set to a 1, the Intel® 6300ESB ICH forwards the following transactions from the Hub Interface to PCI-X regardless of the value of the I/O base and limit registers. The transactions are qualified by the memory enable and I/O enable in the command register. **Memory addresses:** 000A0000h-000BFFFFh **I/O addresses:** 3B0h-3BBh and 3C0h-3DFh. For the I/O addresses, bits [63:16] of the address must be 0, and bits [15:10] of the address are ignored (i.e., aliased). The same holds true from secondary accesses to the primary interface in reverse. That is, when the bit is 0, memory and I/O addresses on the secondary interface between the above ranges are forwarded to the Hub Interface.	0	R/W

Table 604. Offset 3E: BCTRL—Bridge Control (Sheet 3 of 3)

Device 28	***Function*** 0
Offset 3E	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
02	ISA Enable (IE)	Modifies the response by the bridge to ISA I/O addresses. This only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64 Kbytes of PCI-X I/O space. When this bit is set, the bridge blocks any forwarding from primary to secondary of I/O transactions addressing the last 768 bytes in each 1 Kbyte block (offsets 100h to 3FFh). This bit has no effect on transfers originating on the secondary bus as the Intel® 6300ESB ICH does not forward I/O transactions across the bridge.	0	R/W
01	PXSERR# Enable (SE)	When set, the bridge is enabled for SERR reporting.	0	R/W
0	Parity Error Response Enable (PERE)	Controls the Intel® 6300ESB ICH's response to address and data parity errors on the secondary interface. When the bit is cleared, the bridge must ignore any parity errors that it detects and continue normal operation. The Intel® 6300ESB ICH must generate parity even when parity error reporting is disabled.	0	R/W



18.6.1.22 Offset 40: CNF—Intel® 6300ESB I/O Controller Hub Configuration

Table 605. Offset 40: CNF—Intel® 6300ESB I/O Controller Hub Configuration (Sheet 1 of 3)

Device 28	***Function*** 0
Offset 40	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15	Reserved	Reserved.	0	RO
14:10	Disable PXPCLKOUT 4- 0 (DPCLK)	Disables a PCI-X clock output that is not used in the system. Bit 10 refers to PCLKOUT0, bit 11 to PCLKOUT1, etc. When disabled, the PCLKOUT pin is tri-stated. **NOTE:** Bit 14 controls the feedback path. This bit should not be written to a '1'.	00h	R/W
09	Enable I/O Space to 1K Granularity (EN1K)	When set, this enables the I/O space to be decoded to 1K, down from the 4K limit that currently exists in the I/O base and limit registers. It does this by redefining bits [11:10] and bits [3:2] of the IOBL register at offset 1C to be read/write and enables them to be compared with I/O address bits [11:10] to determine if they are within the bridge's I/O range.	0	R/W
08	PCI-X Mode (PMODE)	When set, indicates the bus is operating in PCI-X mode. When cleared, indicates the bus is in PCI mode. The power up value of this register is written based upon the table below: **PCIXCAP=Bit 7 M66EN=Bit 6** **PCIXCAP / M66EN / PCI/XFreq / PMODE** 0 / 0 / PCI / 33 MHz / 0 0 / 1 / PCI / 66 MHz / 0 Mid / N/A / PCI-X / 66 MHz / 1 1 / N/A / PCI-X / 66 MHz / 1 The default value (S-select value) is determined by the values of M66EN and PCIXCAP pins as per the table above. **NOTE:** This register should not be written to and should be treated by Software as Read Only. Writes will change the register value rendering the contents invalid since the value will not affect the PCI-X Mode.		R/W

Table 605. Offset 40: CNF—Intel® 6300ESB I/O Controller Hub Configuration (Sheet 2 of 3)

Device 28	***Function*** 0
Offset 40	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
07:06	PCI-X Frequency (PFREQ)	Determines the frequency the PCI-X bus operates. The power up value of this register is written based upon the following table: **PCIXCAP=Bit 7M66EN=Bit 6** **PCIXCAP M66EN PCI/X Freq** **0 0 PCI 33 MHz** **0 1 PCI 66 MHz** **Mid N/A PCI-X 66 MHz** **1 N/A PCI-X 66 MHz** After software determines the buses' capabilities, it sets this value and the PMODE bit (bit 8 of this register) to the desired frequency and resets the PCI-X bus. The values are encoded as follows: **Bits Freq (MHz) Notes** 0 33 Only valid when PMODE is 0. 1 66 10 not supported 11 not supported The default value (S-select value) is determined by the values of M66EN and PCIXCAP pins as per the table above. **NOTE:** This register should not be written to and should be treated by Software as Read Only. Writes will change the register value rendering the contents invalid since the value will not affect the PCI-X Mode.	0h	R/W
05	Restreaming Disable (RSDIS)	When this bit is set, this bridge of the Intel® 6300ESB ICH no longer performs restreaming. This bit only applies when the bridge is in PCI mode, and not when the bridge is in PCI-X mode. When the PCI transaction ends, either due to a PCI master removing PXFRAME# or the Intel® 6300ESB ICH asserting STOP#, the Intel® 6300ESB ICH discards all data in the prefetch buffer.	0	R/W

Table 605. Offset 40: CNF—Intel® 6300ESB I/O Controller Hub Configuration (Sheet 3 of 3)

Device 28	**Function** 0
Offset 40	**Attribute:** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
04:03	Prefetch Policy (PP)	Controls how the Intel® 6300ESB ICH prefetches data on behalf of PCI masters: 00: Allow prefetching on MRM, MRL, and MR. 01: Allow prefetching on MRM and MRL but not on a memory read. 1x: Disable all prefetching.	00	R/W
02	Delayed Transaction Depth (DTD)	Controls the Intel® 6300ESB ICH behavior relative to the number and size of the delayed transaction buffers: When 0: 4 DTs at 1K for 33/66 MHz When 1: 4 DTs at 1K for all frequencies This bit is set by platform BIOS, based upon the PCI frequency read from bits 8:6 of this register.	0	R/W
01:00	Maximum Delayed Transac-tions (MDT)	Controls the maximum number of delayed transactions the Intel® 6300ESB ICH is allowed to have: **00**: 4 active, 4 pending **01**: 1 active, 1 pending **10**: 2 active, 2 pending **11**: Reserved	00	R/W

18.6.1.23 Offset 42: MTT—Multi-Transaction Timer

Note: This register controls the amount of time that the Intel® 6300ESB ICH's arbiter allows a PCI initiator to perform multiple back-to-back transactions on the PCI-X bus. The number of clocks programmed in the MTT represents the ensured time slice (measured in PCI clocks) allotted to the current agent, after which the arbiter grants another agent that is requesting the bus.

Table 606. Offset 42: MTT—Multi-Transaction Timer

Device 28	**Function** 0
Offset 42	**Attribute:** Read/Write
	Size: 8-bit

Bits	Name	Description	Reset Value	Access
07:03	Timer Count Value (MTC)	This field specifies the amount of time that grant remains asserted to a master continuously asserting its request for multiple transfers. This field specifies the count in an 8-clock (PCI clock) granularity.	00h	R/W
02:00	Reserved	Reserved.	000	RO

18.6.1.24 Offset 44: STRP—PCI Strap Status

Note: This register indicates the states of various straps for this PCI-X interface.

Table 607. Offset 44: STRP—PCI Strap Status

Device 28		***Function*** 0	
Offset 44		***Attribute:*** Read-Only	
		Size: 32-bit	

Bits	Name	Description	Reset Value	Access
31:0	Reserved	Reserved.		RO

18.6.1.25 Offset 50: PX_CAPID—PCI-X Capabilities Identifier

Note: Identifies this item in the Capabilities list as a PCI-X register set. It returns 07h when read.

Table 608. Offset 50: PX_CAPID—PCI-X Capabilities Identifier

Device 28		***Function*** 0	
Offset 50		***Attribute:*** Read-Only	
		Size: 8-bit	

Bits	Name	Description	Reset Value	Access
07:00	Identifier (ID)	Indicates this is a PCI-X capabilities list.	07h	RO

18.6.1.26 Offset 51: PX_NXTP—Next Item Pointer

Note: Indicates where the next item in the capabilities list resides. This is the end of the list and 00h is returned.

Table 609. Offset 51: PX_NXTP—Next Item Pointer

Device 28		***Function*** 0	
Offset 51		***Attribute:*** Read-Only	
		Size: 8-bit	

Bits	Name	Description	Reset Value	Access
07:00	Reserved	Reserved.	00h	RO

18.6.1.27 Offset 52: PX_SSTS—PCI-X Secondary Status

Note: This is the PCI-X command register which controls various modes of the bridge.

Table 610. Offset 52: PX_SSTS—PCI-X Secondary Status (Sheet 1 of 2)

Device 28 ***Function*** 0

Offset 52 ***Attribute:*** Read-Only

Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:09	Reserved	Reserved.	00h	RO
08:06	Secondary Clock Frequency (SCF)	This field is set with the frequency of the secondary bus. The values are: **Bits Max Frequency Clock Period** 000 PCI Mode N/A 001 66 15 010 [Not supported] 011 [Not supported] 1xx Reserved Reserved The default value for this register is given by **PCIXCAP=Bit 7 M66EN=Bit 6** **PCIXCAP M66EN PCI/X Freq** 0 0 PCI 33 MHz 0 1 PCI 66 MHz Mid N/A PCI-X 66 MHz 1 N/A PCI-X 66 MHz	xxx	RO
05	Split Request Delayed. (SRD)	This bit is set by a bridge when it cannot forward a transaction on the secondary bus to the primary bus because there is not enough room within the limit specified in the Split Transaction Commitment Limit field in the Downstream Split Transaction Control register. **NOTE:** The Intel® 6300ESB ICH does not set this bit.	0	RO
04	Split Completion Overrun (SCO)	This bit is set when a bridge terminates a Split Completion on the secondary bus with retry or Disconnect at next ADB because its buffers are full. **NOTE:** The Intel® 6300ESB ICH does not set this bit.	0	RO
03	Unexpected Split Completion (USC)	This bit is set when an unexpected split completion with a requester ID equal to the Intel® 6300ESB ICH's secondary bus number, device number 00h, and function number 0 is received on the secondary interface. This bit is cleared by software writing a '1'.	0	R/WC

Table 610. Offset 52: PX_SSTS—PCI-X Secondary Status (Sheet 2 of 2)

Device 28	**Function** 0
Offset 52	**Attribute:** Read-Only
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
02	Split Completion Discarded (SCD)	This bit is set when the Intel® 6300ESB ICH discards a split completion moving toward the secondary bus because the requester would not accept it. This bit is cleared by software writing a '1'.	0	R/WC
01	133 MHz Capable (C133)	This bit indicates that the Intel® 6300ESB ICH's secondary interface is capable of 133 MHz operation in PCI-X mode. 0 = Not capable 1 = Capable	0	RO
00	64-bit Device (D64)	Indicates the width of the secondary bus as 64-bits.	1	RO

18.6.1.28 Offset 54: PX_BSTS - PCI-X Bridge Status

Note: Identifies PCI-X capabilities and current operating mode of the bridge.

Table 611. Offset 54: PX_BSTS - PCI-X Bridge Status

Device 28	**Function** 0
Offset 54	**Attribute:** Read-Only
	Size: 32-bit

Bits	Name	Description	Reset Value	Access
31:22	Reserved	Reserved.	0	RO
21	Split Request Delayed (SRD)	The Intel® 6300ESB ICH does not support this bit.	0	RO
20	Split Completion Overrun (SCO)	The Intel® 6300ESB ICH does not set this bit because it does not request more data on the Hub Interface than it may receive.	0	RO
19	Unexpected Split Completion (USC)	This does not apply to Hub Interface, which is the primary interface.	0	RO

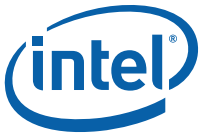


Table 611. Offset 54: PX_BSTS - PCI-X Bridge Status

Device 28	***Function*** 0
Offset 54	***Attribute:*** Read-Only
	Size: 32-bit

Bits	Name	Description	Reset Value	Access
18	Split Completion Discarded (SCD)	This does not apply to Hub Interface.	0	RO
17	Reserved	Reserved.	0	RO
16	64-bit Device (D64)	Default value is 1. The Hub Interface is a 64-bit interface (in HI2.0, it is 128 bits).	1	RO
15:08	Bus Number (BNUM)	An alias to the PBN field of the BNUM register at offset 18h. Available for diagnostic software.	00h	RO
07:03	Device Number (DNUM)	Default value is device 28. Readable from separate PCI-X diagnostic software.	1Ch	RO
02:00	Function Number (FNUM)	Read-only bits for PCI-X diagnostic software.	0h	RO

18.6.1.29 Offset 58: PX_USTC - PCI-X Upstream Split Transaction Control

Note: This register identifies controls behavior of the PCI-X Upstream Split Control buffers for forwarding Split Transactions from the secondary bus to the Hub Interface.

Note: The Intel® 6300ESB ICH maintains these registers internally; programming is not required by end users.

Table 612. Offset 58: PX_USTC - PCI-X Upstream Split Transaction Control

Device	28	***Function***	0
Offset	58	***Attribute:***	Read/Write
		Size:	32-bit

Bits	Name	Description	Reset Value	Access
31:16	Split Transaction Limit (STL)	R/W field available for use by diagnostic software. **NOTE:** Not used by the Intel® 6300ESB ICH for modifying its "commitment" level. The Intel® 6300ESB ICH internal launch algorithms keep buffers from being overallocated.	0000h	R/W
15:00	Split Transaction Capacity (STC)	Infinite capacity due to launch algorithm keeping buffers from overrunning. The Intel® 6300ESB ICH internal launch algorithms keep buffers from being overallocated.	FFFFh	RO

18.6.1.30 Offset 5C: PX_DSTC - PCI-X Downstream Split Transaction Control

Note: This register controls behavior of the PCI-X Downstream Split Control buffers for forwarding Split Transactions from the Hub Interface to the secondary bus.

Note: The Intel® 6300ESB ICH maintains these registers internally, programming not required by end users.

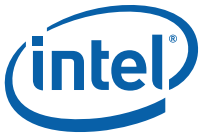


Table 613. Offset 5C: PX_DSTC - PCI-X Downstream Split Transaction Control

Device 28	***Function*** 0
Offset 5C	***Attribute:*** Read/Write
	Size: 32-bit

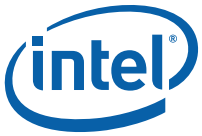
Bits	Name	Description	Reset Value	Access
31:16	Split Transaction Limit (STL)	R/W field available for use by diagnostic software. **NOTE:** Not used by the Intel® 6300ESB ICH for modifying its "commitment" level. The Intel® 6300ESB ICH internal launch algorithms keep buffers from being overallocated.	0000h	R/W
15:00	Split Transaction Capacity (STC)	Infinite capacity due to launch algorithm keeping buffers from overrunning. The Intel® 6300ESB ICH internal launch algorithms keep buffers from being overallocated	FFFFh	RO

18.6.1.31 Offset E0: ACNF – Additional Intel® 6300ESB ICH Configuration

Table 614. Offset E0: ACNF – Additional Intel® 6300ESB ICH Configuration

Device 28	***Function*** 0
Offset E0	***Attribute:*** Read-Only
	Size: 32-bit

Bits	Name	Description	Reset Value	Access
31:16	Reserved	Reserved.	0	RO
15:14	Reserved	Reserved.	0	R/W
13:12	Miscella-neous (MSC)	Reserved but maintained as Read/Write in case a situation arises during the project which may require their use.	0	R/W
11	Reserved	Reserved.	0	R/W
10:07	Reserved	Reserved.	0	R/W
06	Reserved	Reserved.	0	RO
05	Bunit Inbound Pending Queue Bypass (BUPB)	When 1, bunit bypasses it's read pending queue when empty. This bit may only be set to '1' when in PCI mode.	0	R/W
04	Reserved	Reserved.	0	R/W
03:00	Reserved	Reserved.	Fh	R/W



18.6.1.32 Offset E4: PCR - PCI Compensation Register

Table 615. Offset E4: PCR - PCI Compensation Register

Device 28	***Function*** 0
Offset E4	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:10	Reserved	Reserved.	0	R/W
09	SBR Enable (SBRE)	This field specifies the maximum size write a master should request in a single b 0 = PCI-X secondary bus reset (PCIXSBRST#) disabled and SBR bit has no effect. See Section 18.6.1.21, "Offset 3E: BCTRL—Bridge Control", bit 6, for SBR details. 1 = PCI-X secondary bus reset (PCIXSBRST#) enabled and SBR is set. **NOTE:** Processor always writes a '1' into this bit and enables the secondary bus reset for the PCI-X bus.	0	R/W
08:0	Reserved	Reserved.	0	R/W

18.6.1.33 Offset F0: HCCR - Hub Interface Command/Control Register

Table 616. Offset F0: HCCR - Hub Interface Command/Control Register

Device 28	***Function*** 0
Offset F0	***Attribute:*** Read/Write
	Size: 32-bit

Bits	Name	Description	Reset Value	Access
31:20	Reserved	Reserved.	0	RO
19:16	Reserved	Reserved.	0h	RO
15:10	Reserved	Reserved.	0	RO
09:08	Reserved	Reserved.	0	R/W
07:06	Reserved	Reserved.	00	RO
5:4	Reserved	Reserved.	0	RO
03:01	Maximum Data Size (MAXD)	This field specifies the maximum size write a master should request in a single burst, as well as the maximum optimal size read completion a target should return. Encoding: 000: 32 Bytes Others: 64 Bytes	000	R/W
00	Reserved	Reserved.	0	R

18.6.1.34 Offset F8h – Offset FFh: Prefetch Control Registers

The following registers contain prefetch parameters. Each parameter is in 64 byte cache line quantities. BIOS programs the values in these registers upon power up. The values in this register are zero-based – a zero means 64 bytes, and ’1’ means 128 bytes, etc.

There is a fifth parameter in the prefetch algorithm, called “D”, which is the delay to wait before sending a subsequent request of RS when prior requests of RS still have not brought the prefetch buffers above TS. Its value is in PCI clocks, and is [RS]:111. For example, when RS = ‘0101’b, then D is ‘0101111’b.

Note that for Memory Read (MR) and Memory Read Line (MRL) commands in PCI, no prefetching is done. A fetch of one cache line (based upon the cache line size register) is performed and when it drains, the delayed transaction is complete. A new delayed transaction is established when the master wished the burst to continue.

18.6.1.35 Offset F8h: PC33 - Prefetch Control – 33 MHz

Table 617. Offset F8h: PC33 - Prefetch Control – 33 MHz

Device 28	***Function*** 0
Offset F8h	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:12	Subsequent Threshold (TS)	Subsequent threshold size in 64-byte cache lines	1h	R/W
11:08	Subsequent Request (RS)	Subsequent request size in 64-byte cache lines. Allowable programmable values are 00h or 01h only.	1h	R/W
07:04	Initial Threshold (TI)	Initial threshold size in 64-byte cache lines	1h	R/W
03:00	Initial Request (RI)	Initial request size in 64-byte cache lines. Allowable programmable values are 00h or 01h only.	1h	R/W

18.6.1.36 Offset FAh: PC66 - Prefetch Control – 66 MHz

Table 618. Offset FAh: PC66 - Prefetch Control – 66 MHz

Device 28	***Function*** 0
Offset FAh	***Attribute:*** Read/Write
	Size: 16-bit

Bits	Name	Description	Reset Value	Access
15:12	Subsequent Threshold (TS)	Subsequent threshold size in 64-byte cache lines.	3h	R/W
11:08	Subsequent Request (RS)	Subsequent request size in 64-byte cache lines. Allowable programmable values are 00h or 01h only.	1h	R/W
07:04	Initial Threshold (TI)	Initial threshold size in 64-byte cache lines.	2h	R/W
03:00	Initial Request (RI)	Initial request size in 64-byte cache lines. Allowable programmable values are 00h or 01h only.	1h	R/W

18.7 PCI Mode in the PCI-X Interface

This section discusses the specifics of the PCI interface associated with the PCI-X interface while operating in PCI mode. The PCI section of this document describes the "legacy" PCI which is not visible external to the chip. PCI-X is not mentioned in this section. To see specifics on how the PCI-X interface operates, refer to Section 18.8, "PCI-X Interface".

18.7.1 Summary of Changes

For the most part, the PCI interface of the Intel® 6300ESB ICH is exactly the same as the PCI interface for the P64H2.

- Full 64 bit addressing inbound
- Inbound packet size based upon cache line size of the platform.
- I/O space may be programmed to 1K granularity through the EN1K bit of the CNF register.
- When inbound reads are retried, they are moved to the side so posted writes and completion packets may pass. I/O reads and writes on PCI are no longer be forwarded to the Hub Interface.

18.7.2 Transaction Types

Table 619. Intel® 6300ESB I/O Controller Hub PCI Transactions

Type of Transaction		Intel® 6300ESB ICH as		Type of Transaction		Intel® 6300ESB ICH as	
		Master	Target			Master	Target
0000	Interrupt acknowledge	No	No	1000	Reserved†	No	No
0001	Special cycle	Yes	No	1001	Reserved†	No	No
0010	I/O read	Yes	No	1010	Configuration Read	Yes	No
0011	I/O write	Yes	No	1011	Configuration Write	Yes	No
0100	Reserved†	No	No	1100	Memory Read Multiple	No	Yes
0101	Reserved†	No	No	1101	Dual Address Cycle	Yes	Yes
0110	Memory read	Yes	Yes	1110	Memory Read Line	No	Yes
0111	Memory write	Yes	Yes	1111	Memory Write and Invalidate	No	Yes

† The Intel® 6300ESB ICH never initiates a PCI transaction with a reserved command code and ignores reserved command codes as a target.

As a PCI master, the Intel® 6300ESB ICH has access to the 32-bit address space. As a target, the Intel® 6300ESB ICH may accept dual address cycles up to the full 64-bit address space. The Intel® 6300ESB ICH supports the linear increment address mode only for bursting memory transfers (indicated when the low two address bits are equal to '0'). When either of these address bits is nonzero, the Intel® 6300ESB ICH disconnects the transaction after the first data transfer.

The Intel® 6300ESB ICH decodes all PCI cycles in medium DEVSEL# timing.

18.7.3 Detection of 64-Bit Environment

The Intel® 6300ESB ICH drives REQ64# low during PXPCIRST# on each PCI-X interface to signal that the bus is a 64-bit bus.

18.7.4 Data Bus

For supplying data, the Intel® 6300ESB ICH drives the following in the data phase:

- The low 32 bits of data on PXAD[31:0]
- The low four byte enable bits on PXC/BE#[3:0]
- The high 32 bits of data on PXAD[63:32] (64-bit data phases only)
- The high four byte enable bits on PXC/BE#[7:4] (64-bit data phases only)

As a PCI master, when the Intel® 6300ESB ICH drives PXREQ64# and detects PXACK64# asserted in the same clock that it detects PXDEVSEL# asserted, every data phase then consists of 64 bits and eight byte enable bits.

On write transactions, when the Intel® 6300ESB ICH does not detect PXACK64# asserted in the same clock that it detects PXDEVSEL# asserted, it redirects all data to AD[31:0] and byte enables to C/BE#[3:0]. For 64-bit memory-write transactions that end at an odd dWord boundary, the Intel® 6300ESB ICH drives the byte enable bits to '1', and drives random but stable data on PXAD[63:32].

On read transactions, the Intel® 6300ESB ICH drives eight bits of byte enables on PXC/BE#[7:0]. It generates byte enables from the Hub Interface byte enables, with the upper dWord driven on PXC/BE#[7:4]. When ACK64# is not sampled active with PXDEVSEL# active, then the Intel® 6300ESB ICH downshifts the all byte enables PXC/BE#[3:0].

The Intel® 6300ESB ICH does not assert REQ64# when initiating a transfer under the following conditions:

- The Intel® 6300ESB ICH is initiating an I/O transaction.
- The Intel® 6300ESB ICH is initiating a configuration transaction.
- The Intel® 6300ESB ICH is initiating a special cycle transaction.
- A 1-dWord or 2-dWord transaction is being performed.
- When the address of the Hub Interface initiated transaction is not quad word aligned.

As a PCI target, the Intel® 6300ESB ICH does not assert PXACK64# when PXREQ64# was not asserted by the initiator.

18.7.5 Write Transactions

18.7.5.1 Posted

Posted write forwarding is used for memory write and for memory write and invalidate transactions. When the Intel® 6300ESB ICH decodes a memory write transaction for the Hub Interface, it asserts PXDEVSEL# and PXTRDY# in the same clock, provided that enough buffer space is available in the posted data queue. The Intel® 6300ESB ICH adds no target wait states.

The Intel® 6300ESB ICH disconnects a write transaction when:

- The initiator terminates the transaction by de-asserting PXFRAME# and PXIRDY#.
- A 4 Kbyte page boundary is reached.
- The posted write data buffer fills up.

18.7.5.2 Non-Posted

Delayed write forwarding is not used. It is only for I/O write transactions. Since the Intel® 6300ESB ICH does not support I/O write transactions across a bridge, these cycles all result in a master abort.

Note: Configuration cycles are not allowed to cross a bridge per the PCI bridge specification.

18.7.5.3 Fast Back-to-Back

The Intel® 6300ESB ICH allows fast back-to-back write transactions on PCI.

18.7.6 Read Transactions

18.7.6.1 Prefetchable

Any memory read multiple command on PCI that is decoded by the Intel® 6300ESB ICH is prefetched on the Hub Interface. Prefetching may be optionally disabled when bit 4 of the Intel® 6300ESB ICH Configuration Register (offset 40-41h) is set. The Intel® 6300ESB ICH does not prefetch past a 4 Kbyte page boundary.

18.7.6.2 Delayed

All memory read transactions are delayed read transactions. When the Intel® 6300ESB ICH accepts a delayed read request, it samples the address, command, and address parity. This information is entered into the delayed transaction queue and all I/O transactions then master abort.

18.7.7 Transaction Termination

18.7.7.1 Normal Master Termination

As a PCI master, the Intel® 6300ESB ICH uses normal termination when DEVSEL# is returned by the target within five clock cycles of PXFRAME# assertion. It terminates a transaction when the following conditions are met:

- All write data for the transaction is transferred from the Intel® 6300ESB ICH data buffers to the target.
- The master latency timer expires and the Intel® 6300ESB ICH's bus grant is de-asserted.

18.7.7.2 Master Abort Termination

When an Intel® 6300ESB ICH initiated transaction is not responded to with DEVSEL# within five clocks of PXFRAME# assertion, the Intel® 6300ESB ICH terminates the transaction with a master abort. The Intel® 6300ESB ICH sets the received master abort bit in the status register corresponding to the target bus.

Note: When the Intel® 6300ESB ICH performs a Type 1 to special cycle translation, a master abort is the expected termination for the special cycle on the target bus. In this case, the master abort received bit is not set, and the Type 1 con-figuration transaction is disconnected after the first data phase.

18.7.7.3 Target Termination Received by the Intel® 6300ESB ICH

When the Intel® 6300ESB ICH receives a retry or disconnect response from a target, it re-initiates the transfer with the remaining length. When the Intel® 6300ESB ICH receives a target abort, and the cycle requires completion on the Hub Interface, the Intel® 6300ESB ICH returns the target abort code to the Hub Interface as the completion status.

18.7.7.4 Target Termination Initiated by the Intel® 6300ESB ICH

The Intel® 6300ESB ICH returns a target retry to an initiator for memory read transactions when any of the following conditions are met:

- A new transaction for delayed transaction queue.
- The request has already been queued, but has not completed on the Hub Interface.
- The delayed transaction queue is full, and the transaction cannot be queued.
  - — A LOCK transaction has been established from the Hub Interface to PCI.

The Intel® 6300ESB ICH disconnects an initiator when one of the following conditions is met:

- The Intel® 6300ESB ICH cannot accept any more write data
- The Intel® 6300ESB ICH has no more read data to deliver
- When the memory address is non-linear

The Intel® 6300ESB ICH returns a target abort to PCI when the cycle master aborted or target aborted on the Hub Interface.

18.7.8 LOCK Cycles

A lock is established when a memory read from the Hub Interface that targets PCI with the lock bit set, and at least one byte enable active, is responded to with a TRDY# by a PCI target. The Intel® 6300ESB ICH does not support a split-lock request with no byte enables are asserted on the initial locked read request. The bus is unlocked when the Unlock Special Cycle is sent on the Hub Interface.

When the bus is locked, the cycle is retried when a memory cycle originates on PCI that is outside the range of the memory windows. No I/O cycles that are destined across the bridge are accepted, whether the bus is locked or not, and then master abort.

Once the bus is locked, any Hub Interface cycle to PCI is driven with the LOCK# pin, even when that particular cycle is not locked.

18.7.9 Error Handling

The Intel® 6300ESB ICH checks and generates parity on the Hub Interface and parity on the PCI interfaces. Parity errors must always be reported to some system level software, typically the device driver or the OS. This section describes how a standard PCI bridge handles these errors. For enhanced error detection, see the RAS section located section.

The Intel® 6300ESB ICH requires that the “Hub Interface Parity Unsupported” bit (D30:F0:40h:bit 20) is cleared in order to perform any parity checking as described below. Good Hub Interface parity is presented to all logic in the Intel® 6300ESB ICH when the bit is set.

To support error reporting on the PCI bus, the Intel® 6300ESB ICH implements the following:

- PERR# and SERR# signals on PCI
- Primary status (offset 06-07h) and secondary status registers (offset 1E-1Fh)

The Intel® 6300ESB ICH does not have the PERR# or SERR# pins on the Hub Interface. The Intel® 6300ESB ICH is capable of generating NMI, and SMI Address Parity Errors

Address parity errors are very serious and may abort further data transfers, depending upon the direction of the transfer and the setting of the Parity Error Response Enable bit, as described in the following paragraphs. The Intel® 6300ESB ICH checks address parity for all transactions on both the Hub Interface and PCI buses, for all address and all bus commands.

When the Intel® 6300ESB ICH detects an parity error in the header section of a Hub Interface packet, it:

- Sets the Detected Parity Error bit in the Primary status register (bit 15 of offset 06-07h) when the address is targeting the device. The bridge devices log address parity errors independent of the target address.
- Generates NMI/SMI (as enabled) and sets the signaled system error bit in the primary status register (bit 14 of offset 06-07h), when the parity error response bit in the command register (bit 6 of offset 04-05h) is set and SERR# is enabled.
- Attempts to interpret the cycle as best it can, and forwards the cycle with an address parity error tag to the internal logic, where it aborts internally. When a device is not enabled to respond to parity errors, it ignores the address parity error (except for setting the Detected Parity Error bit). When the address targets that device, the device accepts the cycle and responds as though there was no address parity error. The cycle is forwarded to PCI with good address parity when the cycle targets a bridge and it is not enabled to respond to parity errors.

When the Intel® 6300ESB ICH detects an address parity error on the PCI interface, the following events occur:

The Intel® 6300ESB ICH sets the detected parity error bit in the secondary status register (bit 15 of offset 1E-1F).

- When the parity error response bit is ’0’ in the bridge control register (bit ’0’ of offset 3E-3F), the address parity errors are ignored. The cycles would be treated as though no error was observed.
- When the parity error response bit is set and the address parity error is observed on memory cycles, the cycle is accepted as though the address was correct. Delayed Transactions are established for memory reads and data are posted for memory writes. The cycles are forwarded to the Hub Interface with correct address parity.

The Intel® 6300ESB ICH generates NMI/SMI (as enabled) sets the signaled system error bit in the Primary Status Register, when all of the following conditions are met:

- The SERR# enable bit is set in the primary command register.
- The parity error response bit is set in the bridge control register.
- The SERR# enable bit is set in the bridge control register.

The Intel® 6300ESB ICH generates NMI, if the following conditions are met:

- Port70.7 (I/O register at offset 70h, bit 7) is enabled.

- The parity errors response bit is set in the bridge control register (D28:3Eh,0)

18.7.9.1 Data Parity Errors

Unlike address parity errors, data parity errors are not considered as severe and transactions are aborted. The following sections describe the sequence of events when a data parity error is detected for the following transactions:

- Configuration Write Transactions
- Read Transactions (inbound and outbound)
- Posted Write Transaction

18.7.9.1.1 Hub Interface Configuration Write Transactions

When the Intel® 6300ESB ICH detects a data parity error during a Type 0 configuration write transaction to one of the Intel® 6300ESB ICH configuration spaces, the Intel® 6300ESB ICH:

- Does not write the data to the configuration register when parity error response is enabled.
- Sets the Detected Parity Error bit in the Primary status register (bit 15 of offset 06-07h).
- Generates NMI/SMI (depending on which is enabled) and sets the signaled system error bit (bit 14) in the Primary status register, when the Parity Error Response Enable bit in the command register (bit 6 of offset 04-05h) is set.

18.7.9.1.2 Read Transactions from Hub Interface Targeting PCI on the PCI-X

When the Intel® 6300ESB ICH detects a read data parity error on the PCI bus from a Hub Interface initiated read, it:

- Sets the detected parity error bit in the secondary status register (bit 15 of offset 1E-1Fh).
- Sets the Data parity detected bit in the secondary status register (bit 8 of offset 1E-1Fh), when the secondary interface parity error response bit is set in the bridge control register (bit '0' of offset 3E-3Fh).
- Forces bad parity error with the data back to the initiator on the Hub Interface.

18.7.9.1.3 Read Transactions from PCI Targeting Hub Interface

When the Intel® 6300ESB ICH detects a data parity error on a Hub Interface completion packet from a previous memory read request on PCI, the Intel® 6300ESB ICH:

- Sets the detected parity error bit in the primary status register (bit 15 of offset 06-07h).
- Sets the data parity detected bit in the primary status register (bit 8 of offset 06-07h) and generates the NMI/SMI (depending on which is enabled), when the primary interface parity error response bit is set in the command register (bit 6 of offset 04-05h).
- Forwards the bad parity with the data back to PCI.

18.7.9.1.4 Write Transactions on Hub Interface – Intel® 6300ESB ICH as a Hub Interface Target

When the Intel® 6300ESB ICH detects a data parity error on a Hub Interface write request, it:

- Sets the data parity error detected bit in the status register (bit 15 of offset 06-07h) of the target interface (PCI bridge primary).

- Forwards the bad parity with the data to PCI when decoded by the bridge.
- Generates NMI/SMI (depending on which is enabled) and sets the signaled system error bit (bit 14) in the Primary status register, when the parity error response bit (bit 6) is set in the command register.

18.7.9.1.5 Write Transactions on Hub Interface – Intel® 6300ESB ICH as a Hub Interface Master

There is no way of detecting that a northern device detected a parity error from a Hub Interface posted write from PCI. Therefore, no action is taken by the Intel® 6300ESB ICH.

18.7.9.1.6 Write Transactions on PCI – Intel® 6300ESB ICH as PCI Target

When the Intel® 6300ESB ICH detects a data parity error on a PCI write, it:

- Asserts PERR# two cycles after the data transfer, when the secondary interface parity error response bit is set in the bridge control register.
- Sets the secondary interface parity error detected bit in the secondary status register.
- Forces bad parity error condition to the primary bus.

18.7.9.1.7 Write Transactions on PCI – Intel® 6300ESB ICH as PCI Master

When a data parity error is reported on the PCI bus from a Hub Interface or PCI peer initiated write request by the target's assertion of PERR#, the Intel® 6300ESB ICH:

- Sets the Detected Parity Detected bit in the secondary status register (bit 8 of offset 1E-1Fh), when the secondary interface parity error response bit is set in the bridge control register.
- Generates NMI/SMI (depending on which is enabled) and sets the signaled system error bit in the status register, when all of the following conditions are met:
  - The SERR# enable bit is set in the command register.
  - The secondary interface parity error response bit is set in the bridge control register.
  - The primary interface parity error response bit is set in the command register.
  - The Intel® 6300ESB ICH did not detect the parity error on the Hub Interface (i.e., the parity error was not forwarded from the Hub Interface).

18.7.9.2 System Errors

18.7.9.2.1 PCI SERR# Pin Assertion

When SERR# is sampled asserted, the Intel® 6300ESB ICH sets the received system error bit in the secondary status register. The Intel® 6300ESB ICH generates NMI/SMI (depending on which is enabled) when:

- The SERR# forward enable bit is set in the bridge control register, and
- The primary SERR# enable bit is set in the Primary command register.

18.7.9.2.2 Other System Errors

The Intel® 6300ESB ICH also conditionally NMI or SMI as enabled for any of the following reasons:

- Master timeout on delayed transaction when the primary SERR# enable bit is set and SERR# due to timeout enable bit (bit 11 of offset 3E-3Fh) is set.
- The MAM bit (Master Abort Mode) is set in the bridge control register and a posted write from the Hub Interface results in a master abort on PCI, or a posted write from one PCI interface results in a master abort on the other PCI interface. (No

indication is given back on the Hub Interface when a posted PCI write fails on the Hub Interface – the north Hub Interface agent must handle this condition).

18.8 PCI-X Interface

This section is not intended to describe the PCI-X protocol. It is intended to clarify the Intel® 6300ESB ICH behavior in areas of the specification which are open to interpretation. Please see the PCI-X Addendum to the PCI specification, revision 1.0 for all details related to PCI-X operation.

Unless otherwise noted in this section, the Intel® 6300ESB ICH follows all rules of the PCI-X addendum.

18.8.1 Command Encoding

Table 620. PCI-X Interface Command Encoding

Type of Transaction		Intel® 6300ESB ICH As		Type of Transaction		Intel® 6300ESB ICH As	
		Master	Target			Master	Target
0000	Interrupt acknowledge	No	No	1000	Alias to Memory Read Block	No	Yes
0001	Special cycle	No	No	1001	Alias to Memory Write Block	No	Yes
0010	I/O read	Yes	No	1010	Configuration Read	Yes	No
0011	I/O write	Yes	No	1011	Configuration Write	Yes	No
0100	Reserved	No	No	1100	Split Completion	Yes	Yes
0101	Reserved	No	No	1101	Dual Address Cycle	Yes	Yes
0110	Memory Read dWord	Yes	Yes	1110	Memory Read Block	Yes	Yes
0111	Memory Write	Yes	Yes	1111	Memory Write Block	No	Yes

18.8.2 Attributes

The following table describes how the Intel® 6300ESB ICH fills in attribute fields where the
PCI-X specification leaves some implementation leeway.

Table 621. Intel® 6300ESB ICH Implementation of Requester Attribute Fields

Attribute	Function
No Snoop (NS)	As a target, this bit is forwarded with the transaction to allow a north bridge to not snoop the transaction. It goes to bit '1' in the TD Attr field of the Hub Interface packet. It is not generated by the Intel® 6300ESB ICH as a master from a Hub Interface packet. The Intel® 6300ESB ICH takes no action on this bit.
Relaxed Ordering (RO)	This bit allows relaxed ordering of transactions, which the Intel® 6300ESB ICH does not permit. This bit is simply forwarded in the Intel® 6300ESB ICH, and is never generated on PCI-X from a Hub Interface packet.
Tag	Since the Intel® 6300ESB ICH only has one outstanding request on PCI-X at a time, this field is be set to 0.
Byte Counts	From the Hub Interface, this is based upon the length field from the Hub Interface, which is dWord based.

18.8.3 Special Notes for Burst Transactions

The PCI-X specification allows burst transactions to cross page (in the Intel® 6300ESB ICH's case, this is 4K) and 4 Gbyte address boundaries. As a PCI-X master, the Intel® 6300ESB ICH ends the transaction at a 4K boundary. As a PCI-X target, the Intel® 6300ESB ICH allows a burst past a 4K page boundary.

The Intel® 6300ESB ICH does not issue an immediate response as a target for a burst read command, but it must be ready with 128 bytes of data space (an ADQ) as an initiator. When it does not have this space available, it does not issue the transaction.

18.8.4 Device Select Timing

PCI-X targets are required to claim transactions by asserting DEVSEL# as shown in Table 622. The Intel® 6300ESB ICH responds as a type A target.

Table 622. DEVSEL# Timing

Decode Speed	PCI-X
1 clock after address phase(s)	Not Supported
2 clocks after address phase(s)	Decode A
3 clocks after address phase(s)	Decode B
4 clocks after address phase(s)	Decode C
5 clocks after address phase(s)	N/A
6 clocks after address phase(s)	Subtractive

18.8.5 Wait States

The Intel® 6300ESB ICH does not generate wait states as a target. Instead, it ends the transfer.

18.8.6 Split Transactions

18.8.6.1 Completer Attributes

Table 623. Intel® 6300ESB ICH Implementation Completer Attribute Fields

Attribute	Function
Byte Count Modified (BCM)	This bit is used for diagnostic purposes. The Intel® 6300ESB ICH never sets this bit.
Split Completion Error (SCE)	The Intel® 6300ESB ICH is only set this bit when a memory read command from PCI-X master or target aborted on the Hub Interface.
Split Completion Message (SCM)	This bit shadows the SCE bit.

18.8.6.2 Requirements for Accepting Split Completions

The Intel® 6300ESB ICH asserts DEVSEL# and discards the data when the Requester ID matches the bridge, but the Tag does not match that of any outstanding requests from this device, or when the byte count exceeds that of the Split Request.

The Hub Interface accepts more than one completion required request from the Hub Interface, but only one is pending on any PCI/PCI-X interface at a time.

18.8.6.3 Split Completion Messages

The Intel® 6300ESB ICH may only generate error messages for cycles that cross the bridge that master or target abort. No DWORD cycles cross the bridge that require completion (i.e., I/O cycles). Therefore, the Intel® 6300ESB ICH generates a “PCI-X Bridge Error” completion message for the memory read commands as shown in Table 624.

Table 624. Split Completion Messages

Index	Message
00h	**Master-Abort:** The Intel® 6300ESB ICH encountered a Master-Abort on the destination bus.
01h	**Target-Abort:** The Intel® 6300ESB ICH encountered a Target-Abort on the destination bus.

18.8.6.4 Arbitration Among Multiple Split Completions

The Intel® 6300ESB ICH arbitrates among all active split completions so that each completion receives consideration for running on PCI. When there are multiple completions waiting to use PCI, the Intel® 6300ESB ICH internally arbitrates based upon its MLT value, even when no other agents are requesting on the bus. Therefore, the Intel® 6300ESB ICH ends one transaction when its MLT expires, reload, and start another transaction.

When any particular transaction runs out of data, and there are other active transactions to run, the Intel® 6300ESB ICH switches to the next agent, even when the MLT has not expired for that transaction.

Finally, the prefetch algorithm is altered such that several transactions may be active at one time.

18.8.7 Transaction Termination as a PCI-X Target

18.8.7.1 Retry

The Intel® 6300ESB ICH retries a cycle when the Split Request queue is full (i.e., we already have four current and four pending Split Transactions). It has room to accept a split completion as it has a dedicated buffer for split completions. It also retries a cycle when the bus is locked. The Intel® 6300ESB ICH stores no state from the transaction on a retry.

18.8.7.2 Split Response

All cycles that cross the bridge receive this termination, when they are not retried.

18.8.7.3 Master-Abort

Any I/O transaction that would cross from PCI-X to either the Hub Interface or the peer bridge are not decoded and results in a master abort to the PCI-X initiator.

18.8.8 Arbitration

The Intel® 6300ESB ICH parks on the last agent to use PCI. This allows PCI devices operating as a single stream to stay on PCI bus for the duration of their transfer.

18.8.9 Bridge Buffer Requirements

The Intel® 6300ESB ICH has 128 bytes (one ADQ) available for accepting memory write, split completion, and immediate read data. The Intel® 6300ESB ICH contains 1.5K of data total for inbound transactions.

The Intel® 6300ESB ICH PCI-X interface terminates all memory transactions (Memory Read DWORD, Memory Read Block, and Alias to Memory Read Block) that address a device north of the bridge with a Split Response. Other split transaction commands are not decoded by the Intel® 6300ESB ICH.

The Intel® 6300ESB ICH does not implement any split completion buffer allocation algorithm as listed in the PCI-X specification. This is overhead that is not necessary. The Intel® 6300ESB ICH does not request on the Hub Interface more than it has buffer space for on returns, and does not initiate a cycle from the Hub Interface that it cannot accept as a return. The bridge rules of the specification already allow the PCI-X interface to retry split completions when the bridge is temporarily full.

Therefore, the split transaction control registers are not used by the Intel® 6300ESB ICH.

18.8.10 Locked Transactions

The Intel® 6300ESB ICH is not locked until the target has completed at least the first data phase as an Immediate Transaction or a Split Transaction (target signals Split Response).

18.8.11 Error Support

18.8.11.1 General

As a PCI-X Target, the Intel® 6300ESB ICH bridge responds as specified in the PCI-X Addendum.

18.8.11.2 Special Parity Error Rule for Split Response

When the Intel® 6300ESB ICH calculates a data parity error when a target signals Split Response for a read transaction, it records the error as described in section 5.4.1 of the *PCI/X 1.0 Specification*. Furthermore, when the Intel® 6300ESB ICH is enabled to assert PERR# on the secondary bus and enabled to assert SERR# on the Hub Interface, it generates NMI/SMI (depending on which is enabled).

18.9 Transaction Termination Translation between Interfaces

Though Intel® 6300ESB ICH's primary bus is the Hub Interface, from a register and software perspective, the Hub Interface is a PCI-X bus and Intel® 6300ESB ICH is a PCI-X bridge that supports a secondary bus configured as either PCI or PCI-X.

Section 8.7.1.5 of the *PCI-X 1.0 Specification* modified the behavior of a bridge from that specified in the PCI to PCI bridge 1.1 spec regarding returning completions on the primary bus when the secondary bus transaction terminates in either a master abort or target abort. In general, the PCI-X spec does not honor the Master Abort Mode bit for cycles requiring completions, and returns to the primary bus the termination that occurred on the secondary bus without any translation.

The following sections describe the behavior of the Intel® 6300ESB ICH on both the Hub Interface and the PCI/PCI-X under various termination conditions. For specific information as to why the Intel® 6300ESB ICH's PCI, PCI-X, or Hub Interface generates a specific termination, see the specific sections on the interface above.

18.9.1 Behavior of Hub Interface Initiated Cycles to PCI / PCI-X Receiving Immediate Terminations

The behavior described for completion required cycles is independent of the setting of the Master Abort Mode bit, and is independent of whether the cycle is exclusive (locked) or not. The Intel® 6300ESB ICH returns all '1's on data bytes for a read completion that terminates in either Master Abort or Target Abort.

Table 625. Immediate Terminations of Completion Required Cycles to PCI / PCI-X

PCI / PCI-X Termination	Hub Interface Completion	Status Register Bits Set

† The Master Data Parity Error bit is set only when a data parity error is encountered on the PCI/PCI-X bus.

Table 625. Immediate Terminations of Completion Required Cycles to PCI/PCI-X

Successful	Successful	Master Data Parity Error (Sec)†
Master Abort	Master Abort	Received Master Abort (Sec)
Target Abort	Target Abort	Received Target Abort (Sec) Signaled Target Abort (Pri) Master Data Parity Error (Sec)†

† The Master Data Parity Error bit is set only when a data parity error is encountered on the PCI/PCI-X bus.

Table 626. Immediate Terminations of Posted Write Cycles to PCI/PCI-X

PCI/PCI-X Termination	MAM Bit	Action	Status Register Bits Set
Successful	N/A	None	None
Master Abort	1	Generate NMI/SMI	Received Master Abort (Sec) Signaled System Error (Pri)
Master Abort	0	None	Received Master Abort (Sec)
Target Abort	N/A	Generate NMI/SMI as enabled	Received Target Abort (Sec) Signaled System Error (Pri)

18.9.2 Behavior of Hub Interface Initiated Cycles to PCI-X Receiving Split Terminations

The behavior described in the following table is independent of the Master Abort Mode bit and whether or not the cycle is exclusive (locked) or not. The Intel® 6300ESB ICH returns all '1's on all data bytes for a read completion that terminates in either Master Abort or Target Abort on the Hub Interface. Note that when a target or master abort is returned on the Hub Interface, the attached PCI/PCI-X bus is not locked. This is of special importance to the completion messages of "data parity error", "byte count out of range", "write data parity error", "device specific", and reserved/illegal codes. The Intel® 6300ESB ICH must not lock its bus on these errors, even though they are not explicitly master or target aborts on the PCI-X interface.

Table 627. Split Terminations of Completion Required Cycles to PCI-X (Sheet 1 of 2)

PCI-X Split Termination	Message		Hub Interface Completion	Status Register Bits Set
	Class	Index		
Successful	0	00h	Successful	Master Data Parity Error (Sec), when encountered
Master Abort	1	00h	Master Abort	Received Master Abort (Sec)
Target Abort	1	01h	Target Abort	Received Target Abort (Sec) Signaled Target Abort (Pri)
Write Data Parity Error	1	02h	Target Abort	Master Data Parity Error (Sec) Signaled Target Abort (Pri)
Byte Count Out Of Range	2	00h	Target Abort	Signaled Target Abort (Pri)

Table 627. Split Terminations of Completion Required Cycles to PCI-X (Sheet 2 of 2)

PCI-X Split Termination	Message		Hub Interface Completion	Status Register Bits Set
	Class	Index		
Write Data Parity Error	2	01h	Target Abort	Master Data Parity Error (Sec) Signaled Target Abort (Pri)
Device Specific	2	8Xh	Target Abort	Signaled Target Abort (Pri)
Reserved/Illegal	Others		Target Abort	Signaled Target Abort (Pri)

18.9.3 Hub Interface Action on Immediate Responses to PCI-X Split Completions

The following table indicates what the Intel® 6300ESB ICH does when it is returning a split completion to PCI-X from a normal Hub Interface completion, and receives an immediate response indicating some kind of error.

Table 628. Hub Interface Response to PCI-X Split Completion Terminations of Completion Required Cycles

Split Completion Termination	Action	Status Register Bits
Successful	None	None
Master Abort	Assert SERR# †	Received Master Abort (Sec) Split Completion Discarded (Sec) Signaled System Error (Pri)†
Target Abort	Assert SERR# †	Received Target Abort (Sec) Split Completion Discarded (Sec) Signaled System Error (Pri)†

† In this case, the assertion of SERR# and setting of the Signaled System Error bit only occur when the SERR# Enabled in the primary command register is set.

18.9.4 Behavior of PCI/PCI-X Initiated Cycles to Hub Interface

Table 629. Terminations of Completion Required Cycles to Hub Interface

Hub Interface Termination	PCI Completion	Status Register Bits Set
Successful	Successful	None
Master Abort (PCI)	Target Abort[1]	Received Master Abort (Pri) Signaled Target Abort (Sec)
Master Abort (PCI-X)	Split Master Abort[2]	Received Master Abort (Pri)
Target Abort	Target Abort (PCI)[1] Split Target Abort (PCI-X)[2]	Received Target Abort (Pri) Signaled Target Abort (Sec)
Master and Target Abort	Target Abort (PCI)[1] Split Target Abort (PCI-X)[2]	Received Master Abort (Pri) Received Target Abort (Pri) Signaled Target Abort (Sec)

NOTES:
1. The Intel® 6300ESB ICH only signals Target Abort when the error has been logged from the Hub Interface before the initial connect by PCI or when the PCI master reconnects after a previous disconnect. When the Intel® 6300ESB ICH receives an abort on the Hub Interface in the middle of a read completion stream it does not interrupt the stream to signal Target Abort.
2. The Intel® 6300ESB ICH issues a Split Completion Error Message with either Master Abort or Target Abort for the remaining completion sequence when an abort is detected on the Hub Interface. When several bytes of data returned successfully from the Hub Interface and have not yet been sent back on PCI-X, when the abort is detected on the Hub Interface the Intel® 6300ESB ICH stops the current sequence for that data (if it was running) and generates the Split Completion Error Message.

18.10 Delayed/Split Transactions

18.10.1 Number Supported

When in PCI mode, transactions follow the delayed transaction model of PCI 2.2. When in PCI-X mode, transactions follow the split transaction model of PCI-X. For each bridge, the Intel® 6300ESB ICH supports eight delayed / split transactions inbound, and one delayed/split transaction outbound.

The Hub Interface may take four outbound delayed/split transactions, but only launches one at a time to each PCI-X interface. Each PCI interface may take eight delayed/split transactions, but only launches four of those transactions onto the Hub Interface.

The outbound delayed/split transactions does not prefetch from PCI devices, regardless of whether the transaction falls in the prefetchable window or the non-prefetchable window. When in PCI mode, the inbound delayed/split transactions prefetch for all command types. The “memory read” command may optionally have its prefetch turned off as specified in the PCI bridge specification. When in PCI-X mode, the inbound delayed/split transactions do not prefetch – they acquire only the byte count from the request.

18.10.2 Prefetch Algorithm

Since outbound cycles are not prefetched, there is no algorithm. The algorithm for inbound cycles is below. Note that the algorithm changes depending upon whether only one device is requesting or multiple devices are requesting.

18.10.2.1 Parameters

Parameters based upon the Prefetch Parameter Registers at offset F8h – FFh

Ri	Initial request size (bits[03:00])
Ti	Initial threshold (bits[07:04])
Rs	Subsequent request (bits [11:08])
Ts	Subsequent threshold (bits[15:12])
D	Delay to wait between next Ts (calculated). The value is “Rs:111” clocks.

Other Algorithm Parameters

Sb	Buffer size (either 1K or 2K, depending upon the delayed transaction bit (offset 40h, bit 2))
N	Data in buffer + data in flight (requested to SiBUS but not returned)
B	Data in buffer

18.10.2.2 Algorithm (Single Device Only)

1. Establish DT, launch request of size Ri. The actual amount fetched is such that the transfer ends on a naturally aligned 128-byte line. When the initial address is less than 64-bytes into the 128-byte line, the Ri value is rounded down (i.e., eight 64-byte lines become seven 64-byte lines + remainder). When the initial address is more than 64-bytes into the 128-byte line, the Ri value is rounded up (i.e., eight 64-byte lines become nine 64-byte lines + remainder).

   Example 1: Address starts at 32 bytes into a 128-byte line, and the fetch length is 4*64 byte lines (256 bytes). The amount fetched is 256 - 32 = 224 bytes (56 dWords).

   Example 2: Address starts at 96 bytes into a 128-byte line, and the fetch length is 4*64-byte lines (256 bytes). The amount fetched is 256 + (128 - 96) = 288 bytes (72 dWords).

2. Wait until at least some data has returned and master has reconnected. In PCI mode, this is when the first qWord becomes available. In PCI-X, when not running in 133 MHz mode, or running in 133 MHz mode but the request size is less than or equal to 256 bytes, this is when the first ADB becomes available.
3. When N < Ti, launch a request of size Rs (truncated by Sb, when necessary). Start Timer when there are not more active delayed transactions. When there are other active delayed transactions, go to step 5.
4. Check for size B vs. Ts

   When B < Ts, wait for timer to expire before launch of size Rs. Restart timer. Go to Step 4.

   When B > Ts before timer expires, reset timer. Go to Step 5.

5. Wait for B < Ts; launch request of size Rs (truncated by Sb, when necessary). Restart timer. Go to step 4.

18.10.3 Algorithm (Multiple PCI-X Devices Requesting)

When multiple agents are requesting in PCI-X mode, the definition of T_S changes. Instead of just indicating data in buffer, it becomes like T_I, and represents data in buffer plus data in flight. When multiple agents are requesting in PCI-X mode, the Intel® 6300ESB ICH needs to switch between these agents for completions. It does this by utilizing its MLT parameter. When the MLT expires, it stops this stream and switches to another stream.

Differences from P64H algorithm:

- No connect threshold: as soon as the first data is available in the DT buffer, a PCI device is allowed to connect.
- Allows multiple outstanding reads per DT buffer, (so long as restrict size of all outstanding reads for a DT buffer to remaining capacity in DT buffer), yielding smaller prefetch overshoot
- Periodic subsequent fetch: smaller, more frequent requests reduce prefetch overshoot
- T_I and T_S as a low watermark takes into account data in flight; not just data remaining in the DT buffer
- Delayed subsequent launch for multi-stream operation to reduce prefetch overshoot
- First subsequent launch threshold

18.10.4 Accesses From Multiple Agents to Same 4K Page

In order to avoid the need to track the status of the buffers when multiple agents are asking for data from the same 4K page, the Intel® 6300ESB ICH retries a PCI master when the same PCI master has already established another delayed transaction to that 4K page.

18.11 Internal Bus/Device Communication

Internally, all devices that reside on the “logical PCI bus” are connected to an internal bus called “SiBus” (silicon bus). This is a bus architecture developed within PCG that allows for high code reuse and the ability to connect multiple units together in a standard manner. It is split transaction based.

By choosing this micro-architecture, cycles may originate from any agent and be decoded by any other agent. This allows peer-to-peer communication to effectively be free. For this reason, the SM Bus controller is also connected to this bus, allowing PCI configuration cycles that originate either from the Hub Interface or SM Bus to use the same data and control paths to access internal registers.

However, this must be monitored carefully by the micro-architecture. Configuration cycles from SM Bus must be allowed to reach their destination, even when the Hub Interface communication to one of the PCI busses is blocked due to a deadlock condition.

Therefore, the micro-architecture must ensure the following:

- All units that have a configuration space that could be accessed by SM Bus have a “1 command only” depth. This helps ensure that multiple requests are not outstanding, minimizing any possibility of SM Bus accesses being stuck.
- The Intel® 6300ESB ICH does not launch successive requests that have the same hub ID/pipe ID on the internal bus, ensuring that no space has to be reserved to re-order the completion data. Only cycles that have a unique hub ID/pipe ID may be launched simultaneously, and their completions may return in any order.
- Any request from SM Bus (or the Hub Interface) that targets the I/OxAPIC must be able to complete, even when the I/OxAPIC has an interrupt to deliver to the Hub Interface. Otherwise, the completion for the SM Bus/Hub Interface access is blocked behind the I/OxAPIC request to the Hub Interface, and it does not finish. System management software must ensure that the SM Bus does not generate accesses PCI. When this occurs and PCI is blocked, the SM Bus is blocked.

18.12 Data Return Behavior of Hub Interface Initiated Reads

For all Hub Interface initiated memory read cycles targeting PCI/PCI-X, the Intel® 6300ESB ICH ensures a return length of a naturally aligned 128-bytes. When a request is less than 128 bytes and within a single 128-byte line, the Intel® 6300ESB ICH generates one completion. When the request crosses a line, the Intel® 6300ESB ICH returns multiple completions, broken on 128-byte line boundaries, until the request is fulfilled.

The Intel® 6300ESB ICH does not return a dWord completion on a memory read command that is longer than a dWord and is qWord aligned.

The Intel® 6300ESB ICH only generates qWord aligned reads whose length is a multiple number of qWords. The Intel® 6300ESB ICH requires that the completions for these requests be returned as qWords and never dwords. For read streaming to work, the Intel® 6300ESB ICH requires that the driving agent only disconnect read completions on a cache line boundary (64 or 128 bytes).

18.13 Performance Targets

18.13.1 Introduction

This information is organized into three sections. The first section specifies general bus timings. The second specifies single active master throughputs. The third specifies concurrent performance when multiple agents are generating requests from both busses.

18.13.2 Definitions and Assumptions

Bandwidth tests are sustained throughput tests. The system may be run until it reaches steady state and then run longer with the bandwidth measured.

The system under test uses 4x, 8 bit, HL 1.5.

Memory bandwidth in the system under test is sufficient to service the requirements of the PCI-X so that contention for memory and other system resources is not a performance bottleneck.

References to signal timings are stated from the point of view of the bus itself. Internal register timings of the individual devices are not considered.

One Megabyte =10^6 Bytes rather than 2^{20} Bytes.

18.13.3 Active Master Clock Counts

This section specifies clock counts for general bus timings and first word latencies for read requests. Bus timings should be consistent regardless of other system activity. First word latency specifications only apply in situations where contention for system resources does not present a performance bottleneck.

First word latency is measured as the number of clocks from the initial assertion of PXFRAME# (this is clock 0) to the first clock on which valid data is returned in response to the request.

Serial I/O Unit 19

The SIU is similar to currently available Super I/O controllers. It is specifically designed for integration into the Intel® 6300ESB ICH. It is connected externally through the LPC bus and consists of two UARTS, a Serial Interrupt Controller, Port 60/64 Emulation and the LPC interface.

19.1 Features

LPC Interface

- Multiplexed command, address and data bus
- 8-Bit I/O transfers
- 16-Bit address qualification for I/O transactions
- Serial IRQ interface compatible with serialized IRQ support for PCI systems

Note: Each SIU port must use a dedicated interrupt. SIU interrupts cannot be shared with each other or with other devices.

Serial Port

- Two serial ports

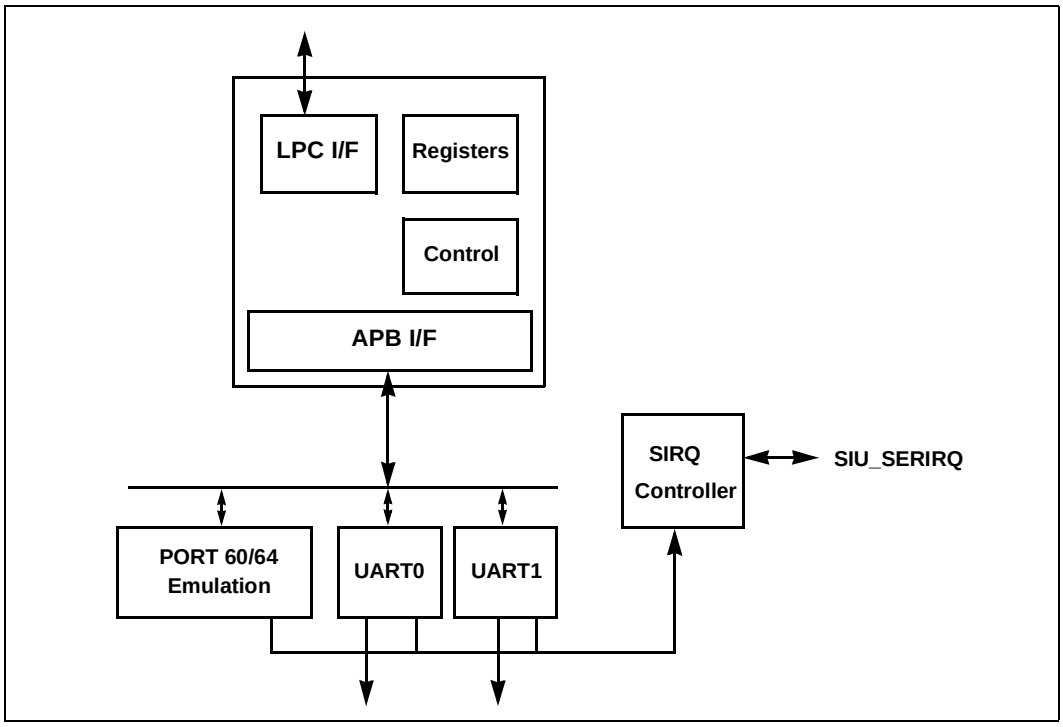
Note: The serial ports of the Intel® 6300ESB ICH are not completely compatible with other 16550 standard devices. A system or software designer must follow the specifications laid out in this document above standard 16550 specifications.

- Configurable I/O addresses and interrupts
- 16-Byte FIFOs
- Supports up to 115 Kbps
- Programmable baud rate generator
- Modem control circuitry

Port 60/64 Emulation

- Configurable unit disable
- Positive decode for I/O cycles to 60h and 64h
- Read/Write scratchpad registers only (sticky bits)
- Configurable interrupt generation on writes to either register

Figure 32. SIU Block Diagram



19.2 Pin Description

19.2.1 Universal Asynchronous Receive And Transmit (UART0, UART1)

Table 630. Universal Asynchronous Receive And Transmit (UART0, UART1) (Sheet 1 of 2)

Signal Name	Type	Description
UART_CLK	I	**Input clock to the SIU.** This clock is passed to the baud clock generation logic of each UART in the SIU.
SIU0_RXD, SIU1_RXD	I	**SERIAL INPUTs for UART0 and UART1:** Serial data input from device pin to the receive port.
SIU0_TXD, SIU1_TXD	O	**SERIAL OUTPUT for UART0 and UART1:** Serial data output to the communication peripheral/modem or data set. Upon reset, the TXD pins will be set to MARKING condition (logic '1' state).

Table 630. Universal Asynchronous Receive And Transmit (UART0, UART1) (Sheet 2 of 2)

Signal Name	Type	Description
SIU0_CTS#, SIU1_CTS#	I	**CLEAR TO SEND:** Active low, this pin indicates that data may be exchanged between the Intel® 6300ESB ICH and external interface. These pins have no effect on the transmitter. **NOTE:** These pins could be used as Modem Status Input whose condition may be tested by the processor by reading bit 4 (CTS) of the Modem Status register (MSR). Bit 4 is the complement of the CTS# signal. Bit 0 (DCTS) of the MSR indicates whether the CTS# input has changed state since the previous reading of the MSR. When the CTS bit of the MSR changes state an interrupt is generated if the Modem Status Interrupt is enabled.
SIU0_DSR#, SIU1_DSR#	I	**DATA SET READY for UART0 and UART1:** Active low, this pin indicates that the external agent is ready to communicate with the Intel® 6300ESB ICH UARTs. These pins have no effect on the transmitter. **NOTE:** These pins could be used as Modem Status Input whose condition may be tested by the processor by reading bit 5 (DSR) of the Modem Status register. Bit 5 is the complement of the DSR# signal. Bit 1 (DDSR) of the Modem status register (MSR) indicates whether the DSR# input has changed state since the previous reading of the MSR. When the DSR bit of the MSR changes state an interrupt is generated if the Modem Status Interrupt is enabled.
SIU0_DCD#, SIU1_DCD#	I	**DATA CARRIER DETECT for UART0 and UART1:** Active low, this pin indicates that data carrier has been detected by the external agent. **NOTE:** These pins are Modem Status Input whose condition may be tested by the processor by reading bit 7 (DCD) of the Modem Status register (MSR). Bit 7 is the complement of the DCD# signal. Bit 3 (DDCD) of the MSR indicates whether the DCD# input has changed state since the previous reading of the MSR. When the DCD bit of the MSR changes from a '1' to 0, an interrupt is generated if the Modem Status Interrupt is enabled.
SIU0_RI# SIU1_RI#	I	**RING INDICATOR for UART0 and UART1:** Active low, this pin indicates that a telephone ringing signal has been received by the external agent. **NOTE:** These pins are Modem Status Input whose condition may be tested by the processor by reading bit 6 (RI) of the Modem Status register (MSR). Bit 6 is the complement of the RI# signal. Bit 2 (TERI) of the MSR indicates whether the RI# input has transition back to an inactive state. When the RI bit of the MSR changes state an interrupt is generated if the Modem Status Interrupt is enabled.
SIU0_DTR#, SIU1_DTR#	O	**DATA TERMINAL READY for UART0 and UART1:** When low these pins informs the modem or data set that the Intel® 6300ESB ICH UART0 and UART1 are ready to establish a communication link. The DTR#x(x=0,1) output signals may be set to an active low by programming the DTRx (x-0,1) (bit0) of the Modem control register to a logic '1'. A Reset operation sets this signal to its inactive state (logic '1'). LOOP mode operation holds this signal in its inactive state.
SIU0_RTS#, SIU1_RTS#	O	**REQUEST TO SEND for UART0 and UART1:** When low these pins informs the modem or data set that Intel® 6300ESB ICH UART0 and UART1 are ready to establish a communication link. The RTS#x(x=0,1) output signals may be set to an active low by programming the RTSx (x-0,1) (bit1) of the Modem control register to a logic '1'. A Reset operation sets this signal to its inactive state (logic '1'). LOOP mode operation holds this signal in its inactive state.

19.3 Functional Description

19.3.1 Host Processor Interface (LPC)

The host processor communicates with the SIU through the LPC bus. Access is through a series of read/write registers and accomplished through I/O cycles or DMA transfers. All registers are 8 bits wide. The SIU registers include global configuration space and device specific regions accessed by setting the Logical Device Number in the SIU Configuration Register 07H (SCR7). See Table 631.

Table 631. Address Map

Address	Block Name	Logical Device
04Eh	Configuration Index	
04Fh	Configuration Data	
Base+(0-7)	Serial Port Com 1	04H
Base+(0-7)	Serial Port Com 2	05H
060h/064h	Port 60/64 Emulation	07h

See Section 19.8, “Configuration” for configuration register descriptions and setting the base address.

19.4 LPC Interface

The LPC interface is used to control all the logical blocks on the SIU. LPC bus signals use PCI 33 MHz electrical signal characteristics. Refer to the *Low Pin Count (LPC) Interface Specification*, Rev 1.0.

19.4.1 LPC Cycles

The following cycle types are supported by the LPC protocol.

Table 632. Supported LPC Cycle Types

Cycle Type	Transfer Size
I/O Write	1 Byte
I/O Read	1 Byte

The SIU ignores cycles that it does not support.

19.4.1.1 I/O Read and Write Cycles

The SIU is the target for I/O cycles. I/O cycles are initiated by the host for register or FIFO accesses and will generally have minimal Sync times.

Data transfers are assumed to be exactly 1-byte. If the processor requested a 16- or 32-bit transfer, the host must break it up into 8-bit transfers.

See the *Low Pin Count (LPC) Interface Specification* for the sequence of cycles for the I/O Read and Write cycles.

19.4.2 Reset Policy

The following rules govern the reset policy:

- The SIU reset (active low) is internally tied to the PCI bus reset.
- When the SIU reset goes active (low):
  - The host drives the LFRAME# signal high, tristates the LAD[3:0] signals, and ignores the LDRQ# signal.
  - The SIU ignores LFRAME#, tristates the LAD[3:0] pins and drives the SIU's LDRQ# signal inactive (high).

Note: LPC bus signals from SIU are tied to primary LPC interface external to the Intel® 6300ESB ICH device. Host LPC and SIU LPC names are used interchangeably throughout.

19.4.3 LPC Transfers

19.4.3.1 I/O Transfers

These will generally be used for register or FIFO accesses, and will generally have minimal Sync times. The minimum number of wait-states between bytes is 1. Data transfers are assumed to be exactly 1 byte. The host is responsible for breaking up larger data transfers into 8 bit cycles.

Table 633. I/O Sync Bits Description

Bits	Indication
0000	Sync Achieved with no error.
0101	Indicates that Sync not Achieved yet, but the part is driving the bus.
0110	Indicates that Sync not Achieved yet, but the part is driving the bus, and expect long Sync.
1010	**Special Case:** Peripheral indicating errors.

19.5 Logical Device 4 and 5: Serial Ports (UARTs)

This section describes the Universal Asynchronous Receiver/Transmitter (UART) serial port used for the two UARTs integrated into the SIU. The UART may be controlled through programmed I/O. The basic programming model is the same for both UARTs with the only difference being the Logical Device Number assigned to each.

19.5.1 Overview

The serial port consists of a UART which supports all the functions of a standard 16550 UART including hardware flow control interface.

The UART performs serial-to-parallel conversion on data characters received from a peripheral device or a modem and parallel-to-serial conversion on data characters received from the processor. The processor may read the complete status of the UART at any time during the functional operation. Available status information includes the type and condition of the transfer operations being performed by the UART and any error conditions (parity, overrun, framing, or break interrupt).

The serial port may operate in either FIFO or non-FIFO mode. In FIFO mode, a 16-byte transmit FIFO holds data from the processor to be transmitted on the serial link and a 16-byte Receive FIFO buffers data from the serial link until read by the processor.

Each UART includes a programmable baud rate generator which is capable of dividing the baud clock input by divisors of 1 to (2^{16} -1) and producing a 16X clock to drive the internal transmitter and receiver logic. Each UART has complete modem control capability and a processor interrupt system. Interrupts may be programmed to the user's requirements, minimizing the computing required to handle the communications link. Each UART may operate in a polled or an interrupt driven environment as configured by software.

The baud rate generator input is a function of the UART_CLK and a configurable pre-divide of 1, 8, or 26. See also SIU Configuration (address 29h) in Table 656. The output of the baud rate generator is 16 times the baud rate.

Table 634. UART Clock Divider Support

Clock Frequency	14.7456 MHz	48.0 MHz
Pre-Divide Value	8	26
Generator Frequency	1.8432 MHz	1.8462 MHz

Note: Some clock chips provide a 14.318x MHz clock output. The Intel® 6300ESB ICH's UART clock must use a 14.7456 MHz frequency; most clock chips do not provide this frequency. An option will be to use the 48.0 MHz clock.

Table 635. Baud Rate Examples

Desired Baud Rate	UART Clock Frequency 14.7456 MHz	UART Clock Frequency 48 MHz	
	Divisor	Divisor	% error
1200	768	2500	0.16
2400	384	1250	0.16
4800	192	625	0.16
7200	128	417	0.16
9600	96	312	0.16
19200	48	156	0.16
38400	24	78	0.16
56000	16	54	3
115200	8	26	0.16

19.5.1.1 UART Feature List

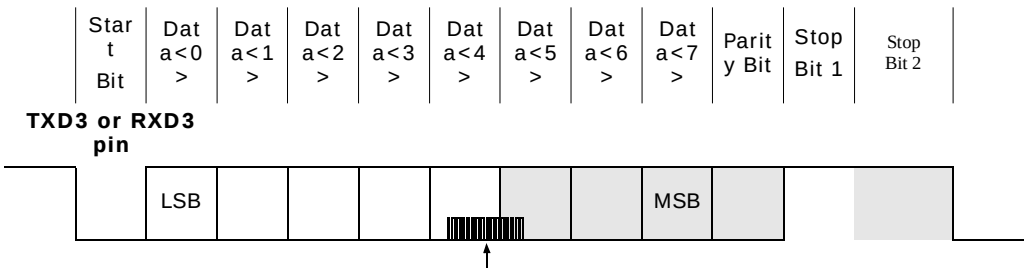
- Adds or deletes standard asynchronous communications bits (start, stop, and parity) to or from the serial data
- Independently controlled transmit, receive, line status and data set interrupts
- Programmable baud rate generator allows division of clock by 1 to (2^{16} -1) and generates an internal 16X clock
- Modem control functions (CTS#, RTS#, DSR#, DTR#, RI#, and DCD#)
- Fully programmable serial-interface characteristics:

- 5, 6, 7 or 8-bit characters
- Even, odd, or no parity detection
- 1, 1-1/2, or 2 stop bit generation
- Baud rate generation (up to 115kbps)
- False start bit detection
- 16-byte Receive FIFO
- Complete status reporting capability
- Line break generation and detection
- Internal diagnostic capabilities include:
- Loopback controls for communications link fault isolation
  - — Break, parity, overrun, and framing error simulation
  - — Fully prioritized interrupt system controls

19.5.1.2 UART Operational Description

The format of a UART data frame is shown in Figure 33.

Figure 33. Example UART Data Frame



Receive data sample counter frequency = 16x bit frequency, each bit is sampled three times in the middle. Shaded bits are optional and may be programmed by users.

Each data frame is between 7 bits and 12 bits long depending on the size of data programmed, if parity is enabled and if two stop bits are selected. The frame begins with a start bit that is represented by a high to low transition. Next, either 5 to 8 bits of data are transmitted, beginning with the least significant bit. An optional parity bit follows, which is set if even parity is enabled and an odd number of '1's exist within the data byte, or if odd parity is enabled and the data byte contains an even number of '1's. The data frame ends with one, one and a half or two stop bits as programmed by the user, which is represented by one or two successive bit periods of a logic '1'.

The unit is disabled upon reset, the user needs to enable the unit by setting bit 6 of Interrupt Enable Register. When the unit is enabled, the receiver starts looking for the start bit of a frame; the transmitter starts transmitting data to the transmit data pin if there is data available in the transmit FIFO. Transmit data may be written to the FIFO before the unit is enabled. When the unit is disabled, the transmitter/receiver finishes the current byte being transmitted/received if it is in the middle of transmitting/ receiving a byte and stops transmitting/receiving more data.

An SIU reset will force the internal register and output signals on the serial port to the values listed in Table 636.

Table 636. SIU Signal Reset States

Register / Signal	Reset Control	Reset State
Interrupt Enable Register	RESET	All bits are low.
Interrupt ID Register	RESET	Bit 0 is forced high. Bits 1-3 and 6-7 are forced low. Bits 4-5 are permanently low.
Line Control Register	RESET	All bits are forced low.
Line Status Register	RESET	Bits 0-4, and 7 are forced low. Bits 5 and 6 are forced high.
Modem Control Register	RESET	Bits 0, 1, 2, 3, 4 are forced low. Bits 5, 6, and 7 are permanently low.
Modem Status Register	RESET/Modem signal, read MSR for bits 3-0.	Low
SIU0_TXD SIU1_TXD	RESET	High
SIU_SERIRQ	RESET/ clear LINE STATUS REG	Low
SIU0_RTS# SIU1_RTS#	RESET	High
SIU0_DTR# SIU1_DTR#	RESET	High

19.5.1.3 Internal Register Descriptions

There are 12 registers in the UART. These registers share eight address locations in the I/O address space. Table 637 shows the registers and their addresses as offsets of a base address. Note that the state of the Divisor Latch Bit (DLAB), which is the MOST significant bit of the Serial Line Control Register, affects the selection of certain of the UART registers. The DLAB bit must be set high by the system software to access the Baud Rate Generator Divisor Latches.

Table 637. Internal Register Descriptions

UART Register Addresses (Base + offset)	DLAB Bit Value	Register Accessed
Base	0	Receive BUFFER (read only)
Base	0	Transmit BUFFER (write only)
Base + 01H	0	Interrupt Enable (R/W)
Base + 02H	X	Interrupt I.D. (read only)
Base + 02H	X	FIFO Control (write only)
Base + 03H	X	Line Control (R/W)
Base + 04H	X	Modem Control (R/W)
Base + 05H	X	Line Status (Read only)
Base + 06H	X	Modem Status (Read only)
Base + 07H	X	Scratch Pad (R/W)
Base	1	Divisor Latch (Lower Byte, R/W)
Base + 01H	1	Divisor Latch (Upper Byte, R/W)

19.5.1.3.1 Receive Buffer Register (RBR)

In non-FIFO mode, this register holds the character received by the UART's Receive Shift Register. If fewer than eight bits are received, the bits are right-justified and the leading bits are zeroed. Reading the register empties the register and resets the Data Ready (DR) bit in the Line Status Register to 0. Other (error) bits in the Line Status Register are not cleared. In FIFO mode, this register latches the value of the data byte at the top of the FIFO.

Table 638. Receive Buffer Register (RBR)

Receive Buffer Register RBR read only		Address: Reset State: Access:	Base (DLAB=0) 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	RB[7:0]	Data byte received, least significant bit first.	

19.5.1.3.2 Transmit Holding Register (THR)

This register holds the next data byte to be transmitted. When the Transmit Shift Register becomes empty, the contents of the Transmit Holding Register are loaded into the shift register and the transmit data request (TDRQ) bit in the Line Status Register is set to '1'.

Table 639. Transmit Holding Register (THR)

Transmit Holding Register THR write only		Address: Reset State: Access:	Base (DLAB=0) 00H 8 bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	TB[7:0]	Data byte transmitted, least significant bit first.	

In FIFO mode, writing to THR puts data to the top of the FIFO. The data at the bottom of the FIFO is loaded to the shift register when it is empty.

19.5.1.3.3 Interrupt Enable Register (IER)

This register enables five types of interrupts which independently activate the int signal and set a value in the Interrupt Identification Register. Each of the five interrupt types may be disabled by resetting the appropriate bit of the IER register. Similarly, by setting the appropriate bits, selected interrupts may be enabled. Receiver time out interrupt may be configured to be separated from the receive data available interrupt (using the bit5: COMP) to avoid interrupt controller and DMA controller serving the receive FIFO at the same time.

Note: The use of bit 4 and 5 is different from the register definition of standard 16550.

Table 640. Interrupt Enable Register (IER)

Interrupt Enable Register IER read/write		Address: Reset State: Access:	Base + 01H (DLAB=0) 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:6	RSVD	RSVD = 0	
5	COMP	**Compatibility Enable:** 0 = Bit 0 of this register also controls RTOIE and bit 4 is RSVD. 1 = Bit 4 of this register controls RTOIE. **NOTE:** The use of bit 5 is different from the register definition of the 16550. The 16550 has this bit always set to 0.	
4	RTOIE	**Receiver Time Out Interrupt Enable:** 0 = Receiver data Time out interrupt disabled. 1 = Receiver data Time out interrupt enabled. **NOTE:** The use of bit 4 is different from the register definition of the 16550. The 16550 has this bit always set to 0.	
3	MIE	**Modem Interrupt Enable:** 0 = Modem Status interrupt disabled. 1 = Modem Status interrupt enabled.	
2	RLSE	**Receiver Line Status Interrupt Enable:** 0 = Receiver Line Status interrupt disabled. 1 = Receiver Line Status interrupt enabled.	
1	TIE	**Transmit Data request Interrupt Enable:** 0 = Transmit FIFO Data Request interrupt disabled. 1 = Transmit FIFO Data Request interrupt enabled.	
0	RAVIE	**Receiver Data Available Interrupt Enable:** When BIT 5 = 1 0 = Receiver Data Available (Trigger level reached) interrupt disabled. 1 = Receiver Data Available (Trigger level reached) interrupt enabled. When BIT 5 = '0' the following additional functionality is used. 0 = Receiver data Time Out Interrupt also disabled. 1 = Receiver data Time Out Interrupt enabled.	

19.5.1.3.4 Interrupt Identification Register (IIR)

In order to minimize software overhead during data character transfers, the UART prioritizes interrupts into four levels (listed in Table 641) and records these in the Interrupt Identification Register. The Interrupt Identification Register (IIR) stores information indicating that a prioritized interrupt is pending and the source of that interrupt.

Table 641. Interrupt Conditions

Priority Level	Interrupt Origin
1 (highest)	**Receiver Line Status:** One or more error bits were set.
2	Received Data is available. In FIFO mode, trigger level was reached; in non-FIFO mode, RBR has data.
2	Receiver Time out occurred. It happens in FIFO mode only, when there is data in the receive FIFO but no activity for a time period.
3	Transmitter requests data. In FIFO mode, the transmit FIFO is half or more than half empty; in non-FIFO mode, THR is read already.
4	**Modem Status:** One or more of the modem input signals has changed state.

Table 642. Interrupt Identification Register (IIR)

Interrupt Identification Register IIR read-only		Address: Reset State: Access:	Base + 02H 01H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:6	FIFOES[1:0]	**FIFO Mode Enable Status:** 00 = Non-FIFO mode is selected. 01 = Reserved 10 = Reserved 11 = FIFO mode is selected (TRFIFOE = 1).	
5:4	--	Reserved	
3	TOD (IID3)	**Time Out Detected:** 0 = No time out interrupt is pending. 1 = Time out interrupt is pending. (FIFO mode only)	
2:1	IID[2:1]	**Interrupt Source Encoded:** 00 = Modem Status (CTS, DSR, RI, DCD modem signals changed state) 01 = Transmit FIFO requests data. 10 = Received Data Available 11 = Receive error (Overrun, parity, framing, break, FIFO error)	
0	IP#	**Interrupt Pending:** 0 = Interrupt is pending. (Active low) 1 = No interrupt is pending.	

Table 643. Interrupt Identification Register Decode

Interrupt ID bits				Interrupt SET/RESET Function			
3	**2**	**1**	**0**	**Priority**	**Type**	**Source**	**RESET Control**
0	0	0	1	-	None	No Interrupt is pending.	-
0	1	1	0	Highest	Receiver Line Status	Overrun Error, Parity Error, Framing Error, Break Interrupt.	Reading the Line Status Register.
0	1	0	0	Second Highest	Received Data Available.	**Non-FIFO mode:** Receive Buffer is full.	**Non-FIFO mode:** Reading the Receiver Buffer Register.
						FIFO mode: Trigger level was reached.	**FIFO mode:** Reading bytes until Receiver FIFO drops below trigger level or setting RESETRF bit in FCR register.
1	1	0	0	Second Highest	Character Timeout indication.	**FIFO Mode only:** At least one character is in receiver FIFO and there was no activity for a time period.	Reading the Receiver FIFO or setting RESETRF bit in FCR register.
0	0	1	0	Third Highest	Transmit FIFO Data Request	**Non-FIFO mode:** Transmit Holding Register Empty	Reading the IIR Register (if the source of the interrupt) or writing into the Transmit Holding Register.
						FIFO mode: Transmit FIFO has half or less than half data.	Reading the IIR Register (if the source of the interrupt) or writing to the Transmitter FIFO.
0	0	0	0	Fourth Highest	Modem Status	Clear to Send, Data Set Ready, Ring Indicator, Received Line Signal Detect	Reading the modem status register

19.5.1.3.5 FIFO Control Register (FCR)

FCR is a write only register that is located at the same address as the IIR (IIR is a read only register). FCR enables/disables the transmitter/receiver FIFOs, clears the transmitter/receiver FIFOs, and sets the receiver FIFO trigger level.

Note: The use of bit 6 and 7 is different from the register definition of standard 16550.

Table 644. FIFO Control Register (FCR) (Sheet 1 of 2)

FIFO Control Register FCR write-only		Address: Reset State: Access:	Base + 02H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:6	ITL[1:0]	**Interrupt Trigger Level:** When the number of bytes in the receiver FIFO equals the interrupt trigger level programmed into this field and the Received Data Available Interrupt is enabled (through IER), an interrupt is generated and appropriate bits are set in the IIR. 00 = 1 byte or more in FIFO causes interrupt (same as 16550). 01 = RSVD 10 = 8 bytes or more in FIFO causes interrupt and DMA request (same as 16550). 11 = RSVD	

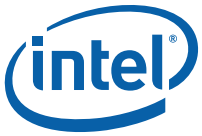


Table 644. FIFO Control Register (FCR) (Sheet 2 of 2)

FIFO Control Register FCR write-only		Address: Reset State: Access:	Base + 02H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
5:3	–	Reserved	
2	RESETTF	**Reset Transmitter FIFO:** When RESETTF is set to 1, the transmitter FIFO counter logic is set to 0, effectively clearing all the bytes in the FIFO. The TDRQ bit in LSR are set and IIR shows a transmitter requests data interrupt if the TIE bit in the IER register is set. The transmitter shift register is not cleared; it completes the current transmission. After the FIFO is cleared, RESETTF is automatically reset to 0. 0 = Writing '0' has no effect. 1 = The transmitter FIFO is cleared (FIFO counter set to 0). After clearing, bit is automatically reset to 0.	
1	RESETRF	**Reset Receiver FIFO:** When RESETRF is set to 1, the receiver FIFO counter is reset to 0, effectively clearing all the bytes in the FIFO. The DR bit in LSR is reset to 0. All the error bits in the FIFO and the FIFOE bit in LSR are cleared. Any error bits, OE, PE, FE or BI, that had been set in LSR are still set. The receiver shift register is not cleared. If IIR had been set to Received Data Available, it is cleared. After the FIFO is cleared, RESETRF is automatically reset to 0. 0 = Writing '0' has no effect. 1 = The receiver FIFO is cleared (FIFO counter set to 0). After clearing, bit is automatically reset to 0.	
0	TRFIFOE	**Transmit and Receive FIFO Enable:** TRFIFOE enables/disables the transmitter and receiver FIFOs. When TRFIFOE = 1, both FIFOs are enabled (FIFO Mode). When TRFIFOE = 0, the FIFOs are both disabled (non-FIFO Mode). Writing a '0' to this bit clears all bytes in both FIFOs. When changing from FIFO mode to non-FIFO mode and vice versa, data is automatically cleared from the FIFOs. This bit must be 1 when other bits in this register are written or the other bits are not programmed. 0 = FIFOs are disabled. 1 = FIFOs are enabled.	

19.5.1.3.6 Line Control Register (LCR)

In the Line Control Register (LCR), the system programmer specifies the format of the asynchronous data communications exchange. The serial data format consists of a start bit (logic 0), five to eight data bits, an optional parity bit, and one or two stop bits (logic 1). The LCR has bits for accessing the Divisor Latch and causing a break condition. The programmer may also read the contents of the Line Control Register. The read capability simplifies system programming and eliminates the need for separate storage in system memory.

Table 645. Line Control Register (LCR) (Sheet 1 of 2)

Serial Line Control Register LCR read/write		Address: Reset State: Access:	Base + 03H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7	DLAB	**Divisor Register Access Bit:** This bit is the Divisor Latch Access Bit. It must be set high (logic 1) to access the Divisor Latches of the Baud Rate Generator during a READ or WRITE operation. It must be set low (logic 0) to access the Receiver Buffer, the Transmit Holding Register, or the Interrupt Enable Register. 0 = Access Transmit Holding register (THR), Receive Buffer Register (RBR) and Interrupt Enable Register. 1 = Access Divisor Latch Registers (DLL and DLM)	
6	SB	**Set Break:** This bit is the set break control bit. It causes a break condition to be transmitted to the receiving UART. When SB is set to a logic 1, the serial output (TXD) is forced to the spacing (logic 0) state and remains there until SB is set to a logic 0. This bit acts only on the TXD pin and has no effect on the transmitter logic. This feature enables the processor to alert a terminal in a computer communications system. If the following sequence is executed, no erroneous characters will be transmitted because of the break: • Load 00H in the Transmit Holding register in response to a TDRQ interrupt • After TDRQ goes high (indicating that 00H is being shifted out), set the break bit before the parity or stop bits reach the TXD pin • Wait for the transmitter to be idle (TEMT = 1) and clear the break bit when normal transmission has to be restored During the break, the transmitter may be used as a character timer to accurately establish the break duration. In FIFO mode, wait for the transmitter to be idle (TEMT=1) to set and clear the break bit. 0 = No effect on TXD output. 1 = Forces TXD output to '0' (space).	
5	STKYP	**Sticky Parity:** This bit is the "sticky parity" bit, which may be used in multiprocessor communications. When PEN and STKYP are logic 1, the bit that is transmitted in the parity bit location (the bit just before the stop bit) is the complement of the EPS bit. If EPS is 0, the bit at the parity bit location will be transmitted as a 1. In the receiver, if STKYP and PEN are 1, the receiver compares the bit that is received in the parity bit location with the complement of the EPS bit. If the values being compared are not equal, the receiver sets the Parity Error bit in LSR and causes an error interrupt if line status interrupts were enabled. For example, if EPS is 0, the receiver expects the bit received at the parity bit location to be 1. If it is not, then the parity error bit is set. By forcing the bit value at the parity bit location, rather than calculating a parity value, a system with a master transmitter and multiple receivers may identify some transmitted characters as receiver addresses and the rest of the characters as data. If PEN = 0, STKYP is ignored. 0 = No effect on parity bit. 1 = Forces parity bit to be opposite of EPS bit value.	

Table 645. Line Control Register (LCR) (Sheet 2 of 2)

Serial Line Control Register LCR read/write		Address: Reset State: Access:	Base + 03H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
4	EPS	**Even Parity Select:** This bit is the even parity select bit. When PEN is a logic '1' and EPS is a logic '0', an odd number of logic '1's is transmitted or checked in the data word bits and the parity bit. When PEN is a logic '1' and EPS is a logic '1', an even number of logic ones is transmitted or checked in the data word bits and parity bit. If PEN = 0, EPS is ignored. 0 = Sends or checks for odd parity. 1 = Sends or checks for even parity.	
3	PEN	**Parity Enable:** This is the parity enable bit. When PEN is a logic '1', a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. (The parity bit is used to produce an even or odd number of '1's when the data word bits and the parity bit are summed.) 0 = No parity function 1 = Allows parity generation and checking.	
2	STB	**Stop Bits:** This bit specifies the number of stop bits transmitted and received in each serial character. If STB is a logic '0', one stop bit is generated in the transmitted data. If STB is a logic '1' when a 5-bit word length is selected through bits '0' and '1', then one and one half stop bits are generated. If STB is a logic '1' when either a 6, 7, or 8-bit word is selected, then two stop bits are generated. The receiver checks the first stop bit only, regardless of the number of stop bits selected. 0 = 1 stop bit 1 = 2 stop bits, except for 5-bit character then 1-1/2 bits	
1:0	WLS[1:0]	**Word Length Select:** The Word Length Select bits specify the number of data bits in each transmitted or received serial character. 00 = 5-bit character (default) 01 = 6-bit character 10 = 7-bit character 11 = 8-bit character	

19.5.1.3.7 Line Status Register (LSR)

This register provides status information to the processor concerning the data transfers. Bits 5 and 6 show information about the transmitter section. The rest of the bits contain information about the receiver.

In non-FIFO mode, three of the LSR register bits, parity error, framing error, and break interrupt, show the error status of the character that has just been received. In FIFO mode, these three bits of status are stored with each received character in the FIFO. LSR shows the status bits of the character at the top of the FIFO. When the character at the top of the FIFO has errors, the LSR error bits are set and are not cleared until software reads LSR, even if the character in the FIFO is read and a new character is now at the top of the FIFO.

Bits 1 through 4 are the error conditions that produce a receiver line status interrupt when any of the corresponding conditions are detected and the interrupt is enabled. These bits are not cleared by reading the erroneous byte from the FIFO or receive buffer. They are cleared only by reading LSR. In FIFO mode, the line status interrupt

occurs only when the erroneous byte is at the top of the FIFO. If the erroneous byte being received is not at the top of the FIFO, an interrupt is generated only after the previous bytes are read and the erroneous byte is moved to the top of the FIFO.

Table 646. Line Status Register (LSR) (Sheet 1 of 2)

Line Status Register LSR read-only		Address: Reset State: Access:	Base + 05H 60H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7	FIFOE	**FIFO Error Status:** In non-FIFO mode, this bit is 0. In FIFO Mode, FIFOE is set to '1' when there is at least one parity error, framing error, or break indication for any of the characters in the FIFO. Note that a processor read to the Line Status register does not reset this bit. FIFOE is reset when all error bytes have been read from the FIFO. FIFOE set to '1' does not generate interrupt. 0 = No FIFO or no errors in receiver FIFO. 1 = At least one character in receiver FIFO has errors.	
6	TEMT	**Transmitter Empty:** TEMT is set to a logic '1' when the Transmit Holding register and the Transmitter Shift register are both empty. It is reset to a logic '0' when either the Transmit Holding register or the transmitter shift register contains a data character. In FIFO mode, TEMT is set to '1' when the transmitter FIFO and the Transmit Shift register are both empty.	
5	TRDQ	**Transmit Data Request:** TDRQ indicates that the UART is ready to accept a new character for transmission. In addition, this bit causes the UART to issue an interrupt to the processor when the transmit data request interrupt enable is set high and generates the DMA request to DMA controller to ask for data. The TDRQ bit is set to a logic '1' when a character is transferred from the Transmit Holding register into the Transmit Shift register. The bit is reset to logic '0' concurrently with the loading of the Transmit Holding register by the processor. In FIFO mode, TDRQ is set to '1' when the transmit FIFO is empty or the RESETTF bit in FCR has been set to 1. It is cleared when at least one byte is written to the transmit FIFO. If more than 16 characters are loaded into the FIFO, the excess characters are lost. 0 = Processor has loaded the Transmit Holding Register. 1 = Transmit FIFO is empty (FIFO mode) or a character has transferred from the Transmit Holding register into the Transmit Shift register.	
4	BI	**Break Interrupt:** BI is set to a logic '1' when the received data input is held in the spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + parity bit + stop bits). The Break indicator is reset when the processor reads the Line Status Register. In FIFO mode, only one character (equal to 00H), is loaded into the FIFO regardless of the length of the break condition. BI shows the break condition for the character at the top of the FIFO, not the most recently received character. 0 = No break signal has been received. 1 = Break signal occurred	

Table 646. Line Status Register (LSR) (Sheet 2 of 2)

Line Status Register LSR read-only		Address: Reset State: Access:	Base + 05H 60H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
3	FE	**Framing Error:** FE indicates that the received character did not have a valid stop bit. FE is set to a logic '1' when the bit following the last data bit or parity bit is detected as a logic '0' bit (spacing level). If the Line Control register had been set for two stop bit mode, the receiver does not check for a valid second stop bit. The FE indicator is reset when the processor reads the Line Status Register. The UART will resynchronize after a framing error. To do this it assumes that the framing error was due to the next start bit, so it samples this "start" bit twice and then takes in the "data". In FIFO mode, FE shows a framing error for the character at the top of the FIFO, not for the most recently received character. 0 = No Framing error 1 = Invalid stop bit has been detected.	
2	PE	**Parity Error:** PE indicates that the received data character does not have the correct even or odd parity, as selected by the even parity select bit. The PE is set to a logic '1' upon detection of a parity error and is reset to a logic '0' when the processor reads the Line Status register. In FIFO mode, PE shows a parity error for the character at the top of the FIFO, not the most recently received character. 0 = No Parity error 1 = Parity error has occurred.	
1	OE	**Overrun Error:** In non-FIFO mode, OE indicates that data in the receiver buffer register was not read by the processor before the next character was transferred into the receiver buffer register, thereby destroying the previous character. In FIFO mode, OE indicates that all 16 bytes of the FIFO are full and the most recently received byte has been discarded. The OE indicator is set to a logic '1' upon detection of an overrun condition and reset when the processor reads the Line Status register. 0 = No data has been lost 1 = Received data has been lost.	
0	DR	**Data Ready:** Bit 0 is set to a logic '1' when a complete incoming character has been received and transferred into the receiver buffer register or the FIFO. In non-FIFO mode, DR is reset to '0' when the receive buffer is read. In FIFO mode, DR is reset to a logic '0' if the FIFO is empty (last character has been read from RBR) or the RESETRF bit is set in FCR. 0 = No data has been received. 1 = Data is available in RBR or the FIFO.	

19.5.1.3.8 Modem Control Register (MCR)

This 8-bit register controls the interface with the modem or data set (or a peripheral device emulating a modem). The contents of the Modem Control register are described in Table 647.

Table 647. Modem Control Register (MCR) (Sheet 1 of 2)

Modem Control Register MCR read/write		Address: Reset State: Access:	Base + 04H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:5	0	Reserved	
4	LOOP	**Loop Back Test Mode:** This bit provides a local Loopback feature for diagnostic testing of the UART. When LOOP is set to a logic 1, the following will occur: The transmitter serial output is set to a logic '1' state. The OUT2# signal is forced to a logic '1' state. The receiver serial input is disconnected from the pin. The output of the Transmitter Shift register is "looped back" into the receiver shift register input. The four modem control inputs (CTS#, DSR#, DCD#, and RI#) are disconnected from the pins and the modem control output pins (RTS# and DTR#) are forced to their inactive state. • Coming out of the loopback test mode may result in unpredictable activation of the delta bits (bits 3:0) in the Modem Status Register (MSR). It is recommended that MSR be read once to clear the delta bits in the MSR. The lower four bits of the Modem Control register are connected to the upper four Modem Status register bits: • DTR = '1' forces DSR to a '1' • RTS = '1' forces CTS to a '1' • OUT1 = '1' forces RI to a '1' • OUT2 = '1' forces DCD to a '1' In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit and receive data paths of the UART. The transmit, receive and modem control interrupts are operational, except the modem control interrupts are activated by Control register bits, not the modem control inputs. A break signal may also be transferred from the transmitter section to the receiver section in loopback mode. 0 = Normal UART operation 1 = Test mode UART operation	
3	OUT2	**Out2# Signal Control:** This bit controls the OUT2# output. When the OUT2 bit is set, OUT2# is asserted low. When the OUT2 bit is cleared, OUT2# is deasserted (set high). Outside of the UART module, the OUT2# signal is used to connect the UART's interrupt output to the Interrupt Controller unit. 0 = OUT2# signal is '1' , which disables the UART interrupt. 1 = OUT2# signal is '0'.	
2	OUT1	**Test Bit:** This bit is used only in Loopback test mode. See (LOOP) Above.	
1	RTS	**Request To Send:** This bit controls the Request to Send (RTS#) output pin. Bit '1' affects the RTS# output in a manner identical to that described below for the DTR bit. 0 = RTS# pin is 1 1 = RTS# pin is 0	

Table 647. Modem Control Register (MCR) (Sheet 2 of 2)

0	DTR	**Data Terminal Ready:** This bit controls the Data Terminal Ready output. When bit 0 is set to a logic '1', the DTR# output is force to a logic '0'. When bit 0 is reset to a logic '0', the DTR# output pin is forced to a logic '1'. • The DTR# output of the UART may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding modem or data set. 0 = DTR# pin is 1 1 = DTR# pin is 0

19.5.1.3.9 Modem Status Register (MSR)

This 8 bit register provides the current state of the control lines from the modem or data set (or a peripheral device emulating a modem) to the processor. In addition to this current state information, four bits of the Modem Status register provide change information. These bits, 3:0, are set to a logic '1' when a control input from the Modem changes state. They are reset to a logic '0' when the processor writes '1's to the bits of the Modem Status register.

When bits 0, 1, 2, or 3 are set to logic 1, a Modem Status interrupt is generated if bit 3 of the Interrupt Enable Register is set.

Table 648. Modem Status Register (MSR) (Sheet 1 of 2)

Modem Status Register **MSR** **read only**		**Address:** **Reset State:** **Access:**	**Base + 06H** **00H** **8-bit**
Bit Number	**Bit Mnemonic**	**Function**	
7	DCD	**Data Carrier Detect:** This bit is the complement of the Data Carrier Detect (DCD#) input. This bit is equivalent to bit OUT2 of the Modem Control register if LOOP in the MCR is set to 1. 0 = DCD# pin is 1 1 = DCD# pin is 0	
6	RI	**Ring Indicator:** This bit is the complement of the ring Indicator (RI#) input. This bit is equivalent to bit OUT1 of the Modem Control register if LOOP in the MCR is set to 1. 0 = RI# pin is 1 1 = RI# pin is 0	
5	DSR	**Data Set Ready:** This bit is the complement of the Data Set Ready (DSR#) input. This bit is equivalent to bit DTR of the Modem Control register if LOOP in the MCR is set to 1. 0 = DSR# pin is 1 1 = DSR# pin is 0	
4	CTS	**Clear to Send:** This bit is the complement of the Clear to Send (CTS#) input. This bit is equivalent to bit RTS of the Modem Control register if LOOP in the MCR is set to 1. 0 = CTS# pin is 1 1 = CTS# pin is 0	
3	DDCD	**Delta Data Carrier Detect:** 0 = No change in DCD# pin since last read of MSR. 1 = DCD# pin has changed state.	

Table 648. Modem Status Register (MSR) (Sheet 2 of 2)

2	TERI	**Trailing Edge Ring Indicator:** 0 = RI# pin has not changed from '0' to '1' since last read of MSR. 1 = RI# pin has changed from '0' to 1.
1	DDSR	**Delta Data Set Ready:** 0 = No change in DSR# pin since last read of MSR. 1 = DSR# pin has changed state.
0	DCTS	**Delta Clear To Send:** 0 = No change in CTS# pin since last read of MSR. 1 = CTS# pin has changed state.

19.5.1.3.10 Scratchpad Register (SCR)

This 8-bit read/write register has no effect on the UART. It is intended as a scratchpad register for use by the programmer.

Table 649. Scratch Pad Register (SCR)

Scratch Pad Register SCR read/write		Address: Reset State: Access:	Base + 07H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	SP[7:0]	No effect on UART functionality	

19.5.1.3.11 Programmable Baud Rate Generator

The UART contains a programmable Baud Rate Generator that is capable of taking the UART_CLK input and dividing it by any divisor from 1 to (2^{16} -1). The output frequency of the Baud Rate Generator is 16 times the baud rate. Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor Latches must be loaded during initialization to ensure proper operation of the Baud Rate Generator. If both Divisor Latches are loaded with 0, the 16X output clock is stopped. Upon loading either of the Divisor latches, a 16-bit baud counter is immediately loaded. This prevents long counts on initial load. Access to the Divisor latch may be done with a word write.

The baud rate of the data shifted in/out of the UART is given by:

BaudRate = UART_CLK(MHz)/[16X Divisor]

For example, if UART_CLK is 14.7456MHz and the divisor is 96, the baud rate is 9600.

A Divisor value of 0 in the Divisor Latch Register is not allowed. The reset value of the divisor
is 02.

Table 650. Divisor Latch Register Low (DLL)

Divisor Latch Register Low DLL read/write		Address: Reset State: Access:	Base (DLAB=1) 02H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	BR[7:0]	Low byte compare value to generate baud rate	

Table 651. Divisor Latch Register High (DLH)

Divisor Latch Register High DLH read/write		Address: Reset State: Access:	Base + 1 (DLAB=1) 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	BR[15:8]	High byte compare value to generate baud rate	

19.5.1.4 FIFO Operation

19.5.1.4.1 FIFO Interrupt Mode Operation

Receiver Interrupt

When the Receive FIFO and receiver interrupts are enabled (FCR[0]=1 and IER[0]=1), receiver interrupts occur as follows:

- The receive data available interrupt is invoked when the FIFO has reached its programmed trigger level. The interrupt is cleared when the FIFO drops below the programmed trigger level.
- The IIR receive data available indication also occurs when the FIFO trigger level is reached, and like the interrupt, the bits are cleared when the FIFO drops below the trigger level.
- The receiver line status interrupt (IIR = C6H), as before, has the highest priority. The receiver data available interrupt (IIR=C4H) is lower. The line status interrupt occurs only when the character at the top of the FIFO has errors.
- The data ready bit (DR in LSR register) is set to '1' as soon as a character is transferred from the shift register to the Receive FIFO. This bit is reset to '0' when the FIFO is empty.

Character Timeout Interrupt

When the receiver FIFO and receiver time out interrupt are enabled, a character timeout interrupt occurs when all of the following conditions exist:

- At least one character is in the FIFO.
- The last received character was longer than four continuous character times ago (if 2 stop bits are programmed the second one is included in this time delay).
- The most recent processor read of the FIFO was longer than four continuous character times ago.
- The receiver FIFO trigger level is greater than one.

The maximum time between a received character and a timeout interrupt is 160 ms at 300 baud with a 12-bit receive character (i.e., 1 start, 8 data, 1 parity, and 2 stop bits).

When a time out interrupt occurs, it is cleared and the timer is reset when the processor reads one character from the receiver FIFO. If a timeout interrupt has not occurred, the timeout timer is reset after a new character is received or after the processor reads the receiver FIFO.

Transmit Interrupt

When the transmitter FIFO and transmitter interrupt are enabled (FCR[0]=1, IER[1]=1), transmit interrupts occur as follows:

- The Transmit Data Request interrupt occurs when the transmit FIFO is half empty or more than half empty. The interrupt is cleared as soon as the Transmit Holding

Register is written (1 to 16 characters may be written to the transmit FIFO while servicing the interrupt) or the IIR is read.

19.5.1.4.2 FIFO Polled Mode Operation

With the FIFOs enabled (TRFIFOE bit of FCR set to 1), setting IER[3:0] to all zeros puts the serial port in the FIFO polled mode of operation. Since the receiver and the transmitter are controlled separately, either one or both may be in the polled mode of operation. In this mode, software checks receiver and transmitter status through the LSR. As stated in the register description:

- LSR[0] is set as long as there is one byte in the receiver FIFO.
- LSR[1] through LSR[4] specify which error(s) has occurred for the character at the top of the FIFO. Character error status is handled the same way as interrupt mode. The IIR is not affected since IER[2] = 0.
- LSR[5] indicates when the transmitter FIFO needs data.
- LSR[6] indicates that both the transmitter FIFO and shift register are empty.
- LSR[7] indicates whether there are any errors in the receiver FIFO.

19.6 Logical Device 7 (07H): Port 60/64 Emulation

This section describes the Port 60/64 Emulation integrated into the SIU.

19.6.1 Feature List

- Configurable unit disable
- Positive decode for I/O cycles to 60h and 64h
- Read/Write Scratchpad Registers Only (sticky bits)
- Interrupt on write and self-interrupt clearing

19.6.2 Overview

The Port 60/64 Emulation Unit consists of two 8-bit I/O registers intended to preserve values written to Port 60 and 64 thus emulating a legacy 8042 device formerly at this legacy I/O address space. These registers may be enabled by BIOS typically in a pre-OS environment and may be disabled during run time. These registers may be used for 8042 keyboard controller emulation but in no way support any controller or functionality beyond a scratchpad register and interrupt generation on writes.

When enabled, this Device will positively decode 8-bit I/O accesses to address 60h and 64h.

Writes to these addresses may generate an interrupt as configured in the Logical Device 07 Primary Interrupt Register (70h). The interrupt generated from this unit will drive active (drives a logical 0) for one SIRQ frame. It does not require any further action (i.e., no EOI required or status bit to clear).

19.6.2.1 Port 60H Emulation (SCR60)

This 8-bit read/write register has no effect. It is intended as a scratchpad register for use by the programmer.

Table 652. Scratch Pad Register P60 (SCR60)

Scratch Pad Register P60 SCR60 read/write		Address: Reset State: Access:	60H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	SP60[7:0]	No effect on SIU functionality	

19.6.2.2 Port 64H Emulation (SCR64)

This 8-bit read/write register has no effect. It is intended as a scratchpad register for use by the programmer.

Table 653. Scratch Pad Register P64 (SCR64)

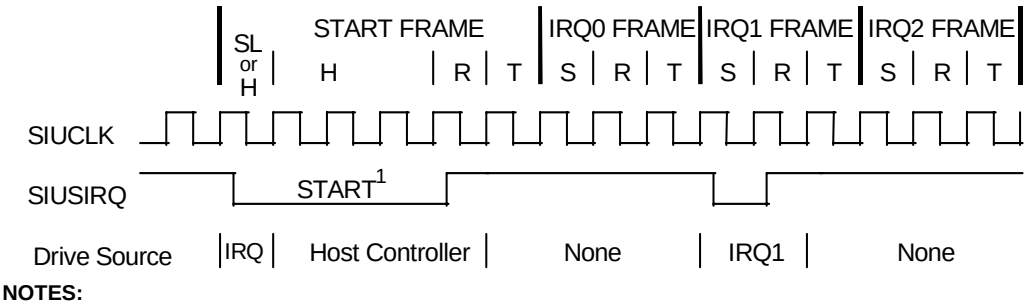
Scratch Pad Register P64 SCR64 read/write		Address: Reset State: Access:	64H 00H 8-bit
Bit Number	**Bit Mnemonic**	**Function**	
7:0	SP64[7:0]	No effect on SIU functionality	

19.7 SERIAL IRQ

The SIU supports the serial interrupt to transmit interrupt information to the host system. The serial interrupt scheme adheres to the Serial IRQ Specification.

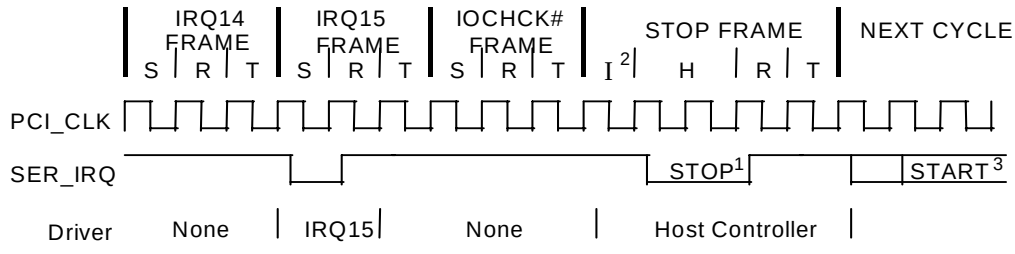
19.7.1 Timing Diagrams For SIU_SERIRQ Cycle

Figure 34. Start Frame Timing with Source Sampled a Low Pulse on IRQ1



NOTES:
1. H=Host Control; R=Recovery; T=Turn-Around; SL=Slave Control; S=Sample
2. Start Frame pulse may be 4-8 clocks wide depending on the location of the device in the PCI bridge hierarchy in a synchronous bridge design.

Figure 35. Stop Frame Timing with Host Using 17 SIU_SERIRQ Sampling Period



NOTES:
1. H=Host Control; R=Recovery; T=Turn-Around; S=Sample; I=Idle
2. Stop pulse is 2 clocks wide for Quiet mode, 3 clocks wide for Continuous mode.
3. There may be none, one or more Idle states during the Stop Frame.
4. The next SIU_SERIRQ cycle's Start Frame pulse may or may not start immediately after the turn-around clock of the Stop Frame.

19.7.1.1 SIU_SERIRQ Cycle Control

There are two modes of operation for the SIU_SERIRQ Start Frame.

1. **Quiet (Active) Mode**: Any device may initiate a Start Frame by driving the SIU_SERIRQ low for one clock, while the SIU_SERIRQ is Idle. After driving low for one clock the SIU_SERIRQ is immediately tri-stated without at any time driving high. A Start Frame may not be initiated while the SIU_SERIRQ is Active. The SIU_SERIRQ is Idle between Stop and Start Frames. The SIU_SERIRQ is Active between Start and Stop Frames. This mode of operation allows the SIU_SERIRQ to be Idle when there are no IRQ/Data transitions which should be most of the time.

   Once a Start Frame has been initiated the Host Controller will take over driving the SIU_SERIRQ low in the next clock and will continue driving the SIU_SERIRQ low for a programmable period of three to seven clocks. This makes a total low pulse width of four to eight clocks. Finally, the Host Controller will drive the SIU_SERIRQ back high for one clock, then tri-state.

   Any SIU_SERIRQ Device (i.e., the SIU) which detects any transition on an IRQ/Data line for which it is responsible must initiate a Start Frame in order to update the Host Controller unless the SIU_SERIRQ is already in an SIU_SERIRQ Cycle and the IRQ/Data transition may be delivered in that SIU_SERIRQ Cycle.

2. **Continuous (Idle) Mode**: Only the Host controller may initiate a Start Frame to update IRQ/Data line information. All other SIU_SERIRQ agents become passive and may not initiate a Start Frame. SIU_SERIRQ will be driven low for four to eight clocks by Host Controller. This mode has two functions. It may be used to stop or idle the SIU_SERIRQ or the Host Controller may operate SIU_SERIRQ in a continuous mode by initiating a Start Frame at the end of every Stop Frame.

   An SIU_SERIRQ mode transition may only occur during the Stop Frame.

Note: Upon reset, SIU_SERIRQ bus is defaulted to Continuous mode, therefore only the Host controller may initiate the first Start Frame. Slaves must continuously sample the Stop Frames pulse width to determine the next SIU_SERIRQ Cycle's mode.

Each SIU port must use a dedicated interrupt. SIU interrupts cannot be shared with each other or with other devices.

19.7.1.2 SIU_SERIRQ Data Frame

Once a Start Frame has been initiated, the SIU will watch for the rising edge of the Start Pulse and start counting IRQ/Data Frames from there. Each IRQ/Data Frame is three clocks: Sample phase, Recovery phase, and Turn-around phase. During the Sample phase the SIU drives the SIU_SERIRQ low, if and only if, its last detected IRQ/Data value was low. If its detected IRQ/Data value is high, SIU_SERIRQ is left tri-stated. During the Recovery phase the SIU drives the SIU_SERIRQ high, if and only if, it had driven the SIU_SERIRQ low during the previous Sample Phase. During the Turn-around Phase the SIU tri-states the SIU_SERIRQ. The SIU will drive the SIU_SERIRQ line low at the appropriate sample point if its associated IRQ/Data line is low, regardless of which device initiated the Start Frame.

The Sample Phase for each IRQ/Data follows the low to high transition of the Start Frame pulse by a number of clocks equal to the IRQ/Data Frame times three, minus one. (e.g., The IRQ5 Sample clock is the sixth IRQ/Data Frame, (6 x 3) - 1 = 17th clock after the rising edge of the Start Pulse).

Table 654. SIU_SERIRQ Sampling Periods

SIU_SERIRQ PERIOD	SIGNAL SAMPLED	# OF CLOCKS PAST START
1	Not Used	2
2	IRQ1	5
3	IRQ2	8
4	IRQ3	11
5	IRQ4	14
6	IRQ5	17
7	IRQ6	20
8	IRQ7	23
9	IRQ8	26
10	IRQ9	29
11	IRQ10	32
12	IRQ11	35
13	IRQ12	38
14	IRQ13	41
15	IRQ14	44
16	IRQ15	47

SIU_SERIRQ Period 13 is used to transfer IRQ12.

19.7.1.3 Stop Cycle Control

Once all IRQ/Data Frames have completed the Host Controller will terminate SIU_SERIRQ activity by initiating a Stop Frame. Only the Host Controller may initiate the Stop Frame. A Stop Frame is indicated when the SIU_SERIRQ is low for two or three clocks. If the Stop Frame's low time is two clocks, the next SIU_SERIRQ Cycle's sampled mode is the Quiet mode; and any SIU_SERIRQ device may initiate a Start Frame in the second clock or more after the rising edge of the Stop Frame's pulse. If the Stop Frame's low time is three clocks, the next SIU_SERIRQ Cycle's sampled mode is the Continuous mode; and only the Host Controller may initiate a Start Frame in the second clock or more after the rising edge of the Stop Frame's pulse.

19.7.1.4 Latency

Latency for IRQ/Data updates over the SIU_SERIRQ bus in bridge-less systems with the minimum Host supported IRQ/Data Frames of seventeen, will range up to 96 clocks (2.88 µs with a 33MHz PCI Bus). If one or more PCI to PCI Bridge is added to a system, the latency for IRQ/Data updates from the secondary or tertiary buses will be a few clocks longer for synchronous buses, and approximately double for asynchronous buses.

19.7.1.5 EOI / ISR Read Latency

Any serialized IRQ scheme has a potential implementation issue related to IRQ latency. IRQ latency could cause an EOI or ISR Read to precede an IRQ transition that it should have followed. This could cause a system fault. The host interrupt controller is responsible for ensuring that these latency issues are mitigated. The recommended solution is to delay EOIs and ISR Reads to the interrupt controller by the same amount as the SIU_SERIRQ Cycle latency in order to ensure that these events do not occur out of order.

19.7.1.6 Reset and Initialization

The SIU_SERIRQ bus uses SIU_LRESET# as its reset signal. The SIU_SERIRQ pin is tri-stated by all agents while SIU_LRESET# is active. With reset, SIU_SERIRQ Slaves are put into the (continuous) IDLE mode. The Host Controller is responsible for starting the initial SIU_SERIRQ Cycle to collect system's IRQ/Data default values. The system then follows with the Continuous/Quiet mode protocol (Stop Frame pulse width) for subsequent SIU_SERIRQ Cycles. It is Host Controller's responsibility to provide the default values to the Interrupt controller and other system logic before the first SIU_SERIRQ Cycle is performed. For SIU_SERIRQ system suspend, insertion, or removal application, the Host controller should be programmed into Continuous (IDLE) mode first. This is to ensure that the SIU_SERIRQ bus is in IDLE state before the system configuration changes.

19.8 Configuration

The Configuration of the SIU is very flexible and is based on the configuration architecture implemented in typical Plug-and-Play components. The SIU is designed for motherboard applications in which the resources required by their components are known. With its flexible resource allocation architecture, the SIU allows the BIOS to assign resources at POST.

19.8.1 Configuration Port Address Selection

The SIU configuration port addresses for INDEX and DATA are fixed at 4Eh/4Fh.

See also Section 8.1.31, "Offset E6h - E7h: LPC_EN—LPC I/F Enables (LPC I/F—D31:F0)" on page 337.

19.8.2 Primary Configuration Address Decoder

After a PCI Reset (SIU_LRESET# pin asserted) or Power On Reset the SIU is in the Run Mode with the two UARTs disabled. They may be configured through two standard Configuration I/O Ports (INDEX and DATA) by placing the SIU into Configuration Mode.

The BIOS uses these configuration ports to initialize the logical devices at POST. The INDEX and DATA ports are only valid when the SIU is in Configuration Mode.

The INDEX and DATA ports are effective only when the chip is in the Configuration State. When the SIU is not in the Configuration State, reads return FFh and write data is ignored.

19.8.2.1 Entering the Configuration State

The device enters the Configuration State by the following contiguous sequence:

Write 80H to Configuration Port.
Write 86H to Configuration Port.

19.8.2.2 Exiting the Configuration State

The device exits the Configuration State by the following contiguous sequence:

Write 68H to Configuration Port.
Write 08H to Configuration Port.

19.8.2.3 Configuration Sequence

To program the configuration registers, the following sequence must be followed:

1. Enter Configuration Mode.
2. Configure the Configuration Registers.
3. Exit Configuration Mode.

19.8.2.4 Configuration Mode

The system sets the logical device information and activates desired logical devices through the INDEX and DATA ports. In configuration mode, the INDEX PORT is located at the CONFIG PORT address and the DATA PORT is at INDEX PORT address + 1.

The desired configuration registers are accessed in two steps:

1. Write the index of the Logical Device Number Configuration Register (i.e., 07) to the INDEX PORT and then write the number of the desired logical device to the DATA PORT.
2. Write the address of the desired configuration register within the logical device to the INDEX PORT and then write or read the configuration register through the DATA PORT.

Note: If accessing the Global Configuration Registers, step (a) is not required.

3. The chip returns to the RUN State.

Note: Only two states are defined: Run and Configuration. In the Run State, the chip will always be ready to enter the Configuration State.

19.8.3 SIU Configuration Registers Summary

Table 655. Configuration Registers Summary

Global Configuration Registers			
Index	**Type**	**Default**	**Configuration Register**
07h	R/W	00h	Logical Device Number
20h	R	00h	Device ID
21h	R	00h	Device Rev
28h	R/W	01h	SIU I/F (wait states)
29h	R/W	02h	SIRQ Configuration
2Eh	R/W	00	Test Mode Configuration Register
Logical Device 4 Registers (Serial Port 0)			
30h	R/W	00h	Enable
60h	R/W	00h	Base I/O Address MSB
61h	R/W	00h	Base I/O Address LSB
70h	R/W	00h	Primary Interrupt Select
74h	R	04h	Reserved
75h	R	04h	Reserved
F0h	R	00h	Vendor Specific Configuration
Logical Device 5 Registers (Serial Port 1)			
30h	R/W	00h	Enable
60h	R/W	00h	Base I/O Address MSB
61h	R/W	00h	Base I/O Address LSB
70h	R/W	00h	Primary Interrupt Select
74h	R	04h	Reserved
75h	R	04h	Reserved
F0h	R	00h	Vendor Specific Configuration
Logical Device 7 Registers (Port Emulation)			
30h	R/W	00h	Enable
60h	R	00h	Base I/O Address MSB
61h	R	60h	Base I/O Address LSB
70h	R/W	00h	Primary Interrupt Select

19.8.3.1 Global Control/Configuration Registers [00h — 2Fh]

The chip-level (global) registers lie in the address range [00h-2Fh]. The design MUST use all 8 bits of the ADDRESS Port for register selection. All unimplemented registers and bits ignore writes and return '0' when read.

The INDEX PORT is used to select a configuration register in the chip. The DATA PORT is then used to access the selected register. These registers are accessible only in the Configuration Mode.

Table 656. Global Control Registers

Register	Address (Type)	Description
Logical Device # Default = 00h	07h (R/W)	**Logical Device Select:** A write to this register selects the current logical device. This allows access to the control and configuration registers for each logical device.
Device ID Default = 00h	20h (R)	**Device ID:** A read only register which provides the Device ID.
Device Rev Default = 01h	21h (R)	**Device Rev:** A read only register which provides device revision information.
SIU Interface Default = 01h	28h (R/W bits 7:2, 0 R- bit 1)	**Bit 1 – LPC bus wait states** 1 = Long wait states (sync 6) 0 = Not supported **Bit 7:2, 0 – RSVD = 0**
SIU Configuration Default = 02h	29h (R/W bits 3:2, 0 R- bit 1)	**Bit 0 – SIRQ enable** 1 = Enabled; participates in interrupt generation 0 = Disabled; serial interrupts disabled **Bit 1 – IRQ mode (Read only, Writes ignored)** 1 = Continuous mode 0 = Quiet mode **Bit 3:2 – UART_CLK pre-divide UART_CLK input** 00 Divide by 1 01 Divide by 8 10 Divide by 26 11 Reserved **Bit 7:4 – RSVD = 0**

19.8.3.2 Logical Device Configuration Registers [30h — FFh]

Used to access the registers that are assigned to each logical unit. This chip supports two logical units and has two sets of logical device registers. The two logical devices are UART0 and UART1. A separate set (bank) of control and configuration registers exists for each logical device and is selected with the Logical Device # Register.

The INDEX PORT is used to select a specific logical device register. These registers are then accessed through the DATA PORT.

The Logical Device registers are accessible only when the device is in the Configuration State. The logical register addresses are shown in Table 657 through Table 659.

Table 657. Logical Device 4 (Serial Port 0)

Logical Device Register	Address	Description

Enable
Default = 00h
	30h
	(R/W)
		Bits[7:1] Reserved, set to '0'.
		Bit[0]
		1 = Enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive

I/O Base Address Default = 00h	60-61h (R/W)	**Registers 60h (MSB) and 61h (LSB)** set the base address for the device. **NOTE:** Decode is on 8 Byte boundaries **Intel® 6300ESB ICH Comm Decode Ranges** 3F8 - 3FF (COM 1) 2F8 - 2FF (COM 2) 220 - 227 228 - 22F 238 - 23F 2E8 - 2EF (COM 4) 338 - 33F 3E8 - 3EF (COM 3)
Primary Interrupt Select Default = 00h	70h (R/W)	**Bits[3:0]** select which interrupt level is used for the primary Interrupt. 00= No interrupt selected 01= IRQ1 02= IRQ2 03= IRQ3 04= IRQ4 05= IRQ5 06= IRQ6 07= IRQ7 08= IRQ8 09= IRQ9 0A= IRQ10 0B= IRQ11 0C= IRQ12 0D= IRQ13 0E= IRQ14 0F= IRQ15 **Bits[7:4]** Reserved **NOTE:** An Interrupt is activated by setting this register to a non-zero value and setting any combination of bits 0-3 in the corresponding UART IER and the OUT2 bit in the MCR **NOTE:** Each SIU port must use a dedicated interrupt. SIU interrupts cannot be shared with each other or with other devices.

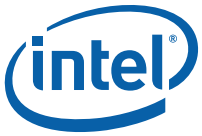


Table 657. Logical Device 4 (Serial Port 0)

Logical Device Register	Address	Description
RSVD Default = 04h	74h (R/W - bit 3:0) (R - bit 7:4)	**Bit 7:0** - RSVD
RSVD Default = 04h	75h (R/W - bit 3:0) (R - bit 7:4)	**Bit 7:0** - RSVD
RSVD Default = 00h	F0h (R/W - bit 0) (R - bit 7:1)	**Bit 7:0** - RSVD

Table 658. Logical Device 5 (Serial Port 1)

Logical Device Register	Address	Description

Enable
Default = 00h
	30h
	(R/W)
		Bits[7:1] Reserved, set to '0'.
		Bit[0]
		1 = Enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive

I/O Base Address Default = 00h	60-61h (R/W)	**Registers 60h (MSB) and 61h (LSB)** set the base address for the device. **NOTE:** Decode is on 8 Byte boundaries. **Intel® 6300ESB ICH Comm Decode Ranges** 3F8 - 3FF (COM 1) 2F8 - 2FF (COM 2) 220 – 227 228 - 22F 238 - 23F 2E8 - 2EF (COM 4) 338 - 33F 3E8 - 3EF (COM 3)
Primary Interrupt Select Default = 00h	70h (R/W)	**Bits[3:0]** select which interrupt level is used for the primary Interrupt. 00 = no interrupt selected 01 = IRQ1 02 = IRQ2 03 = IRQ3 04 = IRQ4 05 = IRQ5 06 = IRQ6 07 = IRQ7 08 = IRQ8 09 = IRQ9 0A = IRQ10 0B = IRQ11 0C = IRQ12 0D = IRQ13 0E = IRQ14 0F = IRQ15 **Bits[7:4]** Reserved **NOTE:** An Interrupt is activated by setting this register to a non-zero value and setting any combination of bits 0-3 in the corresponding UART IER and the OUT2 bit in the MCR **NOTE:** Each SIU port must use a dedicated interrupt. SIU interrupts cannot be shared with each other or with other devices.

Table 658. Logical Device 5 (Serial Port 1)

Logical Device Register	Address	Description
RSVD Default = 04h	74h (R/W - bit 3:0) (R - bit 7:4	**Bit 7:0** - RSVD
RSVD Default = 04h	75h (R/W - bit 3:0) (R - bit 7:4	**Bit 7:0** - RSVD
RSVD Default = 00h	F0h (R/W - bit 0) (R - bit 7:1)	**Bit 7:0** - RSVD

Table 659. Logical Device 7 (Port Emulation)

Logical Device Register	Address	Description

Table 659. Logical Device 7 (Port Emulation)

Enable
Default = 00h
	30h
	(R/W)
		Bits[7:1] Reserved, set to '0'.
		Bit[0]
		1 = Enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive.

I/O Base Address Default = 60h	60-61h (R)	**Registers 60h (MSB) and 61h (LSB)** set the base address for the device. Decode is on 8 Byte boundaries so both 60h and 64h are captured by the single value of 60h in this space. **NOTE:** This device must ignore accesses to unsupported bytes (specifically 61-63h and 65-67h)
Primary Interrupt Select Default = 00h	70h (R/W)	**Bits[3:0]** select which interrupt level is used for the primary Interrupt for Port 60h, *Software Note:* Do not set the interrupt to the same value as the port 64h interrupt. **Bits[7:4]** select which interrupt level is used for the primary Interrupt for Port 64h, *Software Note:* Do not set the interrupt to the same value as the port 60h interrupt. 00 = No interrupt selected 01 = IRQ1 02 = IRQ2 03 = IRQ3 04 = IRQ4 05 = IRQ5 06 = IRQ6 07 = IRQ7 08 = IRQ8 09 = IRQ9 0A = IRQ10 0B = IRQ11 0C = IRQ12 0D = IRQ13 0E = IRQ14 0F = IRQ15 **NOTE:** An Interrupt is activated by enabling this device (offset 30h),setting this register to a non-zero value, and writing to the appropriate I/O address (60h or 64h).

Serial ATA Controller Registers (D31:F2) 20

20.1 PCI Configuration Registers (SATA–D31:F2)

Note: Registers that are not shown should be treated as reserved.

All of the SATA registers are in the core well. They can never be locked.

Table 660. PCI Configuration Map (SATA–D31:F2) (Sheet 1 of 2)

Offset	Mnemonic	Register Name/Function	Default	Type
00-01h	VID	Vendor ID	8086h	RO
02-03h	DID	Device ID	25A3h or 25B0h	RO
04-05h	CMD	Command Register	00h	R/W
06-07h	STS	Device Status	02B0h	R/W
08h	RID	Revision ID	See Note 2	RO
09h	PI	Programming Interface	8Ah	R/W
0Ah	SCC	Sub Class Code	01h or 04h	RO
0Bh	BCC	Base Class Code	01h	RO
0Dh	MLT	Master Latency Timer	00h	RO
0Eh	HTYPE	Header Type	00h	RO
10-13h	PCMD_BAR	Primary Command Block Base Address	00000001h	R/W
14-17h	PCNL_BAR	Primary Control Block Base Address	00000001h	R/W
18-1Bh	SCMD_BAR	Secondary Command Block Base Address	00000001h	R/W
1C-1Fh	SCNL_BAR	Secondary Control Block Base Address	00000001h	R/W
20-23h	BAR	Base Address Register	00000001h	R/W
2C-2Dh	SVID	Subsystem Vendor ID	00h	R/Write-Once
2E-2Fh	SID	Subsystem ID	00h	R/Write-Once
34h	CAP	Capabilities Pointer	80h	RO
3C	INTR_LN	Interrupt Line	00h	R/W
3D	INTR_PN	Interrupt Pin	01h	R/W
40-41h	IDE_TIMP	Primary IDE Timing	0000h	R/W
42-43h	IDE_TIMS	Secondary IDE Timing	0000h	R/W
44h	SIDETIM	Slave IDE Timing	00h	R/W
48h	SDMA_CNT	Synchronous DMA Control Register	00h	R/W

NOTES:
1. The Intel® 6300ESB ICH SATA Controller is not arbitrated as a PCI device, therefore it does not need a master latency timer.
2. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

Table 660. PCI Configuration Map (SATA–D31:F2) (Sheet 2 of 2)

Offset	Mnemonic	Register Name/Function	Default	Type
4A-4Bh	SDMA_TIM	Synchronous DMA Timing Register	0000h	R/W
54-57h	IDE_CONFIG	IDE I/O Configuration Register	00h	R/W
70-71h	PID	PCI Power Management Capability ID	0001h	RO
72-73h	PC	PCI Power Management Capabilities	0002h	RO
74-75h	PMCS	PCI Power Management Control and Status	0000h	R/W
80-81h	MID	Message Signaled Interrupt Capability ID	7005h	RO
82-83h	MC	Message Signaled Interrupt Message Control	0000h	R/W
84-87h	MA	Message Signaled Interrupt Message Address	0000h	R/W
88-89h	MD	Message Signaled Interrupt Message Data	0000h	R/W
90h	MAP	Address Map	00h	R/W
92-93h	PCS	Port Status and Control	0000h	R/W
A0h	SRI	SATA Registers Index	00h	R/W
A4h	SRD	SATA Registers Data	XXh	R/W
E0h–E3h	BFCS	BIST FIS Control/Status	00000000h	R/W, R/WC
E4h–E7h	BFTD1	BIST FIS Transmit Data, DW1	00000000h	R/W
E8h–EBh	BFTD2	BIST FIS Transmit Data, DW2	00000000h	R/W

NOTES:
1. The Intel® 6300ESB ICH SATA Controller is not arbitrated as a PCI device, therefore it does not need a master latency timer.
2. Refer to the Intel® 6300ESB I/O Controller Hub *Specification Update* for the most up-to-date value of the Revision ID register.

20.1.1 Offset 00 - 01h: VID—Vendor ID Register (SATA—D31:F2)

Table 661. Offset 00 - 01h: VID—Vendor ID Register (SATA—D31:F2)

Device:	31	***Function:***	2
Offset:	00-01h	***Attribute:***	Read-Only
Default Value:	8086h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Vendor ID Value	This is a 16-bit value assigned to Intel. Intel VID = 8086h	RO

20.1.2 Offset 02 - 03h: DID—Device ID Register (SATA—D31:F2)

Table 662. Offset 02 - 03h: DID—Device ID Register (SATA—D31:F2)

Device:	31	**Function:**	2
Offset:	02-03h	**Attribute:**	Read-Only
Default Value:	25A3h or 25B0h	**Size:**	16-bit
Lockable:	No	**Power Well:**	Core

Bits	Name	Description	Access
15:0	Device ID Value	Indicates what device number was assigned by the PCI SIG. When Device 31 Function 2, Offset AC h, bit 22=0; DID = 25A3h (hard drive) When Device 31 Function 2, Offset AC h, bit 22=1; DID = 25B0h (RAID)	

20.1.3 Offset 04h - 05h: CMD—Command Register (SATA–D31:F2)

Table 663. Offset 04h - 05h: CMD—Command Register (SATA–D31:F2) (Sheet 1 of 2)

Device:	31	**Function:**	2
Offset:	04h-05h	**Attribute:**	Read-Only, Read/Write
Default Value:	00h	**Size:**	16-bit

Bits	Name	Description	Access
15:11	Reserved	Reserved.	
10	Interrupt Disable	0 = Enables the SATA host controller to assert INTA# (native mode), IRQ14/15 (legacy mode), and MSI (when MSI is enabled). 1 = The interrupt will be deasserted and it may not generate MSIs.	R/W
9	Fast Back-to-Back Enable (FBE)	Reserved as ‘0’.	RO
8	SERR# Enable	Reserved as ‘0’.	RO
7	Wait Cycle Control	Reserved as ‘0’.	RO
6	Parity Error Response	0 = Disabled. SATA Controller will not generate PERR# when a data parity error is detected. 1 = Enabled. SATA Controller will generate PERR# when a data parity error is detected.	R/W
5	VGA Palette Snoop	Reserved as ‘0’.	RO
4	Postable Memory Write Enable (PMWE)	Reserved as ‘0’.	RO

Table 663. Offset 04h - 05h: CMD—Command Register (SATA–D31:F2) (Sheet 2 of 2)

Device:	31	**Function:**	2
Offset:	04h-05h	**Attribute:**	Read-Only, Read/Write
Default Value:	00h	**Size:**	16-bit

Bits	Name	Description	Access
3	Special Cycle Enable (SCE)	Reserved as '0'.	RO
2	Bus Master Enable (BME)	Controls the Intel® 6300ESB ICH's ability to act as a PCI master for IDE Bus Master transfers. This bit does not impact the generation of completions for split transaction commands.	R/W
1	Memory Space Enable (MSE)	The SATA Controller does not contain memory space.	RO
0	IOSE - I/O Space Enable (IOSE)	This bit controls access to the I/O space registers. 0 = Disables access to the Legacy or Native IDE ports (both Primary and Secondary) as well as the Bus Master IO registers. 1 = Enable. Note that the Base Address register for the Bus Master registers should be programmed before this bit is set.	R/W

20.1.4 Offset 06 - 07h: STS—Device Status Register (SATA–D31:F2)

Table 664. Offset 06 - 07h: STS—Device Status Register (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	06-07h	**Attribute:**	Read/Write Clear, Read-Only
Default Value:	02B0h	**Size:**	16-bit

Bits	Name	Description	Access
15	Detected Parity Error (DPE)	0 = No Parity error detected by SATA controller. 1 = SATA Controller detects a parity error on its interface.	R/WC
14	Signaled System Error (SSE)	This bit is set by the Intel® 6300ESB ICH whenever it signals SERR# (internally). The SERR_EN bit (bit 8 in the Command Register) must be '1' for this bit to be set. The following conditions can cause the generation of SERR#: A parity error is seen on address, command, or data (if the data was targeting the EHC) on the internal interface to the USBe host controller due to a parity error on Hub Interface and bit 6 of the Command register is set to 1. An EHC-initiated memory read results in a completion packet with a status other than successful on Hub Interface. The SERR on Aborts Enable bit (bit 3, offset 84h) must also be set in this case. Software clears this bit by writing a '1' to this bit location.	R/WC
13	Received Master-Abort Status (RMA)	0 = 0 Cleared by writing a '1' to it. 1 = Bus Master IDE interface function, as a master, generated a master-abort.	R/WC
12	Received Target-Abort Status (RTA)	Set when the SATA Controller receives a target abort to a cycle it generated.	
11	Signaled Target-Abort Status (STA)	Reserved as '0'.	RO
10:9	DEVSEL# Timing Status (DEVT)	01 = Hardwired; Controls the device select time for the SATA Controller's PCI interface.	RO
8	Master Data Parity Error Detected (DPD)	Set when the SATA Controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set. For the Intel® 6300ESB ICH, this bit may only be set on read completions when there is a parity error.	
7	Fast Back-to-Back Capable	Reserved as '1'.	RO
6	User Definable Features (UDF)	Reserved as '0'.	RO
5	66MHz Capable	Reserved as '1'.	RO
4	Capabilities List (CL)	Indicates the presence of a capabilities list. This bit is hardwired to a '1' indicating the presence of a valid capabilities pointer at offset 34h.	RO
3:0	Reserved	Reserved	

20.1.5 Offset 09h: PI—Programming Interface (SATA–D31:F2)

Table 665. Offset 09h: PI—Programming Interface (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	09h	***Attribute:***	Read/Write
Default Value:	8Ah	***Size:***	8-bit

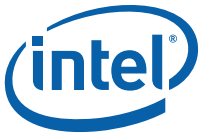
Bits	Name	Description	Access
7		This read-only bit is a '1' to indicate that the SATA Controller supports bus master operation	RO
6:4	Reserved	Reserved. Will always return 0.	
3	SOP_MODE_CAP	This read-only bit is a '1' to indicate that the secondary controller supports both legacy and native modes.	RO
2	SOP_MODE_SEL	This read-write bits determines the mode that the secondary IDE channel is operating in. 0 = Legacy-PCI mode (default) 1 = Native-PCI mode	R/W
1	POP_MODE_CAP	This read-only bit is a '1' to indicate that the primary controller supports both legacy and native modes.	RO
0	POP_MODE_SEL	This read-write bits determines the mode that the primary IDE channel is operating in. 0 = Legacy-PCI mode (default) 1 = Native-PCI mode	R/W

20.1.6 Offset 0Ah: SCC—Sub Class Code (SATA–D31:F2)

Table 666. Offset 0Ah: SCC—Sub Class Code (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	0Ah	***Attribute:***	Read-Only
Default Value:	01h or 04h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Sub Class Code	01h when Dev 31, Func 0, offset ACh, bit 23 is '0'; indicates IDE controller 04h when Dev 31, Func 0, offset ACh, bit 23 is '1'; indicates RAID controller	RO



20.1.7 Offset 0Bh: BCC—Base Class Code (SATA–D31:F2)

Table 667. Offset 0Bh: BCC—Base Class Code (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	0Bh	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Base Class Code	01 = Mass storage device	RO

20.1.8 Offset 0Dh: MLT—Master Latency Timer (SATA–D31:F2)

Table 668. Offset 0Dh: MLT—Master Latency Timer (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	0Dh	***Attribute:***	Read-Only
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Bus Master Latency	Hardwired to 00h. The IDE controller is implemented internally, and is not arbitrated as a PCI device, so it does not need a Master Latency Timer.	RO

20.1.9 Offset 10h - 13h: PCMD_BAR—Primary Command Block Base Address Register (SATA–D31:F2)

Note: This 8-byte I/O space is used in native mode for the Primary Controller's Command Block.

Table 669. Offset 10h - 13h: PCMD_BAR—Primary Command Block Base Address Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	10h-13h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:3	Base Address	Base address of the I/O space (8 consecutive I/O locations).	R/W
2:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to '1', indicating a request for IO space.	RO

20.1.10 Offset 14h - 17h: PCNL_BAR—Primary Control Block Base Address Register (SATA–D31:F2)

Note: This 4-byte I/O space is used in native mode for the Primary Controller's Control Block.

Table 670. Offset 14h - 17h: PCNL_BAR—Primary Control Block Base Address Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	14h-17h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:2	Base Address	Base address of the I/O space (4 consecutive I/O locations).	R/W
1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to '1', indicating a request for IO space.	RO

20.1.11 Offset 18h - 1Bh: SCMD_BAR—Secondary Command Block Base Address Register (IDE D31:F1)

Note: This 4-byte I/O space is used in native mode for the Secondary Controller's Control Block.

Table 671. Offset 18h - 1Bh: SCMD_BAR—Secondary Command Block Base Address Register (IDE D31:F1)

Device:	31	***Function:***	2
Offset:	18h-1Bh	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:3	Base Address	Base address of the I/O space (8 consecutive I/O locations).	R/W
2:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to '1', indicating a request for IO space.	RO

20.1.12 Offset 14h - 17h: SCNL_BAR—Secondary Control Block Base Address Register (IDE D31:F1)

Note: This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

Table 672. Offset 14h - 17h: SCNL_BAR—Secondary Control Block Base Address Register (IDE D31:F1)

Device:	31	***Function:***	2
Offset:	14h-17h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:2	Base Address	Base address of the I/O space (4 consecutive I/O locations).	R/W
1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	This bit is set to '1', indicating a request for IO space.	RO

20.1.13 Offset 20h - 23h: BAR—Legacy Bus Master Base Address Register (SATA–D31:F2)

Note: The Bus Master IDE interface function uses Base Address register 5 to request a 16-byte IO space to provide a software interface to the Bus Master functions. Only 12 bytes are actually used (6 bytes for primary, 6 bytes for secondary). Only bits [15:4] are used to decode the address.

Table 673. Offset 20h - 23h: BAR—Legacy Bus Master Base Address Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	20h-23h	***Attribute:***	Read/Write
Default Value:	00000001h	***Size:***	32-bit

Bits	Name	Description	Access
31:16	Reserved	Reserved.	
15:4	Base Address	Base address of the I/O space (16 consecutive I/O locations).	R/W
3:1	Reserved	Reserved.	
0	Resource Type Indicator (RTE)	Hardwired to '1', indicating a request for IO space.	RO

20.1.14 Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (SATA–D31:F2)

Table 674. Offset 2Ch - 2Dh: SVID—Subsystem Vendor ID (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	2Ch-2Dh	***Attribute:***	Read/Write Once
Default Value:	00h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem Vendor ID (SVID)	The SVID register, in combination with the Subsystem ID (SID) register, enables the operating system (OS) to distinguish subsystems from each other. Software (BIOS) sets the value in this register. After that, the value may be read, but subsequent writes to this register have no effect. The value written to this register will also be readable through the corresponding SVID registers for the USB#1, USB#2 and SMBus functions.	R/WO

20.1.15 Offset 2Eh - 2Fh: SID—Subsystem ID (SATA–D31:F2)

Table 675. Offset 2Eh - 2Fh: SID—Subsystem ID (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	2Eh-2Fh	***Attribute:***	Read/Write-Once
Default Value:	00h	***Size:***	16-bit
Lockable:	No	***Power Well:***	Core

Bits	Name	Description	Access
15:0	Subsystem ID (SID)	The SID register, in combination with the SVID register, enables the operating system (OS) to distinguish subsystems from each other. Software (BIOS) sets the value in this register. After that, the value may be read, but subsequent writes to this register have no effect. The value written to this register will also be readable through the corresponding SID registers for the USB#1, USB#2 and SMBus functions.	R/WO

20.1.16 Offset 34h: CAP—Capabilities Pointer Register (SATA–D31:F2)

Table 676. Offset 34h: CAP—Capabilities Pointer Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	34h	***Attribute:***	Read-Only
Default Value:	80h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Capability Pointer (CP)	This bit indicates that the first capability pointer offset is 80h, the MSI capability. This value will be 70h if the MAP register (offset 90h) indicates that the SATA and IDE functions are combined (values of 100, 101, 110, or 111).	RO

20.1.17 Offset 3Ch: INTR_LN—Interrupt Line Register (SATA–D31:F2)

Table 677. Offset 3Ch: INTR_LN—Interrupt Line Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	3Ch	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:0	Interrupt Line	It is to communicate to software the interrupt line that the interrupt pin is connected to.	R/W

20.1.18 Offset 3Dh: INTR_PN—Interrupt Pin Register (SATA–D31:F2)

Table 678. Offset 3Dh: INTR_PN—Interrupt Pin Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	3Dh	***Attribute:***	Read-Only
Default Value:	01h	***Size:***	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2:0	Interrupt Pin	The value of 01h indicates to "software" that the Intel® 6300ESB ICH will drive INTA#. Note that this is only used in native mode. Also note that the routing to the internal interrupt controller does not necessarily relate to the value in this register.	RO

20.1.19 Offset 40 - 41h: IDE_TIMP—Primary IDE Timing Register (SATA–D31:F2)

Note: This register controls the timings driven on the IDE cable for PIO and 8237 style DMA transfers. It also controls operation of the buffer for PIO transfers.

Table 679. Offset 40 - 41h: IDE_TIMP—Primary IDE Timing Register (SATA–D31:F2) (Sheet 1 of 2)

Device:	31	***Function:***	2
Offset:	Primary: 40-41h Secondary: 42-43h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15	IDE Decode Enable (IDE)	Individually enable/disable the Primary or Secondary decode. 0 = Disable. 1 = Enables the Intel® 6300ESB ICH to decode the associated Command Blocks (1F0-1F7h for primary, 170-177h for secondary) and Control Block (3F6h for primary and 376h for secondary). This bit effects the IDE decode ranges for both legacy and native-Mode decoding.	R/W
14	Drive 1 Timing Register Enable (SITRE)	0 = Use bits 13:12, 9:8 for both drive 0 and drive 1. 1 = Use bits 13:12, 9:8 for drive 0, and use the Slave IDE Timing register for drive 1	R/W
13:12	IORDY Sample Point (ISP)	The setting of these bits determine the number of PCI clocks between IDE IOR#/IOW# assertion and the first IORDY sample point. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	

Table 679. Offset 40 - 41h: IDE_TIMP—Primary IDE Timing Register (SATA–D31:F2) (Sheet 2 of 2)

Device:	31	**Function:**	2
Offset:	Primary: 40-41h Secondary: 42-43h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
11:10	Reserved	Reserved.	
9:8	Recovery Time (RCT)	The setting of these bits determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clock	R/W
7	Drive 1 DMA Timing Enable (DTE1)	0 = Disable. 1 = Enable the fast timing mode for DMA transfers only for this drive. PIO transfers to the IDE data port will run in compatible timing.	R/W
6	Drive 1 Prefetch/Posting Enable (PPE1)	0 = Disable. 1 = Enable Prefetch and posting to the IDE data port for this drive.	R/W
5	Drive 1 IORDY Sample Point Enable (IE1)	0 = Disable IORDY sampling for this drive. 1 = Enable IORDY sampling for this drive.	R/W
4	Drive 1 Fast Timing Bank (TIME1)	0 = Accesses to the data port will use compatible timings for this drive. 1 = When this bit ='1' and bit 14 = '0', accesses to the data port will use bits 13:12 for the IORDY sample point, and bits 9:8 for the recovery time. When this bit = '1' and bit 14 = '1', accesses to the data port will use the IORDY sample point and recover time specified in the slave IDE timing register.	R/W
3	Drive 0 DMA Timing Enable (DTE0)	0 = Disable 1 = Enable fast timing mode for DMA transfers only for this drive. PIO transfers to the IDE data port will run in compatible timing.	R/W
2	Drive 0 Prefetch/Posting Enable (PPE0)	0 = Disable prefetch and posting to the IDE data port for this drive. 1 = Enable prefetch and posting to the IDE data port for this drive.	R/W
1	Drive 0 IORDY Sample Point Enable (IE0)	0 = Disable IORDY sampling is disabled for this drive. 1 = Enable IORDY sampling for this drive.	R/W
0	Drive 0 Fast Timing Bank (TIME0)	0 = Accesses to the data port will use compatible timings for this drive. 1 = Accesses to the data port will use bits 13:12 for the IORDY sample point, and bits 9:8 for the recovery time.	R/W

20.1.20 IDE_TIMS—Secondary IDE Timing Register (SATA–D31:F2)

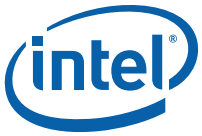
Note: See the above register description for “Primary IDE Timing Register.”

20.1.21 Offset 44h: SIDETIM—Slave IDE Timing Register (SATA–D31:F2)

Table 680. Offset 44h: SIDETIM—Slave IDE Timing Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	44h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:6	Secondary Drive 1 IORDY Sample Point (SISP1)	Determines the number of PCI clocks between IDE IOR#/IOW# assertion and the first IORDY sample point, when the access is to drive 1 data port and bit 14 of the IDE timing register for secondary is set. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	R/W
5:4	Secondary Drive 1 Recovery Time (SRCT1)	Determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle, when the access is to drive 1 data port and bit 14 of the IDE timing register for secondary is set. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clocks	R/W
3:2	Primary Drive 1 IORDY Sample Point (PISP1)	Determines the number of PCI clocks between IOR#/IOW# assertion and the first IORDY sample point, when the access is to drive 1 data port and bit 14 of the IDE timing register for primary is set. 00 = 5 clocks 01 = 4 clocks 10 = 3 clocks 11 = Reserved	R/W
1:0	Primary Drive 1 Recovery Time (PRCT1)	Determines the minimum number of PCI clocks between the last IORDY sample point and the IOR#/IOW# strobe of the next cycle, when the access is to drive 1 data port and bit 14 of the IDE timing register for primary is set. 00 = 4 clocks 01 = 3 clocks 10 = 2 clocks 11 = 1 clocks	R/W



20.1.22 Offset 48h: SDMA_CNT—Synchronous DMA Control Register (SATA–D31:F2)

Table 681. Offset 48h: SDMA_CNT—Synchronous DMA Control Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	48h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7:4	Reserved	Reserved.	
3	Secondary Drive 1 Synchronous DMA Mode Enable (SSDE1)	0 = Disable (default) 1 = Enable Synchronous DMA mode for secondary channel drive 1.	R/W
2	Secondary Drive 0 Synchronous DMA Mode Enable (SSDE0)	0 = Disable (default) 1 = Enable Synchronous DMA mode for secondary drive 0.	R/W
1	Primary Drive 1 Synchronous DMA Mode Enable (PSDE1)	0 = Disable (default) 1 = Enable Synchronous DMA mode for primary channel drive 1.	R/W
0	Primary Drive 0 Synchronous DMA Mode Enable (PSDE0)	0 = Disable (default) 1 = Enable Synchronous DMA mode for primary channel drive 0.	R/W

20.1.23 Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (SATA–D31:F2)

Table 682. Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (SATA–D31:F2) (Sheet 1 of 2)

Device:	31	***Function:***	2
Offset:	4A-4Bh	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:14	Reserved	Reserved.	
13:12	Secondary Drive 1 Cycle Time (SCT1)	For Ultra ATA mode. The setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **SCB1 = 0 (33MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **SCB1 = '1' (66MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_SCB1 = '1' (133MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W
11:10	Reserved	Reserved.	
9:8	Secondary Drive 0 Cycle Time (SCT0)	For Ultra ATA mode. The setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **SCB1 = 0 (33MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **SCB1 = '1' (66MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_SCB1 = '1' (133MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W
7:6	Reserved	Reserved.	

Table 682. Offset 4A - 4Bh: SDMA_TIM—Synchronous DMA Timing Register (SATA–D31:F2) (Sheet 2 of 2)

Device:	31	**Function:**	2
Offset:	4A-4Bh	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
5:4	Primary Drive 1 Cycle Time (PCT1)	For Ultra ATA mode, the setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **PCB1 = 0 (33MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **PCB1 = '1' (66MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_PCB1 = '1' (133MHz clk)** 01 = CT 3 clks, RP 16 clks 00 = Reserved 10 = Reserved 11 = Reserved	R/W
3:2	Reserved	Reserved.	
1:0	Primary Drive 0 Cycle Time (PCT0)	For Ultra ATA mode, the setting of these bits determines the minimum write strobe cycle time (CT). The DMARDY#-to-STOP (RP) time is also determined by the setting of these bits. **PCB1 = 0 (33MHz clk)** 00 = CT 4 clocks, RP 6 clocks 01 = CT 3 clocks, RP 5 clocks 10 = CT 2 clocks, RP 4 clocks 11 = Reserved **PCB1 = '1' (66MHz clk)** 00 = Reserved 01 = CT 3 clocks, RP 8 clocks 10 = CT 2 clocks, RP 8 clocks 11 = Reserved **FAST_PCB1 = '1' (133MHz clk)** 00 = Reserved 01 = CT 3 clks, RP 16 clks 10 = Reserved 11 = Reserved	R/W

20.1.24 Offset 54h: IDE_CONFIG—IDE I/O Configuration Register (SATA–D31:F2)

Table 683. Offset 54h: IDE_CONFIG—IDE I/O Configuration Register (SATA–D31:F2) (Sheet 1 of 2)

Device:	31	***Function:***	2
Offset:	54h	***Attribute:***	Read-Write
Default Value:	00h	***Size:***	32-bit

Bits	Name	Description	Access
31:24	Reserved	Reserved.	
23:20	Scratchpad (SP2)	The Intel® 6300ESB ICH does not perform any actions on these bits.	
19:18	Reserved	Reserved.	R/W
17:16	Reserved	Reserved.	R/W
15	FAST_SCB1: Fast Secondary Drive 1 Base Clock	This bit is used in conjunction with the SCT1 bits to enable/ disable Ultra ATA/100 timings for the Secondary Slave drive. 0 = Disable Ultra ATA/100 timing for the Secondary Slave drive. 1 = Enable Ultra ATA/100 timing for the Secondary Slave drive (overrides bit 3 in this register).	R/W
14	FAST_SCB0: Fast Secondary Drive 0 Base Clock	This bit is used in conjunction with the SCT0 bits to enable/ disable Ultra ATA/100 timings for the Secondary Master drive. 0 = Disable Ultra ATA/100 timing for the Secondary Master drive. 1 = Enable Ultra ATA/100 timing for the Secondary Master drive (overrides bit 2 in this register).	R/W
13	FAST_PCB1: Fast Primary Drive 1 Base Clock	This bit is used in conjunction with the PCT1 bits to enable/ disable Ultra ATA/100 timings for the Primary Slave drive. 0 = Disable Ultra ATA/100 timing for the Primary Slave drive. 1 = Enable Ultra ATA/100 timing for the Primary Slave drive (overrides bit '1' in this register).	R/W
12	FAST_PCB0: Fast Primary Drive 0 Base Clock	This bit is used in conjunction with the PCT0 bits to enable/ disable Ultra ATA/100 timings for the Primary Master drive. 0 = Disable Ultra ATA/100 timing for the Primary Master drive. 1 = Enable Ultra ATA/100 timing for the Primary Master drive (overrides bit '0' in this register).	R/W
11:8	Reserved	Reserved.	
7:4	Scratchpad (SP1)	The Intel® 6300ESB ICH does not perform any action on these bits.	
3	SCB1: Secondary Drive 1 Base Clock	0 = 33 MHz base clock for Ultra ATA timings. 1 = 66 MHz base clock for Ultra ATA timings	R/W

Table 683. Offset 54h: IDE_CONFIG—IDE I/O Configuration Register (SATA–D31:F2) (Sheet 2 of 2)

Device:	31	*Function:*	2
Offset:	54h	*Attribute:*	Read-Write
Default Value:	00h	*Size:*	32-bit

Bits	Name	Description	Access
2	SCBO: Secondary Drive 0 Base Clock	0 = 33 MHz base clock for Ultra ATA timings. 1 = 66 MHz base clock for Ultra ATA timings	R/W
1	PCB1: Primary Drive 1 Base Clock	0 = 33 MHz base clock for Ultra ATA timings. 1 = 66 MHz base clock for Ultra ATA timings	R/W
0	PCB0: Primary Drive 0 Base Clock	0 = 33 MHz base clock for Ultra ATA timings. 1 = 66 MHz base clock for Ultra ATA timings	R/W

20.1.25 Offset 70 - 71h: PID—PCI Power Management Capability ID (SATA–D31:F2)

Table 684. Offset 70 - 71h: PID—PCI Power Management Capability ID (SATA–D31:F2)

Device:	31	*Function:*	2
Offset:	70-71h	*Attribute:*	Read-Only
Default Value:	0001h	*Size:*	16-bit

Bits	Name	Description	Access
15:8	Next Capability (NEXT)	Indicates that this is the last item in the list	
7:0	Cap ID (CID)	Indicates that this pointer is a PCI power management.	

20.1.26 Offset 72 - 73h: PC—PCI Power Management Capabilities (SATA–D31:F2)

Table 685. Offset 72 - 73h: PC—PCI Power Management Capabilities (SATA–D31:F2)

Device:	31	*Function:*	2
Offset:	72-73h	*Attribute:*	Read-Only
Default Value:	0002	*Size:*	16-bit

Bits	Name	Description	Access
15:11	PME_Support	Indicates PME# cannot be generated form the SATA host controller. When in low power state, resume events are not allowed.	
10	D2_Support	The D2 state is not supported	
9	D1_Support	The D1 state is not supported	

Table 685. Offset 72 - 73h: PC—PCI Power Management Capabilities (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	72-73h	**Attribute:**	Read-Only
Default Value:	0002	**Size:**	16-bit

Bits	Name	Description	Access
8:6	Aux_Current	Reports 375mA maximum Suspend well current required when in the D3cold state.	
5	Device Specific Initial-ization (DSI)	Indicates that no device-specific initialization is required.	
4	Reserved	Reserved.	
3	PME Clock (PMEC)	Indicates that PCI clock is not required to generate PME#.	
2:0	Version (VS)	Indicates support for Revision 1.1 of the PCI Power Management Specification.	

20.1.27 Offset 74 - 75h: PMCS—PCI Power Management Control and Status (SATA–D31:F2)

Table 686. Offset 74 - 75h: PMCS—PCI Power Management Control and Status (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	74-75h	**Attribute:**	Read-Only, Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15	PME Status (PMES)	Reserved as ‘0’.	
14:9	Reserved	Reserved.	
8	PME Enable (PMEE).	Reserved as ‘0’.	
7:2	Reserved	Reserved.	
1:0	Power State (PS)	**Power State (PS).** These bits are used both to determine the current power state of the SATA Controller and to set a new power state. 00: D0 state 01: D1 state 10: D2 state 11: D3hot state When in the D3hot state, the controller’s configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked.	

20.1.28 Offset 80 - 81h: MID—Message Signaled Interrupt Identifiers (SATA–D31:F2)

Table 687. Offset 80 - 81h: MID—Message Signaled Interrupt Identifiers (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	80-81h	***Attribute:***	Read-Only
Default Value:	7005h	***Size:***	16-bit

Bits	Name	Description	Access
15:8	Next Pointer (NEXT)	Indicates that the next item in the list the PCI power management pointer.	
7:0	Capability ID (CID)	Capability ID indicates MSI.	

20.1.29 Offset 82 - 83h: MC—Message Signaled Interrupt Message Control (SATA–D31:F2)

Table 688. Offset 82 - 83h: MC—Message Signaled Interrupt Message Control (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	82-83h	***Attribute:***	Read-Only, Read/Write
Default Value:	0000h	***Size:***	16-bit

Bits	Name	Description	Access
15:8	Reserved	Reserved.	
7	64 Bit Address Capable (C64)	Capable of generating 32-bit message only.	RO
6:4	Multiple Message Enable (MME)	These bits are R/W for software compatibility, but only one message is ever sent by the Intel® 6300ESB ICH.	R/W
3:1	Multiple Message Capable (MMC)	Only one message is required.	RO
0	MSI Enable (MSIE)	0 = Disabled. 1 = MSI is enabled and traditional interrupt pins are not used to generate interrupts.	R/W

20.1.30 Offset 84 - 87h: MA—Message Signaled Interrupt Message Address (SATA–D31:F2)

Table 689. Offset 84 - 87h: MA—Message Signaled Interrupt Message Address (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	84-87h	***Attribute:***	Read/Write
Default Value:	0000h	***Size:***	32-bit

Bits	Name	Description	Access
31:2	Address (ADDR)	Lower 32 bits of the system specified message address, always DWORD aligned.	
1:0	Reserved	Reserved.	

20.1.31 Offset 88 - 89h: MD—Message Signaled Interrupt Message Data (SATA–D31:F2)

Table 690. Offset 88 - 89h: MD—Message Signaled Interrupt Message Data (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	88-89h	**Attribute:**	Read/Write
Default Value:	0000h	**Size:**	16-bit

Bits	Name	Description	Access
15:0	Data (DATA)	This field is programmed by system software when MSI is enabled. Its content is driven onto the lower word (PCI AD[15:0]) during the data phase of the MSI memory write transaction.	

20.1.32 Offset 90h: MAP—Address Map (SATA–D31:F2)

Table 691. Offset 90h: MAP—Address Map (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	90h	**Attribute:**	Read-Only, Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7:3	Reserved	Reserved.	
2:0	Map Value	The value of these bits indicate the address range the SATA port responds to, and whether or not the SATA and IDE functions are combined. 000 = Non-combined. P0 is primary master. P1 is secondary master. 001 = Non-combined. P0 is secondary master. P1 is primary master. 100 = Combined. P0 is primary master. P1 is primary slave. P-ATA is secondary. 101 = Combined. P0 is primary slave. P1 is primary master. P-ATA is secondary. 110 = Combined. P-ATA is primary. P0 is secondary master. P1 is secondary slave. 111 = Combined. P-ATA is primary. P0 is secondary slave. P1 is secondary master.	R/W

20.1.33 Offset 92h: PCS—Port Status and Control (SATA–D31:F2)

Table 692. Offset 92h: PCS—Port Status and Control (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	92h	***Attribute:***	Read/Write
Default Value:	000h	***Size:***	16-bit

Bits	Name	Description	Access
15:6	Reserved	Reserved.	
5	Reserved	Reserved. Bit is Read Only, reset to '0' but may be '1' at any given time	
4	Reserved	Reserved. Bit is Read Only, reset to '0' but may be '1' at any given time.	
3:2	Reserved	Reserved.	
1	Port 1 Enabled (P1E)	0 = The port is disabled. The port is in the 'off' state and cannot detect any devices. 1 = The port is enabled. The port may transition between the on, partial, and slumber states and may detect devices.	
0	Port 0 Enabled (P0E)	0 = The port is disabled. The port is in the 'off' state and cannot detect any devices. 1 = The port is enabled. The port may transition between the on, partial, and slumber states and may detect devices.	

20.1.34 Offset A0h: SRI—SATA Registers Index (SATA–D31:F2)

Table 693. Offset A0h: SRI—SATA Registers Index (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	A0h	***Attribute:***	Read/Write
Default Value:	00h	***Size:***	8-bit

Bits	Name	Description	Access
7	Reserved	Reserved.	
6:0	Index (IDX)	This field is a 7-bit index pointer into the SATA Registers space. Data is written into the SRD register (D31:F2:A4h) and read from the SRD register.	R/W

Table 693. Offset A0h: SRI—SATA Registers Index (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	A0h	**Attribute:**	Read/Write
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description		Access
		Index	Name	
		00h–01h	SATA TX Termination Test Register (STT)	
		02h–1Dh	Reserved	
		1Eh	SATA TX Output Test Register (STOT)	
		1Fh–53h	Reserved	
		54h–57h	SError Register Port 0 (SER0)	
		58h–63h	Reserved	
		64h–67h	SError Register Port 1 (SER1)	
		68h–FFh	Reserved	

20.1.35 Offset A4h - A7h: SRD—SATA Registers Data (SATA–D31:F2)

Table 694. Offset A4h - A7h: SRD—SATA Registers Data (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	A4h–A7h	**Attribute:**	Read/Write
Default Value:	XXh	**Size:**	8-bit

Bits	Name	Description	Access
31:0	Data (DTA)	This field is a 32-bit data value that is written to the register pointed to by SRI (D31:F2:A0h) or read from the register pointed to by SRI.	R/W

20.1.36 STTT—SATA TX Termination Test Register A (SATA–D31:F2)

Table 695. STTT—SATA TX Termination Test Register A (SATA–D31:F2)

Device:	31	**Function:**	2
Index Address:	Index 00h–01h	**Attribute:**	Read/Write
Default Value:	XXXXh	**Size:**	16-bit

Bits	Name	Description	Access
15:2	Reserved	Reserved.	
1	Port 1 TX Termination Test Enable	Setting this bit will enable testing of the port's TX termination. This bit is only to be used for system board testing.	R/W
0	Port 0 TX Termination Test Enable	Setting this bit will enable testing of the port's TX termination. This bit is only to be used for system board testing.	R/W

20.1.37 STOT — SATA TX Output Test Register (SATA–D31:F2)

Table 696. STOT — SATA TX Output Test Register (SATA–D31:F2)

Device:	31	**Function:**	2
Index Address:	Index 1Eh	**Attribute:**	Read/Write
Default Value:	XXXXh	**Size:**	16-bit

Bits	Name	Description	Access
15:2	Reserved	Reserved.	
1	Force ALIGN TX Bit	This bit will force the Intel® 6300ESB ICH to repeatedly transmit the SATA ALIGN primitive when set. This bit is only used for system board testing.	R/W
0	Reserved	Reserved.	R/W

20.1.38 Offset Index 54h - 57h: SER0—SATA SError Register Port 0 (SATA–D31:F2)

Table 697. Offset Index 54h - 57h: SER0—SATA SError Register Port 0 (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	Index 54h–57h	***Attribute:***	Read/Write
Default Value:	XXXXXXXXh	***Size:***	32-bit

Bits	Name	Description	Access
31:0	SER0	This register is implemented in accordance with the SERROR register description in Section 10.1.2 of the *SATA 1.0 Specification*.	R/W

20.1.39 Offset Index 64h - 67h: SER1—SATA SError Register Port 1 (SATA–D31:F2)

Table 698. Offset Index 64h - 67h: SER1—SATA SError Register Port 1 (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	Index 64h–67h	***Attribute:***	Read/Write
Default Value:	XXXXXXXXh	***Size:***	32-bit

Bits	Name	Description	Access
31:0	SER1	This register is implemented in accordance with the SERROR register description in Section 10.1.2 of the *SATA 1.0 Specification*.	R/W

20.1.40 Offset E0h - E3h: BFCS—BIST FIS Control/Status Register (SATA–D31:F2)

Table 699. Offset E0h - E3h: BFCS—BIST FIS Control/Status Register (SATA–D31:F2) (Sheet 1 of 3)

Device:	31	***Function:***	2
Offset:	E0h–E3h	***Attribute:***	Read/Write, Read/Write Clear
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:12	Reserved	Reserved.	

Table 699. Offset E0h - E3h: BFCS—BIST FIS Control/Status Register (SATA–D31:F2) (Sheet 2 of 3)

Device:	31	**Function:**	2
Offset:	E0h–E3h	**Attribute:**	Read/Write, Read/Write Clear
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
11	BIST FIS Successful (BFS)	0 = Software clears this bit by writing a '1' to it. 1 = This bit is set any time a BIST FIS transmitted by the Intel® 6300ESB ICH receives an R_OK completion status from the device. **NOTE:** This bit must be cleared by software prior to initiating a BIST FIS.	R/WC
10	BIST FIS Failed (BFF)	0 = Software clears this bit by writing a '1' to it. 1 = This bit is set any time a BIST FIS transmitted by the Intel® 6300ESB ICH receives an R_ERR completion status from the device. **NOTE:** This bit must be cleared by software prior to initiating a BIST FIS.	R/WC
9	Port 1 BIST FIS Initiate (P1BFI)	When a rising edge is detected on this bit field, the Intel® 6300ESB ICH initiates a BIST FIS to the device on Port 1, using the parameters specified in this register and the data specified in BFTD1 and BFTD2. The BIST FIS will only be initiated if a device on Port 1 is present and ready (not partial/slumber state). After a BIST FIS is successfully completed, software must disable and re-enable the port using the PxE bits at offset 92h prior to attempting additional BIST FISes or to return the Intel® 6300ESB ICH to a normal operational mode. If the BIST FIS fails to complete, as indicated by the BFF bit in the register, then software can clear then set the P1BFI bit to initiate another BIST FIS. This can be retried until the BIST FIS eventually completes successfully.	R/W
8	Port 0 BIST FIS Initiate (P0BFI)	When a rising edge is detected on this bit field, the Intel® 6300ESB ICH initiates a BIST FIS to the device on Port 0, using the parameters specified in this register and the data specified in BFTD1 and BFTD2. The BIST FIS will only be initiated if a device on Port 0 is present and ready (not partial/slumber state). After a BIST FIS is successfully completed, software must disable and re-enable the port using the PxE bits at offset 92h prior to attempting additional BIST FISes or to return the Intel® 6300ESB ICH to a normal operational mode. If the BIST FIS fails to complete, as indicated by the BFF bit in the register, then software can clear then set the P0BFI bit to initiate another BIST FIS. This can be retried until the BIST FIS eventually completes successfully.	R/W

Table 699. Offset E0h - E3h: BFCS—BIST FIS Control/Status Register (SATA–D31:F2) (Sheet 3 of 3)

Device:	31	**Function:**	2
Offset:	E0h–E3h	**Attribute:**	Read/Write, Read/Write Clear
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
7:2	BIST FIS Parameters	These 6 bits form the contents of the upper 6 bits of the BIST FIS Pattern Definition in any BIST FIS transmitted by the Intel® 6300ESB ICH. This field is not port specific — its contents will be used for any BIST FIS initiated on port 0 on port 1. The specific bit definitions are: Bit 7: T – Far End Transmit mode Bit 6: A – Align Bypass mode Bit 5: S – Bypass Scrambling Bit 4: L – Far End Retimed Loopback Bit 3: F – Far End Analog Loopback Bit 2: P – Primitive bit for use with Transmit mode	
1:0	Reserved	Reserved.	

20.1.41 Offset E4h - E7h: BFTD1—BIST FIS Transmit Data1 Register (SATA–D31:F2)

Table 700. Offset E4h - E7h: BFTD1—BIST FIS Transmit Data1 Register (SATA–D31:F2)

Device:	31	**Function:**	2
Offset:	E4h–E7h	**Attribute:**	Read/Write
Default Value:	00000000h	**Size:**	32-bit

Bits	Name	Description	Access
31:0	BIST FIS Transmit Data 1	The data programmed into this register will form the contents of the second DWord of any BIST FIS initiated by the Intel® 6300ESB ICH. This register is not port specific — its contents will be used for BIST FIS initiated on port 0 or port 1. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the ‘T’ bit of the BIST FIS is set to indicate “Far-End Transmit mode”, this register’s contents will be transmitted as the BIST FIS 2nd DW regardless of whether or not the ‘T’ bit is indicated in the BFCS register.	R/W

20.1.42 Offset E8h - EBh: BFTD2—BIST FIS Transmit Data2 Register (SATA–D31:F2)

Table 701. Offset E8h - EBh: BFTD2—BIST FIS Transmit Data2 Register (SATA–D31:F2)

Device:	31	***Function:***	2
Offset:	E8h–EBh	***Attribute:***	Read/Write
Default Value:	00000000h	***Size:***	32-bit

Bits	Name	Description	Access
31:0	BIST FIS Transmit Data 2	The data programmed into this register will form the contents of the third DWord of any BIST FIS initiated by the Intel® 6300ESB ICH. This register is not port specific — its contents will be used for BIST FIS initiated on port 0 or port 1. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the 'T' bit of the BIST FIS is set to indicate "Far-End Transmit mode", this register's contents will be transmitted as the BIST FIS 3rd DW regardless of whether or not the 'T' bit is indicated in the BFCS register.	R/W

20.2 Bus Master IDE I/O Registers (D31:F2)

The bus master IDE function uses 16 bytes of I/O space, allocated through the BMIBA register, located in Device 31:Function 1 Configuration space, offset 20h. All bus master IDE I/O space registers may be accessed as byte, word, or DWORD quantities. Reading reserved bits returns an indeterminate, inconsistent value, and writes to reserved bits have no affect (but should not be attempted). The description of the I/O registers is shown below in Table 702.

Table 702. Bus Master IDE I/O Registers

Offset	Mnemonic	Register	Default	Type
00	BMICP	Command Register Primary	00h	R/W
01		Reserved		RO
02	BMISP	Status Register Primary	00h	R/WC
03		Reserved		RO
04-07	BMIDP	Descriptor Table Pointer Primary	xx	R/W
08	BMICS	Command Register Secondary	00h	R/W
09		Reserved		RO
0A	BMISS	Status Register Secondary	00h	R/WC
0B		Reserved		RO
0C-0F	BMIDS	Descriptor Table Pointer Secondary	xx	R/W

20.2.1 BMIC[P,S]—Bus Master IDE Command Register (D31:F2)

Table 703. BMIC[P,S]—Bus Master IDE Command Register (D31:F2)

Device:	31	***Function:***	2
Offset:	Primary: 00h Secondary: 08h	***Attribute:***	Read/Write
Default Value:	01h	***Size:***	8-bit

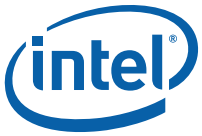
Bits	Name	Description	Access
7:4	Reserved	Reserved. Returns '0'.	
3	Read / Write Control (RWC)	This bit sets the direction of the bus master transfer: This bit must NOT be changed when the bus master function is active. 0 = Memory reads 1 = Memory writes	R/W
2:1	Reserved	Reserved. Returns '0'.	
0	Start/Stop Bus Master (START)	0 = All state information is lost when this bit is cleared. Master mode operation cannot be stopped and then resumed. When this bit is reset while bus master operation is still active (i.e., the Bus Master IDE Active bit of the Bus Master IDE Status register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded instead of being written to system memory. 1 = Enables bus master operation of the controller. Bus master operation begins when this bit is detected changing from a '0' to a '1'. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation may be halted by writing a '0' to this bit. **NOTE:** This bit is intended to be cleared by software after the data transfer is completed, as indicated by either the Bus Master IDE Active bit being cleared or the Interrupt bit of the Bus Master IDE Status register for that IDE channel being set, or both. Hardware does not clear this bit automatically.	R/W

20.2.2 BMIS[P,S]—Bus Master IDE Status Register (D31:F2)

Table 704. BMIS[P,S]—Bus Master IDE Status Register (D31:F2)

Device:	31	**Function:**	2
Offset:	Primary: 02h Secondary: 0Ah	**Attribute:**	Read/Write Clear
Default Value:	00h	**Size:**	8-bit

Bits	Name	Description	Access
7	PRD Interrupt Status (PRDIS)	0 = When this bit is cleared by software, the interrupt is cleared. 1 = This bit is set when the host control execution of a PRD that has its PRD_INT bit set.	R/WC
6	Drive 1 DMA Capable	0 = Not Capable. 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The Intel® 6300ESB ICH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.	R/W
5	Drive 0 DMA Capable	0 = Not Capable 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The Intel® 6300ESB ICH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.	R/W
4:3	Reserved	Reserved. Returns '0'.	
2	Interrupt	Software may use this bit to determine if an IDE device has asserted its interrupt line (IRQ 14 for the Primary channel, and IRQ 15 for Secondary). 0 = This bit is cleared by software writing a '1' to the bit position. When this bit is cleared while the interrupt is still active, this bit will remain clear until another assertion edge is detected on the interrupt line. 1 = Set by the rising edge of the IDE interrupt line, regardless of whether or not the interrupt is masked in the 8259 or the internal I/O APIC. When this bit is read as a '1', all data transferred from the drive is visible in system memory.	R/WC
1	Error	0 = This bit is cleared by software writing a '1' to the bit position. 1 = This bit is set when the controller encounters a target abort or master abort when transferring data on PCI.	R/WC
0	Bus Master IDE Active (ACT)	0 = This bit is cleared by the Intel® 6300ESB ICH when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared by the Intel® 6300ESB ICH when the Start bit is cleared in the Command register. When this bit is read as a '0', all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted. 1 = Set by the Intel® 6300ESB ICH when the Start bit is written to the Command register.	RO



20.2.3 BMID[P,S]—Bus Master IDE Descriptor Table Pointer Register (D31:F2)

Table 705. BMID[P,S]—Bus Master IDE Descriptor Table Pointer Register (D31:F2)

Device:	31	***Function:***	2
Offset:	Primary: 04h Secondary: 0Ch	***Attribute:***	Read/Write
Default Value:	All bits undefined	***Size:***	32-bit

Bits	Name	Description	Access
31:2	Address of Descriptor Table (ADDR)	Corresponds to A[31:2]. The Descriptor Table must be DWORD-aligned. The Descriptor Table must not cross a 64-K boundary in memory.	R/W
1:0	Reserved	Reserved.	

November 2007
Order Number: 300641-004US

Package Information 21

21.1 Ball Location

Figure 36. Ball Diagram (Top View - Left Side)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A			VSS	AD[16]	AD[20]	PIRQ[B]#	VSS	GPIO[5] / PIRQ[H]#	PIRQ[C]#	GPIO[3] / PIRQ[F]#	NC	NC	NC	LAD[3]
B		VSS	AD[7]	AD[11]	STOP#	AD[22]	AD[18]	PIRQ[D]#	VSS	GPIO[2] / PIRQ[E]#	REQ[2]#	NC	NC	LFRAME#
C	VSS	C/BE[0]#	AD[2]	VSS	Vcc3_3	VSS	AD[15]	VSS	Vcc3_3	VSS	GNT[2]#	VSS	Vcc3_3	VSS
D	AD[14]	PERR#	Vcc3_3	AD[5]	AD[9]	AD[6]	VSS	REQ[0]#	GNT[1]#	GPIO[4] / PIRQ[G]#	VSS	REQ[3]#	VSS	THRM#
E	AD[17]	AD[10]	IRDY#	SERR#	VSS	Vcc3_3	AD[26]	AD[24]	Vcc3_3	GNT[0]#	V5REF	PIRQ[A]#	NC	VSS
F	AD[27]	AD[19]	VSS	C/BE[2]#	AD[3]	AD[13]	VSS	VSS	TRDY#	AD[4]	VSS	Vcc1_5	NC	LDRQ[0]#
G	GPIO[33] / PXIRQ[0]#	AD[31]	Vcc3_3	VSS	DEVSEL#	AD[12]	C/BE[1]#	AD[1]	PAR	FRAME#	AD[30]	REQ[1]#	GNT[3]#	LDRQ[1]#
H	GPIO[34] / PXIRQ[1]#	GPIO[35] / PXIRQ[2]#	Vcc3_3	AD[29]	C/BE[3]#	VSS	AD[23]	AD[8]	Vcc3_3	VSS	AD[0]	AD[28]		
J	GPIO[0] / PXREQ[2]#	PXPCLKO[0]	GPIO[36] / PXIRQ[3]#	VSS	PCICLK	PLOCK#	Vcc3_3	Vcc1_5						
K	PXGNT0#	PXAD[31]	Vcc3_3	PXPCLKO[3]	Vcc3_3	PXPCLKO[2]	AD[25]	AD[21]						
L	PXAD[27]	PXREQ[0]#	PXAD[28]	VSS	PXPCLKO[4]	PXREQ[1]#	VSS	PXPCLKO[1]						
M	PXAD[25]	PXAD[26]	VSS	Vcc3_3	PXAD[30]	VSS	PXAD[29]	Vcc3_3				VSS	VSS	VSS
N	PXAD[22]	PXAD[20]	PXAD[23]	PXAD[24]	Vcc3_3	PXC/BE[3]#	Vcc1_5					VSS	VSS	VSS
P	PXAD[17]	PXAD[18]	VSS	Vcc3_3	PXAD[19]	VSS	PXAD[21]					VSS	VSS	VSS
R	PXAD[16]	PXC/BE[2]#	Vcc3_3	PXIRDY#	Vcc3_3	PXFRAME#	PXPCIXCAP					VSS	VSS	VSS
T	PXPLOCK#	PXSTOP#	PXDEVSEL#	VSS	PXTRDY#	VSS	PXAD[15]					VSS	VSS	VSS
U	PXSERR#	PXC/BE[1]#	VSS	PXPAR	Vcc3_3	PXPERR#	Vcc3_3					VSS	VSS	VSS
V	PXAD[12]	PXAD[11]	VSS	PXAD[13]	VSS	VCCREF	PXAD[2]	Vcc1_5				VSS	VSS	VSS
W	PXAD[8]	PXAD[9]	Vcc3_3	PXM66EN	Vcc3_3	Vcc3_3	PXAD[1]	PXAD[5]						
Y	PXAD[14]	PXAD[6]	VSS	PXAD[3]	VSS	PXAD[43]	GPIO[17] / PXGNT[3]#	GPIO[1] / PXREQ[3]#						
AA	PXC/BE[0]#	PXAD[4]	Vcc3_3	PXAD[10]	Vcc3_3	PXAD[7]	PCIXSBRST#	VCCPLL0						
AB	PXAD[47]	PXAD[45]	PXAD[42]	VSS	PXGNT1#	Vcc3_3	PXAD[44]	Vcc1_5	PXAD[54]	Vcc1_5	Vcc3_3	VSS		
AC	PXAD[46]	GPIO[16] / PXGNT[2]#	VSS	PXPCLKI	PXPCICLK	PXAD[39]	VSS	VCCREF	Vcc3_3	PXAD[50]	VSS	PXRCOMP	Vcc1_5	GPIO[13]
AD	PXAD[41]	PXAD[40]	Vcc3_3	PXAD[38]	VSS	PXAD[37]	PXAD[36]	PXPAR64	VSS	PXAD[48]	VSS	VSS	NC	SLP_S4#
AE	RASERR#	PXAD[35]	PXAD[34]	PXAD[33]	PXREQ64#	VSS	PXC/BE[4]#	Vcc3_3	PXAD[51]	PXAD[49]	Vcc3_3	GPIO[8]	VSS	GPIO[24]
AF	PXAD[32]	PXAD[0]	VSS	PXACK64#	Vcc3_3	PXC/BE[7]#	PXC/BE[6]#	VSS	PXAD[53]	VSS	GPIO[12]	VSS	VccSus3_3	VSS
AG	VSS	PXC/BE[5]#	Vcc3_3	VSS	PXAD[60]	VSS	Vcc3_3	VSS	VSS	GPIO[27]	VSS	VccSus3_3	VSS	SLP_S3#
AH		VSS	PXAD[63]	PXAD[62]	PXAD[59]	PXAD[57]	PXAD[55]	PXPCIRST#	GPIO[25]	RI#	SYSRESET#	SMLINK[1]	SMLINK[0]	SMBCLK
AJ			VSS	PXAD[61]	PXAD[58]	PXAD[56]	PXAD[52]	PME#	GPIO[28]	PWRBTN#	SLP_S5#	SUSCLK	SMBDATA	NC
	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Figure 37. Ball Diagram (Top View - Right Side)

15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	
LAD[2]	SIU0_RI#	SIU0_TXD	UART_CLK	SIU1_DCD#	SIU1_CTS#	NC	NC	VccSus3_3	USBP3P	VSS	USBP2P	VSS			A
LAD[1]	SIU0_DCD#	SIU0_RXD	SIU1_DTR#	SIU1_DSR#	NC	NC	GPIO[7]	VccSus3_3	USBP3N	VSS	USBP2N	VSS	VSS		B
LAD[0]	VSS	Vcc3_3	VSS	SIU1_TXD	VSS	GPIO[6]	GPIO[56]	VccSus3_3	VSS	USBP1N	VSS	USBP0P	VSS	VSS	C
Vcc3_3	SIU0_CTS#	SIU1_RI#	SIU1_RXD	NC	GPIO[37]	VccSus1_5	GPIO[57]	VccSus3_3	VccSus3_3	USBP1P	VSS	USBP0N	VSS	USBRBIAS N	D
SIU0_DTR#	VSS	SIU0_DSR#	VSS	GPIO[39]	AC_SDOUT	AC_SDIN1	AC_SDIN2	AC_RST#	OC[3]#	VSS	VSS	VSS	USBRBIAS P	CLK48	E
VSS	SIU0_RTS#	VSS	VSS	VSS	AC_SYNC	VccSus1_5	AC_SDIN0	OC[1]#	OC[0]#	VSS	VSS	NC	NC	GPIO[38]	F
Vcc3_3	SIU1_RTS#	SERIRQ	NC	AC_BIT_CLK	Vcc1_5	VccSus3_3	V5REF_Sus	OC[2]#	VccSus1_5	NC	VSS	NC	NC	GPIO[42]	G
			VSS	Vcc1_5	VccSus3_3	Vcc1_5	VccSus1_5	VccSus1_5	NC	NC	GPIO[41]	VSS	GPIO[20]	GPIO[40]	H
							VCCPLL3	NC	VSS	VccSus1_5	Vcc3_3	GPIO[18]	SATALED#	NC	J
							VSS	Vcc3_3	GPIO[43]	GPIO[21]	VSS	SPKR	NC	NC	K
							GPIO[23]	VSS	GPIO[19]	VSS	CLK14	VSS	NC	NC	L
VSS	VSS	VSS	VSS				VSS	Vcc3_3	GPIO[32] / WDT_TOUT#	NC	VSS	Vcc3_3	HI11	NC	M
VSS	VSS	VSS	VSS					Vcc1_5	VccSus1_5	HI9	VSS	VccHI	HI10	HI8	N
VSS	VSS	VSS	VSS					VccHI	HIREF	VSWING	VccHI	VSS	HI1	HI0	P
VSS	VSS	VSS	VSS					VccHI	HI7	VccHI	VSS	VccHI	HI3	HI2	R
VSS	VSS	VSS	VSS					Vcc1_5	HICLK	VSS	HI6	HICOMP	HI_STB/ HI_STBS	HI_STB# / HI_STBF	T
VSS	VSS	VSS	VSS					VccSus1_5	IGNNE#	V_CPU_IO	VSS	V_CPU_IO	HI5	HI4	U
VSS	VSS	VSS	VSS				VSS	IRQ[15]	SDA[0]	VSS	RCIN#	VSS	CPUSLP#	SMI#	V
							SDD[6]	Vcc3_3	SIORDY / (SDRSTB/ SWDMARDY#)	SDIOR# / (SDWSTB/ PRDMARDY#)	V_CPU_IO	A20M#	INIT#	STPCLK#	W
							VSS	SDD[8]	SDDACK#	SDIOW# / (SDSTOP)	VRMPWRGD	VSS	INTR	NMI	Y
							VCCA	PDA[2]	SDD[4]	VSS	SDD[0]	SDCS3#	THRMTRIP#	FERR#	AA
			VSS	V5REF	PDD[9]	PDIOR# (/ PDWSTB / PRDMARDY#)	PDDACK#	PDIOW# (/PDSTOP)	SDD[10]	SDD[13]	Vcc3_3	SDD[2]	SDCS1#	A20GATE	AB
VSS	Vcc1_5	Vcc1_5	VSS	VCCRTC	PDD[7]	Vcc3_3	PDD[10]	IRQ[14]	VSS	Vcc3_3	SDD[11]	VSS	SDD[15]	SDA[2]	AC
SUS_STAT#	VCCPLL1	VSS	Vcc1_5	VBIAS	RTCX2	PDD[5]	PDD[3]	VSS	Vcc3_3	PDCS3#	SDD[3]	SDD[14]	SDDREQ	SDA[1]	AD
VccSus1_5	VCCPLL2	Vcc1_5	Vcc1_5	VSS	Vcc1_5	RTCX1	Vcc3_3	PDD[12]	PDD[14]	VSS	SDD[9]	VSS	SDD[12]	SDD[1]	AE
GPIO[11] / SMBALERT#	Vcc1_5	VSS	SATA[0]TXP	VSS	SATA[1]TXP	Vcc3_3	PWROK	PDD[8]	PDDREQ	PDD[0]	PIORDY (/ PDRSTB / PWDMARDY#)	PDCS1#	SDD[7]	SDD[5]	AF
VccSus1_5	Vcc1_5	VSS	SATA[0]TXN	VSS	SATA[1]TXN	VSS	RTCRST#	INTRUDER#	VSS	PDD[1]	VSS	PDA[1]	PDA[0]	VSS	AG
VSS	VSS	SATACLKP	Vcc1_5	SATA[0]RXP	VSS	SATA[1]RXP	VSS	SATARBIASP	PDD[6]	PDD[11]	PDD[2]	PDD[15]	VSS		AH
NC	VSS	SATACLKN	VSS	SATA[0]RXN	VSS	SATA[1]RXN	VSS	SATARBIASN	RSMRST#	PDD[4]	PDD[13]	VSS			AJ
15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	

Figure 38. Mechanical Drawing

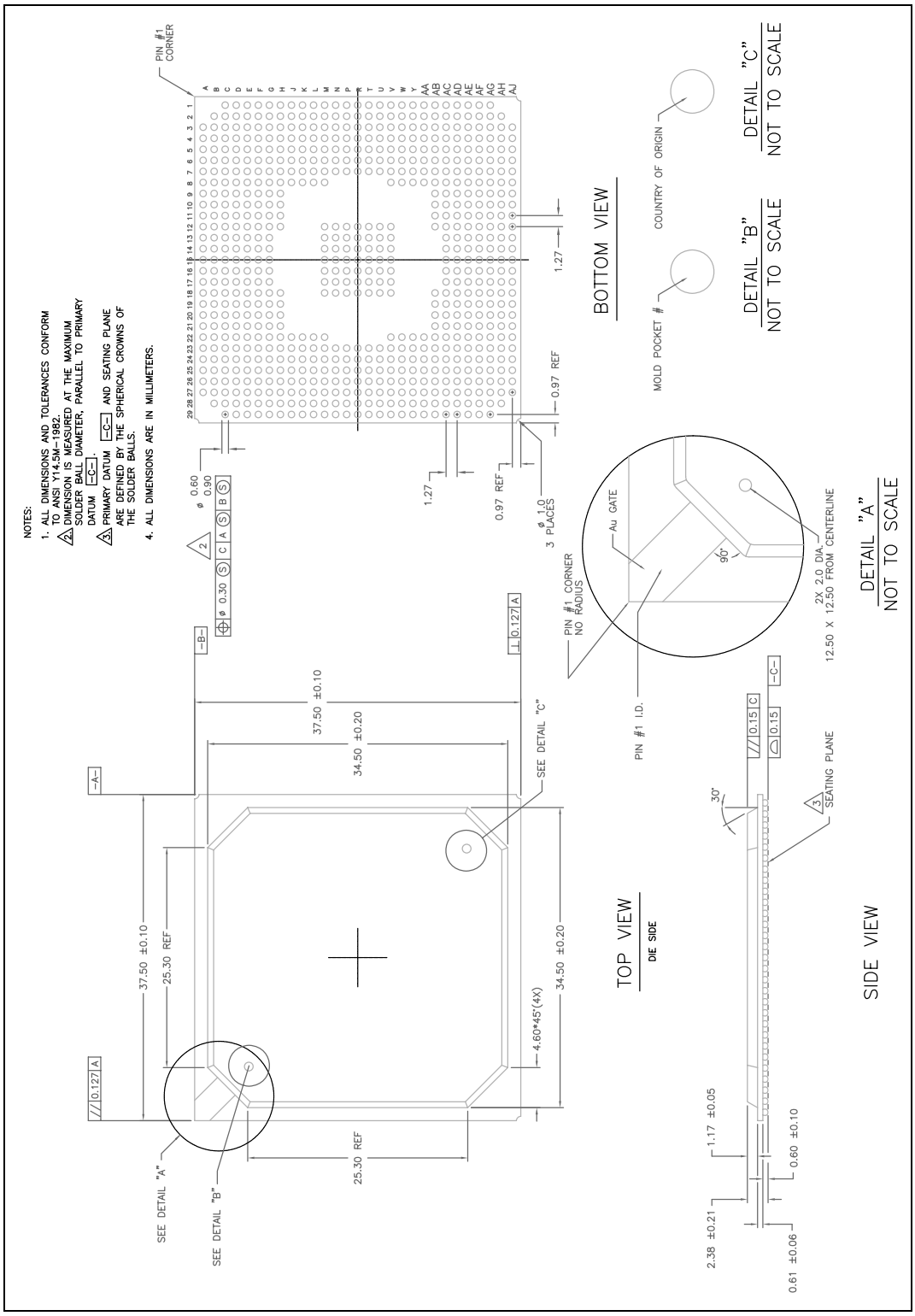


Table 706. Signal List (Alphabetical List)

Signal	Location
A20GATE	AB29
A20M#	W27
AC_BIT_CLK	G19
AC_RST#	E23
AC_SDIN0	F22
AC_SDIN1	E21
AC_SDIN2	E22
AC_SDOUT	E20
AC_SYNC	F20
AD[0]	H11
AD[1]	G8
AD[2]	C3
AD[3]	F5
AD[4]	F10
AD[5]	D4
AD[6]	D6
AD[7]	B3
AD[8]	H8
AD[9]	D5
AD[10]	E2
AD[11]	B4
AD[12]	G6
AD[13]	F6
AD[14]	D1
AD[15]	C7
AD[16]	A4
AD[17]	E1
AD[18]	B7
AD[19]	F2
AD[20]	A5
AD[21]	K8
AD[22]	B6
AD[23]	H7
AD[24]	E8
AD[25]	K7
AD[26]	E7
AD[27]	F1
AD[28]	H12
AD[29]	H4
AD[30]	G11
AD[31]	G2
C/BE[0]#	C2
C/BE[1]#	G7
C/BE[2]#	F4
C/BE[3]#	H5
CLK14	L26
CLK48	E29
CPUSLP#	V28
DEVSEL#	G5
FERR#	AA29

Table 706. Signal List (Alphabetical List)

Signal	Location
FRAME#	G10
GNT[0]#	E10
GNT[1]#	D9
GNT[2]#	C11
GNT[3]#	G13
GPIO[0] / PXREQ[2]#	J1
GPIO[1] / PXREQ[3]#	Y8
GPIO[2] / PIRQ[E]#	B10
GPIO[3] / PIRQ[F]#	A10
GPIO[4] / PIRQ[G]#	D10
GPIO[5] / PIRQ[H]#	A8
GPIO[6]	C21
GPIO[7]	B22
GPIO[8]	AE12
GPIO[11] / SMBALERT#	AF15
GPIO[12]	AF11
GPIO[13]	AC14
GPIO[16] / PXGNT[2]#	AC2
GPIO[17] / PXGNT[3]#	Y7
GPIO[18]	J27
GPIO[19]	L24
GPIO[20]	H28
GPIO[21]	K25
GPIO[23]	L22
GPIO[24]	AE14
GPIO[25]	AH9
GPIO[27]	AG10
GPIO[28]	AJ9
GPIO[32] / WDT _TOUT#	M24
GPIO[33] / PXIRQ[0]#	G1
GPIO[34] / PXIRQ[1]#	H1
GPIO[35] / PXIRQ[2]#	H2
GPIO[36] / PXIRQ[3]#	J3
GPIO[37]	D20
GPIO[38]	F29
GPIO[39]	E19
GPIO[40]	H29
GPIO[41]	H26
GPIO[42]	G29

Table 706. Signal List (Alphabetical List)

Signal	Location
GPIO[43]	K24
GPIO[56]	C22
GPIO[57]	D22
HI_STB#/HI_STBF	T29
HI_STB/HI_STBS	T28
HIREF	P24
HIHIHIHI0	P29
HI1	P28
HI2	R29
HI3	R28
HI4	U29
HI5	U28
HI6	T26
HI7	R24
HI8	N29
HI9	N25
HI10	N28
HI11	M28
HICLK	T24
HICOMP	T27
IGNNE#	U24
INIT#	W28
INTR	Y28
INTRUDER#	AG23
IRDY#	E3
IRQ[14]	AC23
IRQ[15]	V23
LAD[0]	C15
LAD[1]	B15
LAD[2]	A15
LAD[3]	A14
LDRQ[0]#	F14
LDRQ[1]#	G14
LFRAME#	B14
NC	B20
NC	B21
NC	B13
NC	B12
NC	A11
NC	A12
NC	A13
NC	A21
NC	A22
NC	AD13
NC	AJ14
NC	AJ15
NC	D19
NC	E13
NC	F13
NC	F27
NC	F28

Table 706. Signal List (Alphabetical List)

Signal	Location
NC	G18
NC	G25
NC	G27
NC	G28
NC	H24
NC	H25
NC	J23
NC	J29
NC	K28
NC	K29
NC	L28
NC	L29
NC	M25
NC	M29
NMI	Y29
OC[0]#	F24
OC[1]#	F23
OC[2]#	G23
OC[3]#	E24
PAR	G9
PCICLK	J5
PCIXSBRST#	AA7
PDA[0]	AG28
PDA[1]	AG27
PDA[2]	AA23
PDCS1#	AF27
PDCS3#	AD25
PDD[0]	AF25
PDD[1]	AG25
PDD[2]	AH26
PDD[3]	AD22
PDD[4]	AJ25
PDD[5]	AD21
PDD[6]	AH24
PDD[7]	AC20
PDD[8]	AF23
PDD[9]	AB20
PDD[10]	AC22
PDD[11]	AH25
PDD[12]	AE23
PDD[13]	AJ26
PDD[14]	AE24
PDD[15]	AH27
PDDACK#	AB22
PDDREQ	AF24
PDIOR# (/ PDWSTB / PRDMARDY#)	AB21
PDIOW# (/ PDSTOP)	AB23
PERR#	D2

Table 706. Signal List (Alphabetical List)

Signal	Location
PIORDY (/PDRSTB /PWDMARDY#)	AF26
PIRQ[A]#	E12
PIRQ[B]#	A6
PIRQ[C]#	A9
PIRQ[D]#	B8
PLOCK#	J6
PME#	AJ8
PWRBTN#	AJ10
PWROK	AF22
PXACK64#	AF4
PXAD[0]	AF2
PXAD[1]	W7
PXAD[2]	V7
PXAD[3]	Y4
PXAD[4]	AA2
PXAD[5]	W8
PXAD[6]	Y2
PXAD[7]	AA6
PXAD[8]	W1
PXAD[9]	W2
PXAD[10]	AA4
PXAD[11]	V2
PXAD[12]	V1
PXAD[13]	V4
PXAD[14]	Y1
PXAD[15]	T7
PXAD[16]	R1
PXAD[17]	P1
PXAD[18]	P2
PXAD[19]	P5
PXAD[20]	N2
PXAD[21]	P7
PXAD[22]	N1
PXAD[23]	N3
PXAD[24]	N4
PXAD[25]	M1
PXAD[26]	M2
PXAD[27]	L1
PXAD[28]	L3
PXAD[29]	M7
PXAD[30]	M5
PXAD[31]	K2
PXAD[32]	AF1
PXAD[33]	AE4
PXAD[34]	AE3
PXAD[35]	AE2
PXAD[36]	AD7
PXAD[37]	AD6
PXAD[38]	AD4
PXAD[39]	AC6

Table 706. Signal List (Alphabetical List)

Signal	Location
PXAD[40]	AD2
PXAD[41]	AD1
PXAD[42]	AB3
PXAD[43]	Y6
PXAD[44]	AB7
PXAD[45]	AB2
PXAD[46]	AC1
PXAD[47]	AB1
PXAD[48]	AD10
PXAD[49]	AE10
PXAD[50]	AC10
PXAD[51]	AE9
PXAD[52]	AJ7
PXAD[53]	AF9
PXAD[54]	AB9
PXAD[55]	AH7
PXAD[56]	AJ6
PXAD[57]	AH6
PXAD[58]	AJ5
PXAD[59]	AH5
PXAD[60]	AG5
PXAD[61]	AJ4
PXAD[62]	AH4
PXAD[63]	AH3
PXC/BE[0]#	AA1
PXC/BE[1]#	U2
PXC/BE[2]#	R2
PXC/BE[3]#	N6
PXC/BE[4]#	AE7
PXC/BE[5]#	AG2
PXC/BE[6]#	AF7
PXC/BE[7]#	AF6
PXDEVSEL#	T3
PXFRAME#	R6
PXGNT0#	K1
PXGNT1#	AB5
PXIRDY#	R4
PXM66EN	W4
PXPAR	U4
PXPAR64	AD8
PXPCICLK	AC5
PXPCIRST#	AH8
PXPCIXCAP	R7
PXPCLKI	AC4
PXPCLKO[0]	J2
PXPCLKO[1]	L8
PXPCLKO[2]	K6
PXPCLKO[3]	K4
PXPCLKO[4]	L5
PXPERR#	U6
PXPLOCK#	T1

Table 706. Signal List (Alphabetical List)

Signal	Location
PXRCOMP	AC12
PXREQ[0]#	L2
PXREQ[1]#	L6
PXREQ64#	AE5
PXSERR#	U1
PXSTOP#	T2
PXTRDY#	T5
RASERR#	AE1
RCIN#	V26
REQ[0]#	D8
REQ[1]#	G12
REQ[2]#	B11
REQ[3]#	D12
RI#	AH10
RSMRST#	AJ24
RTCRST#	AG22
RTCX1	AE21
RTCX2	AD20
SATA[0]RXN	AJ19
SATA[0]RXP	AH19
SATA[0]TXN	AG18
SATA[0]TXP	AF18
SATA[1]RXN	AJ21
SATA[1]RXP	AH21
SATA[1]TXN	AG20
SATA[1]TXP	AF20
SATACLKN	AJ17
SATACLKP	AH17
SATALED#	J28
SATARBIASN	AJ23
SATARBIASP	AH23
SDA[0]	V24
SDA[1]	AD29
SDA[2]	AC29
SDCS1#	AB28
SDCS3#	AA27
SDD[0]	AA26
SDD[1]	AE29
SDD[2]	AB27
SDD[3]	AD26
SDD[4]	AA24
SDD[5]	AF29
SDD[6]	W22
SDD[7]	AF28
SDD[8]	Y23
SDD[9]	AE26
SDD[10]	AB24
SDD[11]	AC26
SDD[12]	AE28
SDD[13]	AB25
SDD[14]	AD27

Table 706. Signal List (Alphabetical List)

Signal	Location
SDD[15]	AC28
SDDACK#	Y24
SDDREQ	AD28
SDIOR# / (SDWSTB/ PRDMARDY#)	W25
SDIOW# / (SDSTOP)	Y25
SERIRQ	G17
SERR#	E4
SIORDY / (SDRSTB/ SWDMARDY#)	W24
SIU0_CTS#	D16
SIU0_DCD#	B16
SIU0_DSR#	E17
SIU0_DTR#	E15
SIU0_RI#	A16
SIU0_RTS#	F16
SIU0_RXD	B17
SIU0_TXD	A17
SIU1_CTS#	A20
SIU1_DCD#	A19
SIU1_DSR#	B19
SIU1_DTR#	B18
SIU1_RI#	D17
SIU1_RTS#	G16
SIU1_RXD	D18
SIU1_TXD	C19
SLP_S3#	AG14
SLP_S4#	AD14
SLP_S5#	AJ11
SMBCLK	AH14
SMBDATA	AJ13
SMI#	V29
SMLINK[0]	AH13
SMLINK[1]	AH12
SPKR	K27
STOP#	B5
STPCLK#	W29
SUS_STAT#	AD15
SUSCLK	AJ12
SYSRESET#	AH11
THRM#	D14
THRMTRIP#	AA28
TRDY#	F9
UART_CLK	A18
USBP0N	D27
USBP0P	C27
USBP1N	C25
USBP1P	D25
USBP2N	B26

Table 706. Signal List (Alphabetical List)

Signal	Location
USBP2P	A26
USBP3N	B24
USBP3P	A24
USBRBIASN	D29
USBRBIASP	E28
V_CPU_IO	U25
V_CPU_IO	U27
V_CPU_IO	W26
V5REF	AB19
V5REF	E11
V5REF_Sus	G22
VBIAS	AD19
Vcc1_5	AB10
Vcc1_5	AB8
Vcc1_5	AC13
Vcc1_5	AC16
Vcc1_5	AC17
Vcc1_5	AD18
Vcc1_5	AE17
Vcc1_5	AE18
Vcc1_5	AE20
Vcc1_5	AF16
Vcc1_5	AG16
Vcc1_5	AH18
Vcc1_5	F12
Vcc1_5	G20
Vcc1_5	H19
Vcc1_5	H21
Vcc1_5	J8
Vcc1_5	N23
Vcc1_5	N7
Vcc1_5	T23
Vcc1_5	V8
Vcc3_3	AA3
Vcc3_3	AA5
Vcc3_3	AB11
Vcc3_3	AB26
Vcc3_3	AB6
Vcc3_3	AC21
Vcc3_3	AC25
Vcc3_3	AC9
Vcc3_3	AD24
Vcc3_3	AD3
Vcc3_3	AE11
Vcc3_3	AE22
Vcc3_3	AE8
Vcc3_3	AF21
Vcc3_3	AF5
Vcc3_3	AG3
Vcc3_3	AG7
Vcc3_3	C13

Table 706. Signal List (Alphabetical List)

Signal	Location
Vcc3_3	C17
Vcc3_3	C5
Vcc3_3	C9
Vcc3_3	D15
Vcc3_3	D3
Vcc3_3	E6
Vcc3_3	E9
Vcc3_3	G15
Vcc3_3	G3
Vcc3_3	H3
Vcc3_3	H9
Vcc3_3	J26
Vcc3_3	J7
Vcc3_3	K23
Vcc3_3	K3
Vcc3_3	K5
Vcc3_3	M23
Vcc3_3	M27
Vcc3_3	M4
Vcc3_3	M8
Vcc3_3	N5
Vcc3_3	P4
Vcc3_3	R3
Vcc3_3	R5
Vcc3_3	U5
Vcc3_3	U7
Vcc3_3	W23
Vcc3_3	W3
Vcc3_3	W5
Vcc3_3	W6
VCCA	AA22
VccHI	N27
VccHI	P23
VccHI	P26
VccHI	R23
VccHI	R25
VccHI	R27
VCCPLL0	AA8
VCCPLL1	AD16
VCCPLL2	AE16
VCCPLL3	J22
VCCREF	AC8
VCCREF	V6
VCCRTC	AC19
VccSus1_5	AE15
VccSus1_5	AG15
VccSus1_5	D21
VccSus1_5	F21
VccSus1_5	G24
VccSus1_5	H22
VccSus1_5	H23

Table 706. Signal List (Alphabetical List)

Signal	Location
VccSus1_5	J25
VccSus1_5	N24
VccSus1_5	U23
VccSus3_3	B23
VccSus3_3	A23
VccSus3_3	AF13
VccSus3_3	AG12
VccSus3_3	C23
VccSus3_3	D23
VccSus3_3	D24
VccSus3_3	G21
VccSus3_3	H20
VRMPWRGD	Y26
VSS	B2
VSS	B25
VSS	B27
VSS	B28
VSS	B9
VSS	A25
VSS	A27
VSS	A3
VSS	A7
VSS	AA25
VSS	AB12
VSS	AB18
VSS	AB4
VSS	AC11
VSS	AC15
VSS	AC18
VSS	AC24
VSS	AC27
VSS	AC3
VSS	AC7
VSS	AD11
VSS	AD12
VSS	AD17
VSS	AD23
VSS	AD5
VSS	AD9
VSS	AE13
VSS	AE19
VSS	AE25
VSS	AE27
VSS	AE6
VSS	AF10
VSS	AF12
VSS	AF14
VSS	AF17
VSS	AF19
VSS	AF3
VSS	AF8

Table 706. Signal List (Alphabetical List)

Signal	Location
VSS	AG1
VSS	AG11
VSS	AG13
VSS	AG17
VSS	AG19
VSS	AG21
VSS	AG24
VSS	AG26
VSS	AG29
VSS	AG4
VSS	AG6
VSS	AG8
VSS	AG9
VSS	AH15
VSS	AH16
VSS	AH2
VSS	AH20
VSS	AH22
VSS	AH28
VSS	AJ16
VSS	AJ18
VSS	AJ20
VSS	AJ22
VSS	AJ27
VSS	AJ3
VSS	C1
VSS	C10
VSS	C12
VSS	C14
VSS	C16
VSS	C18
VSS	C20
VSS	C24
VSS	C26
VSS	C28
VSS	C29
VSS	C4
VSS	C6
VSS	C8
VSS	D11
VSS	D13
VSS	D26
VSS	D28
VSS	D7
VSS	E14
VSS	E16
VSS	E18
VSS	E25
VSS	E26
VSS	E27
VSS	E5

Table 706. Signal List (Alphabetical List)

Signal	Location
VSS	F11
VSS	F15
VSS	F17
VSS	F18
VSS	F19
VSS	F25
VSS	F26
VSS	F3
VSS	F7
VSS	F8
VSS	G26
VSS	G4
VSS	H10
VSS	H18
VSS	H27
VSS	H6
VSS	J24
VSS	J4
VSS	K22
VSS	K26
VSS	L23
VSS	L25
VSS	L27
VSS	L4
VSS	L7
VSS	M12
VSS	M13
VSS	M14
VSS	M15
VSS	M16
VSS	M17
VSS	M18
VSS	M22
VSS	M26
VSS	M3
VSS	M6
VSS	N12
VSS	N13
VSS	N14
VSS	N15
VSS	N16
VSS	N17
VSS	N18
VSS	N26
VSS	P12
VSS	P13
VSS	P14
VSS	P15
VSS	P16
VSS	P17
VSS	P18

Table 706. Signal List (Alphabetical List)

Signal	Location
VSS	P27
VSS	P3
VSS	P6
VSS	R12
VSS	R13
VSS	R14
VSS	R15
VSS	R16
VSS	R17
VSS	R18
VSS	R26
VSS	T12
VSS	T13
VSS	T14
VSS	T15
VSS	T16
VSS	T17
VSS	T18
VSS	T25
VSS	T4
VSS	T6
VSS	U12
VSS	U13
VSS	U14
VSS	U15
VSS	U16
VSS	U17
VSS	U18
VSS	U26
VSS	U3
VSS	V12
VSS	V13
VSS	V14
VSS	V15
VSS	V16
VSS	V17
VSS	V18
VSS	V22
VSS	V25
VSS	V27
VSS	V3
VSS	V5
VSS	Y22
VSS	Y27
VSS	Y3
VSS	Y5
VSWING	P25

Table 707. Signal List (by Location)

Location	Signal
A3	VSS
A4	AD[16]
A5	AD[20]
A6	PIRQ[B]#
A7	VSS
A8	GPIO[5] / PIRQ[H]#
A9	PIRQ[C]#
A10	GPIO[3] / PIRQ[F]#
A11	NC
A12	NC
A13	NC
A14	LAD[3]
A15	LAD[2]
A16	SIU0_RI#
A17	SIU0_TXD
A18	UART_CLK
A19	SIU1_DCD#
A20	SIU1_CTS#
A21	NC
A22	NC
A23	VccSus3_3
A24	USBP3P
A25	VSS
A26	USBP2P
A27	VSS
AA1	PXC/BE[0]#
AA2	PXAD[4]
AA3	Vcc3_3
AA4	PXAD[10]
AA5	Vcc3_3
AA6	PXAD[7]
AA7	PCIXSBRST#
AA8	VCCPLL0
AA22	VCCA
AA23	PDA[2]
AA24	SDD[4]
AA25	VSS
AA26	SDD[0]
AA27	SDCS3#
AA28	THRMTRIP#
AA29	FERR#
AB1	PXAD[47]
AB2	PXAD[45]
AB3	PXAD[42]
AB4	VSS
AB5	PXGNT1#
AB6	Vcc3_3
AB7	PXAD[44]
AB8	Vcc1_5
AB9	PXAD[54]

Table 707. Signal List (by Location)

Location	Signal
AB10	Vcc1_5
AB11	Vcc3_3
AB12	VSS
AB18	VSS
AB19	V5REF
AB20	PDD[9]
AB21	PDIOR# (/ PDWSTB / PRDMARDY#)
AB22	PDDACK#
AB23	PDIOW# (/ PDSTOP)
AB24	SDD[10]
AB25	SDD[13]
AB26	Vcc3_3
AB27	SDD[2]
AB28	SDCS1#
AB29	A20GATE
AC1	PXAD[46]
AC2	GPIO[16] / PXGNT[2]#
AC3	VSS
AC4	PXPCLKI
AC5	PXPCICLK
AC6	PXAD[39]
AC7	VSS
AC8	VCCREF
AC9	Vcc3_3
AC10	PXAD[50]
AC11	VSS
AC12	PXRCOMP
AC13	Vcc1_5
AC14	GPIO[13]
AC15	VSS
AC16	Vcc1_5
AC17	Vcc1_5
AC18	VSS
AC19	VCCRTC
AC20	PDD[7]
AC21	Vcc3_3
AC22	PDD[10]
AC23	IRQ[14]
AC24	VSS
AC25	Vcc3_3
AC26	SDD[11]
AC27	VSS
AC28	SDD[15]
AC29	SDA[2]
AD1	PXAD[41]
AD2	PXAD[40]
AD3	Vcc3_3
AD4	PXAD[38]
AD5	VSS

Table 707. Signal List (by Location)

Location	Signal
AD6	PXAD[37]
AD7	PXAD[36]
AD8	PXPAR64
AD9	VSS
AD10	PXAD[48]
AD11	VSS
AD12	VSS
AD13	NC
AD14	SLP_S4#
AD15	SUS_STAT#
AD16	VCCPLL1
AD17	VSS
AD18	Vcc1_5
AD19	VBIAS
AD20	RTCX2
AD21	PDD[5]
AD22	PDD[3]
AD23	VSS
AD24	Vcc3_3
AD25	PDCS3#
AD26	SDD[3]
AD27	SDD[14]
AD28	SDDREQ
AD29	SDA[1]
AE1	RASERR#
AE2	PXAD[35]
AE3	PXAD[34]
AE4	PXAD[33]
AE5	PXREQ64#
AE6	VSS
AE7	PXC/BE[4]#
AE8	Vcc3_3
AE9	PXAD[51]
AE10	PXAD[49]
AE11	Vcc3_3
AE12	GPIO[8]
AE13	VSS
AE14	GPIO[24]
AE15	VccSus1_5
AE16	VCCPLL2
AE17	Vcc1_5
AE18	Vcc1_5
AE19	VSS
AE20	Vcc1_5
AE21	RTCX1
AE22	Vcc3_3
AE23	PDD[12]
AE24	PDD[14]
AE25	VSS
AE26	SDD[9]
AE27	VSS
AE28	SDD[12]

Table 707. Signal List (by Location)

Location	Signal
AE29	SDD[1]
AF1	PXAD[32]
AF2	PXAD[0]
AF3	VSS
AF4	PXACK64#
AF5	Vcc3_3
AF6	PXC/BE[7]#
AF7	PXC/BE[6]#
AF8	VSS
AF9	PXAD[53]
AF10	VSS
AF11	GPIO[12]
AF12	VSS
AF13	VccSus3_3
AF14	VSS
AF15	GPIO[11] / SMBALERT#
AF16	Vcc1_5
AF17	VSS
AF18	SATA[0]TXP
AF19	VSS
AF20	SATA[1]TXP
AF21	Vcc3_3
AF22	PWROK
AF23	PDD[8]
AF24	PDDREQ
AF25	PDD[0]
AF26	PIORDY (/ PDRSTB / PWDMARDY#)
AF27	PDCS1#
AF28	SDD[7]
AF29	SDD[5]
AG1	VSS
AG2	PXC/BE[5]#
AG3	Vcc3_3
AG4	VSS
AG5	PXAD[60]
AG6	VSS
AG7	Vcc3_3
AG8	VSS
AG9	VSS
AG10	GPIO[27]
AG11	VSS
AG12	VccSus3_3
AG13	VSS
AG14	SLP_S3#
AG15	VccSus1_5
AG16	Vcc1_5
AG17	VSS
AG18	SATA[0]TXN
AG19	VSS

Table 707. Signal List (by Location)

Location	Signal
AG20	SATA[1]TXN
AG21	VSS
AG22	RTCRST#
AG23	INTRUDER#
AG24	VSS
AG25	PDD[1]
AG26	VSS
AG27	PDA[1]
AG28	PDA[0]
AG29	VSS
AH2	VSS
AH3	PXAD[63]
AH4	PXAD[62]
AH5	PXAD[59]
AH6	PXAD[57]
AH7	PXAD[55]
AH8	PXPCIRST#
AH9	GPIO[25]
AH10	RI#
AH11	SYSRESET#
AH12	SMLINK[1]
AH13	SMLINK[0]
AH14	SMBCLK
AH15	VSS
AH16	VSS
AH17	SATACLKP
AH18	Vcc1_5
AH19	SATA[0]RXP
AH20	VSS
AH21	SATA[1]RXP
AH22	VSS
AH23	SATARBIASP
AH24	PDD[6]
AH25	PDD[11]
AH26	PDD[2]
AH27	PDD[15]
AH28	VSS
AJ3	VSS
AJ4	PXAD[61]
AJ5	PXAD[58]
AJ6	PXAD[56]
AJ7	PXAD[52]
AJ8	PME#
AJ9	GPIO[28]
AJ10	PWRBTN#
AJ11	SLP_S5#
AJ12	SUSCLK
AJ13	SMBDATA
AJ14	NC
AJ15	NC
AJ16	VSS
AJ17	SATACLKN

Table 707. Signal List (by Location)

Location	Signal
AJ18	VSS
AJ19	SATA[0]RXN
AJ20	VSS
AJ21	SATA[1]RXN
AJ22	VSS
AJ23	SATARBIASN
AJ24	RSMRST#
AJ25	PDD[4]
AJ26	PDD[13]
AJ27	VSS
B2	VSS
B3	AD[7]
B4	AD[11]
B5	STOP#
B6	AD[22]
B7	AD[18]
B8	PIRQ[D]#
B9	VSS
B10	GPIO[2] / PIRQ[E]#
B11	REQ[2]#
B12	NC
B13	NC
B14	LFRAME#
B15	LAD[1]
B16	SIU0_DCD#
B17	SIU0_RXD
B18	SIU1_DTR#
B19	SIU1_DSR#
B20	NC
B21	NC
B22	GPIO[7]
B23	VccSus3_3
B24	USBP3N
B25	VSS
B26	USBP2N
B27	VSS
B28	VSS
C1	VSS
C2	C/BE[0]#
C3	AD[2]
C4	VSS
C5	Vcc3_3
C6	VSS
C7	AD[15]
C8	VSS
C9	Vcc3_3
C10	VSS
C11	GNT[2]#
C12	VSS
C13	Vcc3_3
C14	VSS

Table 707. Signal List (by Location)

Location	Signal
C15	LAD[0]
C16	VSS
C17	Vcc3_3
C18	VSS
C19	SIU1_TXD
C20	VSS
C21	GPIO[6]
C22	GPIO[56]
C23	VccSus3_3
C24	VSS
C25	USBP1N
C26	VSS
C27	USBP0P
C28	VSS
C29	VSS
D1	AD[14]
D2	PERR#
D3	Vcc3_3
D4	AD[5]
D5	AD[9]
D6	AD[6]
D7	VSS
D8	REQ[0]#
D9	GNT[1]#
D10	GPIO[4] / PIRQ[G]#
D11	VSS
D12	REQ[3]#
D13	VSS
D14	THRM#
D15	Vcc3_3
D16	SIU0_CTS#
D17	SIU1_RI#
D18	SIU1_RXD
D19	NC
D20	GPIO[37]
D21	VccSus1_5
D22	GPIO[57]
D23	VccSus3_3
D24	VccSus3_3
D25	USBP1P
D26	VSS
D27	USBP0N
D28	VSS
D29	USBRBIASN
E1	AD[17]
E2	AD[10]
E3	IRDY#
E4	SERR#
E5	VSS
E6	Vcc3_3
E7	AD[26]

Table 707. Signal List (by Location)

Location	Signal
E8	AD[24]
E9	Vcc3_3
E10	GNT[0]#
E11	V5REF
E12	PIRQ[A]#
E13	NC
E14	VSS
E15	SIU0_DTR#
E16	VSS
E17	SIU0_DSR#
E18	VSS
E19	GPIO[39]
E20	AC_SDOUT
E21	AC_SDIN1
E22	AC_SDIN2
E23	AC_RST#
E24	OC[3]#
E25	VSS
E26	VSS
E27	VSS
E28	USBRBIASP
E29	CLK48
F1	AD[27]
F2	AD[19]
F3	VSS
F4	C/BE[2]#
F5	AD[3]
F6	AD[13]
F7	VSS
F8	VSS
F9	TRDY#
F10	AD[4]
F11	VSS
F12	Vcc1_5
F13	NC
F14	LDRQ[0]#
F15	VSS
F16	SIU0_RTS#
F17	VSS
F18	VSS
F19	VSS
F20	AC_SYNC
F21	VccSus1_5
F22	AC_SDIN0
F23	OC[1]#
F24	OC[0]#
F25	VSS
F26	VSS
F27	NC
F28	NC
F29	GPIO[38]

Table 707. Signal List (by Location)

Location	Signal
G1	GPIO[33] / PXIRQ[0]#
G2	AD[31]
G3	Vcc3_3
G4	VSS
G5	DEVSEL#
G6	AD[12]
G7	C/BE[1]#
G8	AD[1]
G9	PAR
G10	FRAME#
G11	AD[30]
G12	REQ[1]#
G13	GNT[3]#
G14	LDRQ[1]#
G15	Vcc3_3
G16	SIU1_RTS#
G17	SERIRQ
G18	NC
G19	AC_BIT_CLK
G20	Vcc1_5
G21	VccSus3_3
G22	V5REF_Sus
G23	OC[2]#
G24	VccSus1_5
G25	NC
G26	VSS
G27	NC
G28	NC
G29	GPIO[42]
H1	GPIO[34] / PXIRQ[1]#
H2	GPIO[35] / PXIRQ[2]#
H3	Vcc3_3
H4	AD[29]
H5	C/BE[3]#
H6	VSS
H7	AD[23]
H8	AD[8]
H9	Vcc3_3
H10	VSS
H11	AD[0]
H12	AD[28]
H18	VSS
H19	Vcc1_5
H20	VccSus3_3
H21	Vcc1_5
H22	VccSus1_5
H23	VccSus1_5
H24	NC
H25	NC

Table 707. Signal List (by Location)

Location	Signal
H26	GPIO[41]
H27	VSS
H28	GPIO[20]
H29	GPIO[40]
J1	GPIO[0] / PXREQ[2]#
J2	PXPCLKO[0]
J3	GPIO[36] / PXIRQ[3]#
J4	VSS
J5	PCICLK
J6	PLOCK#
J7	Vcc3_3
J8	Vcc1_5
J22	VCCPLL3
J23	NC
J24	VSS
J25	VccSus1_5
J26	Vcc3_3
J27	GPIO[18]
J28	SATALED#
J29	NC
K1	PXGNT0#
K2	PXAD[31]
K3	Vcc3_3
K4	PXPCLKO[3]
K5	Vcc3_3
K6	PXPCLKO[2]
K7	AD[25]
K8	AD[21]
K22	VSS
K23	Vcc3_3
K24	GPIO[43]
K25	GPIO[21]
K26	VSS
K27	SPKR
K28	NC
K29	NC
L1	PXAD[27]
L2	PXREQ[0]#
L3	PXAD[28]
L4	VSS
L5	PXPCLKO[4]
L6	PXREQ[1]#
L7	VSS
L8	PXPCLKO[1]
L22	GPIO[23]
L23	VSS
L24	GPIO[19]
L25	VSS
L26	CLK14
L27	VSS

Table 707. Signal List (by Location)

Location	Signal
L28	NC
L29	NC
M1	PXAD[25]
M2	PXAD[26]
M3	VSS
M4	Vcc3_3
M5	PXAD[30]
M6	VSS
M7	PXAD[29]
M8	Vcc3_3
M12	VSS
M13	VSS
M14	VSS
M15	VSS
M16	VSS
M17	VSS
M18	VSS
M22	VSS
M23	Vcc3_3
M24	GPIO[32] / WDT_TOUT#
M25	NC
M26	VSS
M27	Vcc3_3
M28	HI11
M29	NC
N1	PXAD[22]
N2	PXAD[20]
N3	PXAD[23]
N4	PXAD[24]
N5	Vcc3_3
N6	PXC/BE[3]#
N7	Vcc1_5
N12	VSS
N13	VSS
N14	VSS
N15	VSS
N16	VSS
N17	VSS
N18	VSS
N23	Vcc1_5
N24	VccSus1_5
N25	HI9
N26	VSS
N27	VccHI
N28	HI10
N29	HI8
P1	PXAD[17]
P2	PXAD[18]
P3	VSS
P4	Vcc3_3
P5	PXAD[19]

Table 707. Signal List (by Location)

Location	Signal
P6	VSS
P7	PXAD[21]
P12	VSS
P13	VSS
P14	VSS
P15	VSS
P16	VSS
P17	VSS
P18	VSS
P23	VccHI
P24	HIREF
P25	VSWING
P26	VccHI
P27	VSS
P28	HI1
P29	HI0
R1	PXAD[16]
R2	PXC/BE[2]#
R3	Vcc3_3
R4	PXIRDY#
R5	Vcc3_3
R6	PXFRAME#
R7	PXPCIXCAP
R12	VSS
R13	VSS
R14	VSS
R15	VSS
R16	VSS
R17	VSS
R18	VSS
R23	VccHI
R24	HI7
R25	VccHI
R26	VSS
R27	VccHI
R28	HI3
R29	HI2
T1	PXPLOCK#
T2	PXSTOP#
T3	PXDEVSEL#
T4	VSS
T5	PXTRDY#
T6	VSS
T7	PXAD[15]
T12	VSS
T13	VSS
T14	VSS
T15	VSS
T16	VSS
T17	VSS
T18	VSS
T23	Vcc1_5

Table 707. Signal List (by Location)

Location	Signal
T24	HICLK
T25	VSS
T26	HI6
T27	HICOMP
T28	HI_STB/ HI_STBS
T29	HI_STB#/ HI_STBF
U1	PXSERR#
U2	PXC/BE[1]#
U3	VSS
U4	PXPAR
U5	Vcc3_3
U6	PXPERR#
U7	Vcc3_3
U12	VSS
U13	VSS
U14	VSS
U15	VSS
U16	VSS
U17	VSS
U18	VSS
U23	VccSus1_5
U24	IGNNE#
U25	V_CPU_IO
U26	VSS
U27	V_CPU_IO
U28	HI5
U29	HI4
V1	PXAD[12]
V2	PXAD[11]
V3	VSS
V4	PXAD[13]
V5	VSS
V6	VCCREF
V7	PXAD[2]
V8	Vcc1_5
V12	VSS
V13	VSS
V14	VSS
V15	VSS
V16	VSS
V17	VSS
V18	VSS
V22	VSS
V23	IRQ[15]
V24	SDA[0]
V25	VSS
V26	RCIN#
V27	VSS
V28	CPUSLP#
V29	SMI#

Table 707. Signal List (by Location)

Location	Signal
W1	PXAD[8]
W2	PXAD[9]
W3	Vcc3_3
W4	PXM66EN
W5	Vcc3_3
W6	Vcc3_3
W7	PXAD[1]
W8	PXAD[5]
W22	SDD[6]
W23	Vcc3_3
W24	SIORDY / (SDRSTB/ SWDMARDY#)
W25	SDIOR# / (SDWSTB/ PRDMARDY#)
W26	V_CPU_IO
W27	A20M#
W28	INIT#
W29	STPCLK#
Y1	PXAD[14]
Y2	PXAD[6]
Y3	VSS
Y4	PXAD[3]
Y5	VSS
Y6	PXAD[43]
Y7	GPIO[17] / PXGNT[3]#
Y8	GPIO[1] / PXREQ[3]#
Y22	VSS
Y23	SDD[8]
Y24	SDDACK#
Y25	SDIOW# / (SDSTOP)
Y26	VRMPWRGD
Y27	VSS
Y28	INTR
Y29	NMI

Electrical Characteristics 22

This chapter provides the absolute maximum ratings, DC characteristics, AC characteristics and AC timing diagrams for the Intel® 6300ESB ICH component.

22.1 Absolute Maximum Ratings

Voltage on any 3.3 V pin with respect to Ground = –0.5 to Vcc3_3 +0.5 V
Voltage on any 5 V tolerant pin with respect to Ground (V5REF = 5 V) = –0.5 to V5REF + 0.5 V
1.5 V supply voltage with respect to Vss = –0.5 to +2.1 V
3.3 V supply voltage with respect to Vss = –0.5 to +4.6 V
5.0 V supply voltage (V5REF) with respect to Vss = –0.5 to +5.5 V

Warning: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. See Section 22.2, "Functional Operating Range" for the Functional Operating Range of the Intel® 6300ESB ICH.

22.2 Functional Operating Range

All of the AC and DC Characteristics specified in this document assume that the Intel® 6300ESB ICH component is operating within the Functional Operating Range given in this section. Operation outside of the Functional Operating Range is not recommended, and extended exposure outside of the Functional Operating Range may affect component reliability.

- 1.5 V supply voltage (Vcc1_5, VccHI, VccSus1_5) with respect to Vss = 1.425 V to 1.575 V
- 3.3 V supply voltage (Vcc3_3, VccSus3_3) with respect to Vss = 3.135 V to 3.465 V
- 5 V supply voltage (V5REF, V5REF_Sus) with respect to Vss = 4.75 V to 5.25 V
- V_CPU_IO voltage with respect to Vss = 0.8 V—1.75 V
- VCCRTC voltage with respect to Vss = 2.0 V to 3.6 V
- Case temperature under Bias = 0° C to +105° C

Note: A non-condensing environment is required to maintain RTC accuracy.

22.3 DC Characteristics

Table 708. DC Current Characteristics (Preliminary)

Power Plane	Maximum Power Consumption			
Symbol	**S0**	**S1**	**S3/S4/S5**	**G3**
Vcc1_5 Core	1184 mA	573 mA	N/A	N/A
Vcc3_3 I/O	875 mA	1.3 mA	N/A	N/A
VccSus1_5	75 mA	45 mA	15.6 mA	N/A
VccSus3_3	142 mA	2.1 mA	2.3 mA	N/A
VccHI	99 mA	99 mA	N/A	N/A
V5REF	10 µA	10 µA	N/A	N/A
V5REF_SUS	10 µA	10 µA	10 µA	N/A
VCCREF (3.3V)	150 µA	150 µA	N/A	N/A
V_CPU_IO	2.5 mA	2.5 mA	N/A	N/A
VCCRTC	N/A	N/A	N/A	8.5µA(1)

1. Icc(RTC) data is taken with Vcc(RTC) at 3.0 V while the system is in a mechanical off (G3) state at room temperature (25° C).

Table 709. DC Characteristic Input Signal Association (Sheet 1 of 2)

Symbol	Associated Signals
V_{IH0}/V_{IL0}	**PCI-X Signals:** PXAD[63:0], PXC/BE[7:0]#, PXDEVSEL#, PXFRAME#, PXIRDY#, PXTRDY#, PXSTOP#, PXPAR, PXPERR#, PXPLOCK#, PXSERR#, PXREQ[1:0]#,PXREQ[2]#/GPIO[0], PXREQ[3]#/GPIO[1], PXRCOMP, PXPAR64, PXREQ64#, PXACK64#, PXM66EN, PXPCIXCAP, PXIRQ[3:0]/GPIO[36:33] **Clock Signals**: PXPCLKI, PXPCICLK
V_{IH1}/V_{IL1} (5V Tolerant)	**PCI Signals:** AD[31:0], C/BE[3:0]#, DEVSEL#, FRAME#, IRDY#, TRDY#, STOP#, PAR, PERR#, PLOCK#, SERR#, REQ[3:0]#
V_{IH2}/V_{IL2} (5V Tolerant)	**Interrupt Signals:** IRQ[15:14], PIRQ[D:A]#, PIRQ[H:E]#/GPIO[5:2] **Legacy Signals:** RCIN#, A20GATE
V_{IH3}/V_{IL3}	**Clock Signals**: CLK48, CLK14 **Interrupt Signals:** SERIRQ **Power Management Signals:** PME#, PWRBTN#, RI#, SYS_RESET#, THRM# **GPIO Signals:** GPIO[43:37,31, 28:27, 25:24, 13:12, 8:6]
V_{IH4}/V_{IL4}	**Clock Signals:** PCICLK **LPC/FWH Signals:** LDRQ[1:0]#, LAD[3:0]/FWH[3:0], LFRAME# **SIU Signals:** SIU0_CTS#, SIU0_DCD#, SIU0_DSR#, SIU0_RI#, SIU0_RXD, SIU1_CTS#; SIU1_DCD#, SIU1_DSR#, SIU1_RI#, SIU1_RXD, UART_CLK
V_{IH5}/V_{IL5}	**SMBus Signals:** SMBCLK, SMBDATA **System Management Signals:** INTRUDER#, SMLINK[1:0], SMBALERT#/GPIO[11] **Power Management Signals:** RSMRST#, RTCRST#, PWROK
V_{IL6}/V_{IH6}	**CPU Signals:** FERR#, THRMTRIP#
V_{IL7}/V_{IH7}	**Hub Interface Signals**: HI[11:0], HI_STBS, HI_STBF
V_{IL8}/V_{IH8}	**Real Time Clock Signals:** RTCX1, RTCX2

Table 709. DC Characteristic Input Signal Association (Sheet 2 of 2)

Symbol	Associated Signals
V_{IL9}/V_{IH9}	**SATA Signals:** SATA[1:0]RX[P,N]
V_{IL10}/V_{IH10} (5V Tolerant)	**USB Signals:** OC[3:0]#
V_{IL11}/V_{IH11}	**AC'97 Signals:** AC_BITCLK, AC_SDIN[2:0]
V_{IL12}/V_{IH12}	**Clock Signals:** SATACLKP, SATACLKN (CLK100P, CLK100N)
V+/V-/VHYS/ VTHRAVG/VRING (5V Tolerant)	**IDE Signals:** PDD[15:0], SDD[15:0], PDDREQ, PIORDY, SDDREQ, SIORDY For Ultra DMA Mode 4 and lower these signals, follow the DC characteristics for V_{IH2}/V_{IL2}.
V_{DI} / V_{CM} / V_{SE}	**USB Signals:** USBP[3:0][P,N] (Low-speed and Full-speed)
V_{HSSQ} / V_{HSDSC} / V_{HSCM}	**USB Signals:** USBP[3:0][P,N] (in High-speed Mode)

Table 710. DC Input Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
V_{IL0}	Input Low Voltage	-0.5	0.35Vcc3_3	V	
V_{IH0}	Input High Voltage	0.5Vcc3_3	Vcc3_3 + 0.5	V	
V_{IL1}	Input Low Voltage	-0.5	0.3Vcc3_3	V	
V_{IH1}	Input High Voltage	0.5Vcc3_3	V5REF + 0.5	V	
V_{IL2}	Input Low Voltage	-0.5	0.8	V	
V_{IH2}	Input High Voltage	2.0	V5REF + 0.5	V	
V_{IL3}	Input Low Voltage	-0.5	0.8	V	
V_{IH3}	Input High Voltage	2.0	Vcc3_3 + 0.5	V	
V_{IL4}	Input Low Voltage	-0.5	0.3Vcc3_3	V	
V_{IH4}	Input High Voltage	0.5Vcc3_3	Vcc3_3 + 0.5	V	
V_{IL5}	Input Low Voltage	-0.5	0.8	V	
V_{IH5}	Input High Voltage	2.1	VccSus3_3 + 0.5	V	
V_{IL6}	Input Low Voltage	-0.15	0.58(V_CPU_IO)	V	
V_{IH6}	Input High Voltage	0.73(V_CPU_IO)	V_CPU_IO	V	
V_{IL7}	Input Low Voltage	-0.3	HIREF - 0.10	V	Note 7
V_{IH7}	Input High Voltage	HIREF + 0.10	1.2	V	Note 7
V_{IL8}	Input Low Voltage	-0.5	0.10	V	
V_{IH8}	Input High Voltage	0.40	1.0	V	

NOTES:
1. Applies to Ultra DMA Modes greater than Ultra DMA Mode 4.
2. This is an AC Characteristic that represents transient values for these signals.
3. V_{DI} = | USBPx[P] - USBPx[N].
4. Includes V_{DI} range.
5. Applies to High-speed USB 2.0.
6. SATA Vdiff,rx is measured at the SATA connector on the receive side.
7. When probed at the receiver pin of the ICH for data/strobe, the waveform may show a "knee" due to package parasitics. Simulation verifies that this "knee" represents no risk, since a clean waveform is present at the ICH ball input due to internal receiver termination.

Table 710. DC Input Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
V_{IL9}	Differential Input Low Voltage	325		mVp-p	Note 6
V_{IH9}	Differential Input High Voltage		600	mVp-p	Note 6
V_{IL10}	Input Low Voltage	-0.5	0.8	V	
V_{IH10}	Input High Voltage	2.0	V5REF_SUS + 0.5	V	
V_{IL11}	Input Low Voltage	-0.5	0.35Vcc3_3	V	
V_{IH11}	Input High Voltage	0.65Vcc3_3	Vcc3_3 + 0.3	V	
V_{IL12}	Input Low Voltage	-0.150	0.150	V	
V_{IH12}	Input High Voltage	0.660	1.850	V	
V+	Low to high input threshold	1.5	2.0	V	Note 1
V-	High to low input threshold	1.0	1.5	V	Note 1
VHYS	Difference between input thresholds: (V+current value) - (V-current value)	320		mV	Note 1
VTHRAVG	Average of thresholds: ((V+current value) + (V-current value))/2	1.3	1.7	V	Note 1
VRING	AC Voltage at recipient connector	-1	6	V	Note 1, 2
V_{DI}	Differential Input Sensitivity	0.2		V	Note 3, 5
V_{CM}	Differential Common Mode Range	0.8	2.5	V	Note 4, 5
V_{SE}	Single-Ended Receiver Threshold	0.8	2.0	V	Note 5
V_{HSSQ}	HS Squelch Detection Threshold	100	150	mV	Note 5
V_{HSDSC}	HS Disconnect Detection Threshold	525	625	mV	Note 5
V_{HSCM}	HS Data Signaling Common Mode Voltage Range	-50	500	mV	Note 5
V_{HSSQ}	HS Squelch detection threshold	100	150	mV	Note 5
V_{HSDSC}	HS disconnect detection threshold	525	625	mV	Note 5
V_{HSCM}	HS data signaling common mode voltage range	-50	500	mV	Note 5

NOTES:
1. Applies to Ultra DMA Modes greater than Ultra DMA Mode 4.
2. This is an AC Characteristic that represents transient values for these signals.
3. V_{DI} = | USBPx[P] - USBPx[N].
4. Includes V_{DI} range.
5. Applies to High-speed USB 2.0.
6. SATA Vdiff,rx is measured at the SATA connector on the receive side.
7. When probed at the receiver pin of the ICH for data/strobe, the waveform may show a “knee” due to package parasitics. Simulation verifies that this “knee” represents no risk, since a clean waveform is present at the ICH ball input due to internal receiver termination.

Table 711. DC Characteristic Output Signal Association

Symbol	Associated Signals
V_{OH1}/V_{OL1}	**IDE Signals:** PDD[15:0], SDD[15:0], PDIOW#/PDSTOP, SDIOW#/SDSTOP, PDIOR#/ PDWSTB/PRDMARDY, SDIOR#/STWSTB/SRDMARDY, PDDACK#, SDDACK#, PDA[2:0], SDA[2:0], PDCS[3,1]#, SDCS[3,1]#
V_{OH2}/V_{OL2}	**CPU Signals:** A20M#, CPUSLP#, IGNNE#, INIT#, INTR, NMI, SMI#, STPCLK#,
V_{OH4}/V_{OL4}	**PCI-X Signals:** PXAD[63:0], PXC/BE[7:0]#, PXPCIRST#, PXGNT[3]#/GPIO [17], PXGNT[2]#/GPIO [16], PXGNT[1:0]#, PXPAR, PXDEVSEL#, PXPERR#, PXPLOCK#, PXSTOP#, PXTRDY#, PXIRDY#, PXFRAME#, PXSERR#,PXPCLK[0:4], PXSERR#, RASERR#,PXRCOMP, PXIRQ[3:0]/GPIO[36:33**],** PXPAR64, PXREQ64#, PXACK64#, PCIXSBRST# **PCI Signals:** AD[31:0], C/BE[3:0]#, GNT[3:0]#, PAR, DEVSEL#, PERR#, PLOCK#, STOP#, TRDY#, IRDY#, FRAME#, SERR# (1) **LPC/FWH:** LAD[3:0]/FWH[3:0], LFRAME# **AC'97 Signals:** AC_RST#, AC_SDOUT, AC_SYNC **SIU Signals:** SIU0_TXD, SIU1_TXD, SIU0_DTR#, SIU1_DTR#, SIU0_RTS#, SIU1_RTS# **GPIO Signals:** GPIO [43:40]
V_{OL5}/V_{OH5}	**SMBus Signals:** SMBCLK (1), SMBDATA (1) **System Management Signals:** SMLINK[1:0](1)
V_{OL6}/V_{OH6}	**Power Management Signals:** PME#(1), SLP_S3#, SLP_S4#, SLP_S5#, SUS_STAT#, SUSCLK **GPIO Signals:** GPIO[57:56(1), 39:37, 31, 28:27, 25:23, 21:18]; **Interrupt Signals:** SERIRQ, PIRQ[D:A]# (1), PIRQ[H:E]#/GPIO[5:2](1) **Other Signals:** SPKR, WDT_TOUT#/GPIO[32], SATALED#(1)
V_{OL7}/V_{OH7}	**USB Signals**: USBP[3:0][P,N] in Low and Full Speed Modes
V_{OL8}/V_{OH8} Zpd/Zpu	**Hub Interface Signals**: HI[11:0], HI_STBS, HI_STBF
V_{OL9}/V_{OH9}	**SATA Signals:** SATA[1:0]TX[P,N]
V_{HSOI} V_{HSOH} V_{HSOL} V_{CHIRPJ} V_{CHIRPK}	**USB Signals:** USBP[3:0][P:N] in High Speed Modes

1. These signals are open drain.

Table 712. DC Output Characteristics

Symbol	Parameter	Min	Max	Unit	I_{OL} / I_{OH}	Notes
V_{OL1}	Output Low Voltage		0.51	V	6 mA	
V_{OH1}	Output High Voltage	Vcc3_3 - 0.51		V	-6 mA	
V_{OL2}	Output Low Voltage	-0.15	.25(V_CPU_IO)	V	1.5 mA	
V_{OH2}	Output High Voltage	0.9(V_CPU_IO)		V	-.5 mA	Note 1
V_{OL4}	Output Low Voltage		0.55	V	6 mA	
V_{OH4}	Output High Voltage	0.9Vcc3_3		V	-0.5 mA	Note 1
V_{OL5}	Output Low Voltage		0.4	V	4 mA	
V_{OH5}	Output High Voltage	N/A		V	N/A	Note 1
V_{OL6}	Output Low Voltage		0.4	V	4 mA	
V_{OH6}	Output High Voltage	Vcc3_3 - 0.5		V	-2 mA	Note 1
V_{OL7}	Output Low Voltage		0.4	V	5 mA	
V_{OH7}	Output High Voltage	Vcc3_3 - 0.5		V	-2 mA	
V_{OL8}	Output Low Voltage		0.05	V	1 mA	
V_{OH8}	Output High Voltage	0.750	.850	V	-12 mA	
V_{OL9}	Output Low Voltage	400		mVp-p		Note 2
V_{OH9}	Output High Voltage		600	mVp-p		Note 2
Zpd	Pull Down Impedance	48		Ohm		
Zpu	Pull Up Impedance	46		Ohm		
V_{HSOI}	HS Idle Level	-10.0	10.0	mV		
V_{HSOH}	HS Data Signaling High	360	440	mV		
V_{HSOL}	HS Data Signaling Low	-10.0	10.0	mV		
V_{CHIRPJ}	Chirp J Level	700	1100	mV		
V_{CHIRPK}	Chirp K Level	-900	-500	mV		

NOTES:
1. The CPUPWRGD, SERR#, PIRQ[A:H], SATALED#, SMBDATA, SMBCLK, and SMLINK[1:0], RASERR#, PXSERR#, PME#, GPIO [57:56] signals have an open drain driver, and the V_{OH} spec does not apply. This signal must have external pull up resistor.
2. SATA Vdiff,tx is measured at the SATA connector on the transmit side.

Table 713. Other DC Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
V5REF	Intel® 6300ESB ICH Core Well Reference Voltage	4.75	V5REF + 0.5	V	
Vcc3_3	I/O Buffer Voltage	3.135	3.465	V	
VCCREF	Reference voltage for PCI-X	3.135	3.465	V	

NOTES:
1. Includes CLK14, CLK48, HICLK, PCICLK and PXPCICLK.

Table 713. Other DC Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
Vcc1_5, VccHI, VccPLL	Internal Logic Voltage	1.425	1.575	V	
HIREF	Hub Interface Reference Voltage	0.343	0.357	V	
HIVSWING	Hub Interface Voltage Swing (Input to HI_VSWING pin)	0.784	0.816	V	
V5REF_Sus	Suspend Well Reference Voltage	4.75	5.25	V	
VccSus3_3	Suspend Well I/O Buffer Voltage	3.135	3.465	V	
VccSus1_5	Suspend Well Logic Voltage	1.425	1.575	V	
VccRTC	Battery Voltage	2.0	3.6	V	
V_{IT+}	Hysteresis Input Rising Threshold	1.9		V	Applied to USBP[3:0][P,N]
V_{IT-}	Hysteresis Input Falling Threshold		1.3	V	Applied to USBP[3:0]P,N]
V_{DI}	Differential Input Sensitivity	0.2		V	|(USBPx+,USBPx-)|
V_{CM}	Differential Common Mode Range	0.8	2.5	V	Includes V_{DI}
V_{CRS}	Output Signal Crossover Voltage	1.3	2.0	V	
V_{SE}	Single Ended Rcvr Threshold	0.8	2.0	V	
I_{LI1}	ATA Input Leakage Current	-200	200	µA	(0 V < V_{IN} < 5V)
I_{LI2}	PCI_3V Hi-Z State Data Line Leakage	-10	10	µA	(0 V < V_{IN} < 3.3V)
I_{LI3}	PCI_5V Hi-Z State Data Line Leakage	-70	70	µA	Max V_{IN} = 2.7 V Min V_{IN} = 0.5 V
I_{LI4}	Input Leakage Current - Clock signals	-100	+100	µA	Note 1
I_{LI5}	PCI-X Hi-Z State Data Line Leakage	-10	10	µA	(0 V < V_{IN} < 3.3V)
C_{IN}	Input Capacitance - Hub interface Input Capacitance - All Other		8 12	pF	F_C = 1 MHz
C_{OUT}	Output Capacitance		12	pF	F_C = 1 MHz
$C_{I/O}$	I/O Capacitance		12	pF	F_C = 1 MHz
		Typical Value			
C_L	XTAL1	6		pF	
C_L	XTAL2	6		pF	

NOTES:
1. Includes CLK14, CLK48, HICLK, PCICLK and PXPCICLK.

22.4 AC Characteristics

Table 714. Clock Timings (Sheet 1 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure
PCI-X Clock (PXPCLKO[0:4])						
Tcyc	CLK Cycle Time	15	30	ns	7, 9, 10	Figure 39
Thigh	CLK High Time	6		ns		Figure 39
Tlow	CLK Low Time	6		ns		Figure 39
	CLK Slew Rate	1.5	4	V/ns	8, 10	
Spread Spectrum Requirements						
fmod	Modulation Frequency	30	33	kHz		
fspread	Frequency Spread	-1	0	%		
PCI Clock (PCICLK)						
	Period	30	33.3	ns		Figure 44
	High Time	11		ns		Figure 44
	Low Time	11		ns		Figure 44
	Rise Time		4	ns		Figure 44
	Fall Time		4	ns		Figure 44
Oscillator Clock (CLK14)						
	Period	67	70	ns		Figure 44
	High Time	20				Figure 44
	Low Time	20		ns		Figure 44
USB Clock (CLK48)						
f_{clk48}	Operating Frequency	48		MHz	1	
	Frequency Tolerance		500	ppm	2	
	High Time	7		ns		Figure 44
	Low Time	7		ns		Figure 44
	Rise Time		1.2	ns		Figure 44
	Fall Time		1.2	ns		Figure 44
SMBus Clock (SMBCLK)						
f_{smb}	Operating Frequency	10	16	KHz		
t18	High time	4.0	50	us	3	Figure 59
t19	Low time	4.7		us		Figure 59
t20	Rise time		1000	ns		Figure 59
t21	Fall time		300	ns		Figure 59
AC'97 Clock (BITCLK)						
f_{ac97}	Operating Frequency	12.288		MHz		
t26	Output Jitter		750	ps		

Table 714. Clock Timings (Sheet 2 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure
t27	High time	32.56	48.84	ns		Figure 44
t28	Low time	32.56	48.84	ns		Figure 44
t29	Rise time	2.0	6.0	ns	4	Figure 44
t30	Fall time	2.0	6.0	ns	4	Figure 44
Hub Interface Clock (HICLK)						
f_{hi}	Operating Frequency	66		MHz		
t31	High time	6.0		ns		Figure 44
t32	Low time	6.0		ns		Figure 44
t33	Rise time	0.25	1.2	ns		Figure 44
t34	Fall time	0.25	1.2	ns		Figure 44
t35	HICLK leads PCICLK	1.0	4.5	ns	5	
SATA Clock (SATACLKP, SATACLKN)						
t36	Period	9.997	10.003	ns		
t37	Rise time	175	700	ps		
t38	Fall time	175	700	ps		
Suspend Clock (SUSCLK)						
f_{susclk}	Operating Frequency	32		kHz	6	
t39	High Time	10		us	6	
t40a	Low Time	10		us	6	
UART Clock (UART_CLK)						
t8a	Operating Frequency	14.7456	48	MHz		
t9a	Frequency Tolerance		2500	ppm		
t10a	High Time	7		ns		
t11a	Low Time	7		ns		
t12a	Rise Time		3	ns		
t13a	Fall Time		3	ns		

Table 714. Clock Timings (Sheet 3 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure

NOTES:

1. CLK48 is a 48 MHz clock that expects a 40/60% duty cycle.
2. CLK48 is a pass-thru clock that is not altered by the Intel® 6300ESB ICH. This frequency tolerance specification is required for USB 2.0 compliance and is affected by external elements such as the clock generator and the system board.
3. The maximum high time (t18 Max) provide a simple ensured method for devices to detect bus idle conditions.
4. BITCLK Rise and Fall times are measured from 10%VDD and 90%VDD.
5. This specification includes pin-to-pin skew from the clock generator as well as board skew.
6. SUSCLK duty cycle can range from 30% minimum to 70% maximum.
7. For clock frequencies above 33 MHz, the clock frequency may not change beyond the spread-spectrum limits except while RST# is asserted.
8. This slew rate must be met across the minimum peak-to-peak portion of the clock waveform as shown in Figure 43.
9. The minimum clock period must not be violated for any single clock cycle, i.e., accounting for all system jitter.
10. All PCI-X devices must be capable of operating in conventional PCI 33 mode and optionally are capable of conventional PCI 66 mode.

Table 715. PCI-X Interface Timings

Symbol	Parameter	Min	Max	Units	Notes
T_{val}	PXCLKO[4:0] to Signal Valid Delay-bused signals	0.7	3.8	ns	1, 2, 3, 10, 11
$T_{val}(ptp)$	PXCLKO[4:0] to Signal Valid Delay-point to point signals	0.7	3.8	ns	1, 2, 3, 10, 11
T_{on}	Float to Active Delay	0		ns	1, 7, 10, 11
T_{off}	Active to Float Delay		7	ns	1, 7, 11
T_{su}	Input Setup Time to PXCLKO[4:0]-Bused signals	1.7		ns	3, 4, 8
$T_{su}(ptp)$	Input Setup Time to PXCLKO[4:0]-point to point	1.7		ns	3, 4
T_{h}	Input Hold Time from PXCLKO[4:0]	0.5		ns	4
T_{rst}	Reset Active Time after power stable	1		ms	5
$T_{rst-clk}$	Reset Active Time after PXCLKO[4:0] stable	100		µ s	5
$T_{rst-off}$	Reset Active to output float delay		40	ns	5, 6
T_{rrsu}	PXREQ64# to PXPCIRST# setup time	10		ns	
T_{rrh}	PXPCIRST# to PXREQ64# hold Time	0	50	ns	
T_{rhfa}	PXPCIRST# high to first configuration access	2^{26}		clocks	
T_{rhff}	PXPCIRST# high to first PXFRAME# Assertion	5		clocks	
T_{pvrh}	Power valid to PXPCIRST# high	100		ms	
T_{prsu}	PCI-X initialization pattern to PXPCIRST# setup time	10		clocks	
T_{prh}	PXPCIRST# to PCI-X initialization pattern hold time	0	50	ns	9
T_{rlcx}	Delay from PXPCIRST# low to PXCLKO[4:0] frequency change	0		ns	

NOTES:
1. Refer to Figure 41. For timing and measurement condition details, refer to the *PCI-X Addendum to the PCI Local BUS Specification* document.
2. Minimum times are measured at the package pin (not a test point).
3. Setup time for point-to-point signals applies to PXREQ[3:0] and PXGNT[3:0] only. All other signals are bused.
4. See timing measurement conditions in Figure 42.
5. PXPCIRST# is asserted and deasserted asynchronously with respect to PXCLKO[4:0].
6. All output drivers must be floated when RSTIN# is active.
7. For purposes of Active/Float timing measurements, the Hi-Z or “off” state is defined to be when the total current delivered through the component pin is less than or equal to the leakage current specification
8. Setup time applies only when the device is not driving the pin. Devices cannot drive and receive signals at the same time.
9. Maximum value is also limited by delay to the first transaction (T_{rhfa}). The PCI-X initialization pattern control signals after the rising edge of PXPCIRST# must be deasserted no later than two clocks before the first FRAME# and must be floated no later than one clock before FRAME# is asserted.
10. A PCI-X device is permitted to have the minimum values shown for T_{val}, $T_{val(ptp)}$, and T_{on} only in PCI-X mode. In conventional mode, the device must meet the requirements specified in *PCI Local Bus Specification,* Revision 2.2, for the appropriate clock frequency.
11. Device must meet this specification independent of how many outputs switch simultaneously.

Table 716. PCI Interface Timing

Sym	Parameter	Min	Max	Units	Notes	Figure
t40	AD[31:0] Valid Delay	2	11	ns	Min: 0 pF Max: 50 pF	Figure 45
t41	AD[31:0] Setup Time to PCICLK Rising	7		ns		Figure 46
t42	AD[31:0] Hold Time from PCICLK Rising	0		ns		Figure 46
t43	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, PAR, PERR#, PLOCK#, DEVSEL# Valid Delay from PCICLK Rising	2	11	ns	Min: 0 pF Max: 50 pF	Figure 45
t44	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, PAR, PERR#, PLOCK#, IDSEL, DEVSEL# Output Enable Delay from PCICLK Rising	2		ns		Figure 49
t45	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, PERR#, PLOCK#, DEVSEL#, GNT[A:B]# Float Delay from PCICLK Rising	2	28	ns		Figure 47
t46	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, SERR#, PERR#, DEVSEL#, Setup Time to PCICLK Rising	7		ns		Figure 46
t47	C/BE[3:0]#, FRAME#, TRDY#, IRDY#, STOP#, SERR#, PERR#, DEVSEL#, REQ[A:B]# Hold Time from PCICLK Rising	0		ns		Figure 46
t48	PXPCIRST# Low Pulse Width	1		ms		Figure 48
t49	GNT[A:B}#, GNT[5:0]# Valid Delay from PCICLK Rising	2	12	ns		
t50	REQ[A:B]#, REQ[5:0]# Setup Timer to PCICLK Rising	12		ns		

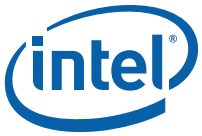


Table 717. IDE PIO and Multiword DMA ModeTiming

Sym	Parameter	Min	Max	Units	Notes	Figure
t60	PDIOR#/PDIOW#/SDIOR#/SDIOW# Active From HICLK Rising	2	20	ns		Figure 50 Figure 51
t61	PDIOR#/PDIOW#/SDIOR#/SDIOW# Inactive From HICLK Rising	2	20	ns		Figure 50 Figure 51
t62	PDA[2:0]/SDA[2:0] Valid Delay From HICLK Rising	2	30	ns		Figure 50
t63	PDCS1#/SDCS1#, PDCS3#/SDCS3# Active From HICLK Rising	2	30	ns		Figure 50
t64	PDCS1#/SDCS1#, PDCS3#/SDCS3# Inactive From HICLK Rising	2	30	ns		Figure 50
t65	PDDACK#/SDDACK# Active From HICLK Rising	2	20	ns		Figure 51
t66	PDDACK#/SDDACK# Inactive From HICLK Rising	2	20	ns		
t67	PDDREQ/SDDREQ Setup Time to HICLK Rising	7		ns		Figure 51
t68	PDDREQ/SDDREQ Hold From HICLK Rising	7		ns		Figure 51
t69	PDD[15:0]/SDD[15:0] Valid Delay From HICLK Rising	2	30	ns		Figure 50 Figure 51
t70	PDD[15:0]/SDD[15:0] Setup Time to HICLK Rising	10		ns		Figure 50 Figure 51
t71	PDD[15:0]/SDD[15:0] Hold From HICLK Rising	7		ns		Figure 50 Figure 51
t72	PIORDY/SIORDY Setup Time to HICLK Rising	7		ns	1	Figure 50
t73	PIORDY/SIORDY Hold From HICLK Rising	7		ns	1	Figure 50
t74	PIORDY/SIORDY Inactive Pulse Width	48		ns		Figure 50
t75	PDIOR#/PDIOW#/SDIOR#/SDIOW# Pulse Width Low				2, 3	Figure 50 Figure 51
t76	PDIOR#/PDIOW#/SDIOR#/SDIOW# Pulse Width High				3, 4	Figure 50 Figure 51

NOTES:
1. IORDY is internally synchronized. This timing is to ensure recognition on the next clock.
2. PIORDY sample point from DIOx# assertion and PDIOx# active pulse width is programmable from 2-5 PCI clocks when the drive mode is Mode 2 or greater. Refer to the ISP field in the IDE Timing Register.
3. PIORDY sample point from DIOx# assertion, PDIOx# active pulse width and PDIOx# inactive pulse width cycle time is the compatible timing when the drive mode is Mode 0/1. Refer to the TIM0/1 field in the IDE timing register.
4. PDIOx# inactive pulse width is programmable from 1-4 PCI clocks when the drive mode is Mode 2 or greater. Refer to the RCT field in the IDE Timing Register.

Table 718. Ultra ATA Timing (Mode 0, Mode 1, Mode 2) (Sheet 1 of 2)

Sym	Parameter (1)	Mode 0 (ns)		Mode 1 (ns)		Mode 2 (ns)		Measuring Location	Figure
		Min	Max	Min	Max	Min	Max		
t80	Sustained Cycle Time (T2cyctyp)	240		160		120		Sender Connector	
t81	Cycle Time (Tcyc)	112		73		54		End Recipient Connector	Figure 53
t82	Two Cycle Time (T2cyc)	230		153		115		Sender Connector	Figure 53
t83a	Data Setup Time (Tds)	15		10		7		Recipient Connector	Figure 53
t83b	Recipient IC data setup time (from data valid until STROBE edge) (see Note 2) (Tdsic)	14.7		9.7		6.8		Intel® 6300ESB ICH ball	
t84a	Data Hold Time (Tdh)	5		5		5		Recipient Connector	Figure 53
t84b	Recipient IC data hold time (from STROBE edge until data may become invalid) (see Note 2) (Tdhic)	4.8		4.8		4.8		Intel® 6300ESB ICH ball	
t85a	Data Valid Setup Time (Tdvs)	70		48		31		Sender Connector	Figure 53
t85b	Sender IC data valid setup time (from data valid until STROBE edge) (see Note 2) (Tdvsic)	72.9		50.9		33.9		Intel® 6300ESB ICH ball	
t86a	Data Valid Hold Time (Tdvh)	6.2		6.2		6.2		Sender Connector	Figure 53
t86b	Sender IC data valid hold time (from STROBE edge until data may become invalid) (see Note 2) (Tdvhic)	9		9		9		Intel® 6300ESB ICH ball	
t87	Limited Interlock Time (Tli)	0	150	0	150	0	150	See Note 2	Figure 55
t88	Interlock Time w/ Minimum (Tmli)	20		20		20		Host Connector	Figure 55
t89	Envelope Time (Tenv)	20	70	20	70	20	70	Host Connector	Figure 52
t90	Ready to Pause Time (Trp)	160		125		100		Recipient Connector	Figure 54
t91	DMACK setup/hold Time (Tack)	20		20		20		Host Connector	Figure 52, Figure 55
t92a	CRC Word Setup Time at Host (Tcvs)	70		48		31		Host Connector	

NOTES:
1. The specification symbols in parentheses correspond to the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification name.
2. See the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification for further details on measuring these timing parameters.

Table 718. Ultra ATA Timing (Mode 0, Mode 1, Mode 2) (Sheet 2 of 2)

Sym	Parameter (1)	Mode 0 (ns)		Mode 1 (ns)		Mode 2 (ns)		Measuring Location	Figure
		Min	Max	Min	Max	Min	Max		
t92b	CRC word valid hold time at sender (from DMACK# negation until CRC may become invalid) (see Note 2) (Tcvh)	6.2		6.2		6.2		Host Connector	
t93	STROBE output released-to-driving to the first transition of critical timing (Tzfs)	0		0		0		Device Connector	Figure 55
t94	Data Output Released-to-Driving Until the First Tunisian of Critical Timing (Tdzfs)	70		48		31		Sender Connector	Figure 52
t95	Unlimited Interlock Time (Tui)	0		0		0		Host Connector	Figure 52
t96a	Maximum time allowed for output drivers to release (from asserted or negated) (Taz)		10		10		10	See Note 2	
t96b	Minimum time for drivers to assert or negate (from released) (Tzad)	0		0		0		Device Connector	
t97	Ready-to-final-STROBE time (no STROBE edges shall be sent this long after negation of DMARDY#) (Trfs)		75		70		60	Sender Connector	Figure 52
t98a	Maximum time before releasing IORDY (Tiordyz)		20		20		20	Device Connector	
t98b	Minimum time before driving IORDY (see Note 2) (Tziordy)	0		0		0		Device Connector	
t99	Time from STROBE edge to negation of DMARQ or assertion of STOP (when sender terminates a burst) (Tss)	50		50		50		Sender Connector	Figure 54

NOTES:
1. The specification symbols in parentheses correspond to the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification name.
2. See the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification for further details on measuring these timing parameters.

Table 719. Ultra ATA Timing (Mode 3, Mode 4, Mode 5) (Sheet 1 of 2)

Sym	Parameter (1)	Mode 3 (ns)		Mode 4 (ns)		Mode 5 (ns)		Measuring Location	Figure
		Min	Max	Min	Max	Min	Max		
t80	Sustained Cycle Time (T2cyctyp)	90		60		40		Sender Connector	
t81	Cycle Time (Tcyc)	39		25		16.8		End Recipient Connector	Figure 53
t82	Two Cycle Time (T2cyc)	86		57		38		Sender Connector	Figure 53
t83	Data Setup Time (Tds)	7		5		4.0		Recipient Connector	Figure 53
t83b	Recipient IC data setup time (from data valid until STROBE edge) (see Note 2) (Tdsic)	6.8		4.8		2.3		Intel® 6300ESB ICH Balls	
t84	Data Hold Time (Tdh)	5		5		4.6		Recipient Connector	Figure 53
t84b	Recipient IC data hold time (from STROBE edge until data may become invalid) (see Note 2) (Tdhic)	4.8		4.8		2.8		Intel® 6300ESB ICH Balls	
t85	Data Valid Setup Time (Tdvs)	20		6.7		4.8		Sender Connector	Figure 52 Figure 53
t85b	Sender IC data valid setup time (from data valid until STROBE edge) (see Note 2) (Tdvsic)	22.6		9.5		6.0		Intel® 6300ESB ICH Balls	
t86	Data Valid Hold Time (Tdvh)	6.2		6.2		4.8		Sender Connector	Figure 52 Figure 53
t86b	Sender IC data valid hold time (from STROBE edge until data may become invalid) (see Note 2) (Tdvhic)	9.0		9.0		6.0		Intel® 6300ESB ICH Balls	
t87	Limited Interlock Time (Tli)	0	100	0	100	0	75	See Note 2	Figure 55
t88	Interlock Time w/ Minimum (Tmli)	20		20		20		Host Connector	Figure 55
t89	Envelope Time (Tenv)	20	55	20	55	20	50	Host Connector	Figure 53
t90	Ready to Pause Time (Trp)	100		100		85		Recipient Connector	Figure 54

NOTES:
1. The specification symbols in parentheses correspond to the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification name.
2. See the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification for further details on measuring these timing parameters.

Table 719. Ultra ATA Timing (Mode 3, Mode 4, Mode 5) (Sheet 2 of 2)

Sym	Parameter (1)	Mode 3 (ns)		Mode 4 (ns)		Mode 5 (ns)		Measuring Location	Figure
		Min	Max	Min	Max	Min	Max		
t91	DMACK setup/hold Time (Tack)	20		20		20		Host Connector	Figure 5 5
t92a	CRC Word Setup Time at Host (Tcvs)	20		6.7		10		Host Connector	
t92b	CRC Word Hold Time at Sender CRC word valid hold time at sender (from DMACK# negation until CRC may become invalid) (see Note 2) (Tcvh)	6.2		6.2		10.0		Host Connector	
t93	STROBE output released-to-driving to the first transition of critical timing (Tzfs)	0		0		35		Device Connector	Figure 5 5
t94	Data Output Released-to-Driving Until the First Transition of Critical Timing (Tdzfs)	20.0		6.7		25		Sender Connector	Figure 5 2
t95	Unlimited Interlock Time (Tui)	0		0		0		Host Connector	Figure 5 2
t96a	Maximum time allowed for output drivers to release (from asserted or negated) (Taz)		10		10		10	See Note 2	
t96b	Drivers to assert or negate (from released) (Tzad)	0		0		0		Device Connector	
t97	Ready-to-final-STROBE time (no STROBE edges shall be sent this long after negation of DMARDY#) (Trfs)		60		60		50	Sender Connector	Figure 5 2
t98a	Maximum time before releasing IORDY (Tiordyz)		20		20		20	Device Connector	
t98b	Minimum time before driving IORDY (see Note 2) (Tziordy)	0		0		0		Device Connector	
t99	Time from STROBE edge to negation of DMARQ or assertion of STOP (when sender terminates a burst) (Tss)	50		50		50		Sender Connector	Figure 5 4

NOTES:
1. The specification symbols in parentheses correspond to the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification name.
2. See the AT Attachment - 6 with Packet Interface (ATA/ATAPI - 6) specification for further details on measuring these timing parameters.

Table 720. Universal Serial Bus Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
Full Speed Source (Note 7)						
t100	USBPx+, USBPx- Driver Rise Time	4	20	ns	1, C_L = 50 pF	Figure 56
t101	USBPx+, USBPx- Driver Fall Time	4	20	ns	1, C_L = 50 pF	Figure 56
t102	Source Differential Driver Jitter To Next Transition For Paired Transitions	 -3.5 -4	 3.5 4	 ns ns	2, 3	Figure 57
t103	Source SE0 interval of EOP	160	175	ns	4	Figure 58
t104	Source Jitter for Differential Transition to SE0 Transition	-2	5	ns	5	
t105	Receiver Data Jitter Tolerance To Next Transition For Paired Transitions	 -18.5 -9	 18.5 9	 ns ns	3	Figure 57
t106	EOP Width: Must accept as EOP	82		ns	4	Figure 58
t107	Width of SE0 interval during differential transition		14	ns		
Low Speed Source (Note 8)						
t108	USBPx+, USBPx- Driver Rise Time	75	300	ns	1, 6 C_L = 50 pF C_L = 350 pF	Figure 56
t109	USBPx+, USBPx- Driver Fall Time	75	300	ns	1, 6 C_L = 50 pF C_L = 350 pF	Figure 56
t110	Source Differential Driver Jitter To Next Transition For Paired Transitions	 -25 -14	 25 14	 ns ns	2, 3	Figure 57
t111	Source SE0 interval of EOP	1.25	1.50	µs	4	Figure 58
t112	Source Jitter for Differential Transition to SE0 Transition	-40	100	ns	5	
t113	Receiver Data Jitter Tolerance To Next Transition For Paired Transitions	 -152 -200	 152 200	 ns ns	3	Figure 57

NOTES:
1. Driver output resistance under steady state drive is spec'd at 28 ohms at minimum and 43 ohms at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. Full Speed Data Rate has minimum of 11.97 Mbps and maximum of 12.03 Mbps.
8. Low Speed Data Rate has a minimum of 1.48 Mbps and a maximum of 1.52 Mbps.
9. Refer to the latest revision of the *Universal Serial Bus Specification* for High speed source timings

Table 720. Universal Serial Bus Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
Full Speed Source (Note 7)						
t114	EOP Width: Must accept as EOP	670		ns	4	Figure 58
t115	Width of SE0 interval during differential transition		210	ns		

NOTES:
1. Driver output resistance under steady state drive is spec'd at 28 ohms at minimum and 43 ohms at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. Full Speed Data Rate has minimum of 11.97 Mbps and maximum of 12.03 Mbps.
8. Low Speed Data Rate has a minimum of 1.48 Mbps and a maximum of 1.52 Mbps.
9. Refer to the latest revision of the *Universal Serial Bus Specification* for High speed source timings

Table 721. SATA Interface Timings

Sym	Parameter	Min	Max	Units	Notes	Figure
	Operating Data Period	666.43	670.12	ps		
	Rise Time	0.2	0.41	UI	1	
	Fall Time	0.2	0.41	UI	2	
	TX differential skew		20	ps		
	COMRESET	310.4	329.6	ns	3	
	COMWAKE transmit spacing	103.5	109.9	ns	3	
	OOB Operating Data period	646.67	686.67	ns	4	

NOTES:
1. 20% - 80% at transmitter
2. 80% - 20% at transmitter
3. As measured from 100 mV differential crosspoints of last and first edges of burst.
4. Operating data period during Out-Of-Band burst transmissions.

Table 722. SMBus Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
t130	Bus Tree Time Between Stop and Start Condition	4.7		µs		Figure 59
t131	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4.0		µs		Figure 59
t132	Repeated Start Condition Setup Time	4.7		µs		Figure 59
t133	Stop Condition Setup Time	4.0		µs		Figure 59
t134	Data Hold Time	0		ns	4	Figure 59
t135	Data Setup Time	250		ns		Figure 59
t136	Device Time Out	25	35	ms	1	
t137	Cumulative Clock Low Extend Time (slave device)		25	ms	2	Figure 60
t138	Cumulative Clock Low Extend Time (master device)		10	ms	3	Figure 60

NOTES:
1. A device will timeout when any clock low exceeds this value.
2. t137 is the cumulative time a slave device is allowed to extend the clock cycles in one message from the initial start to stop. If a slave device exceeds this time, it is expected to release both its clock and data lines and reset itself.
3. t138 is the cumulative time a master device is allowed to extend its clock cycles within each byte of a message as defined from start-to-ack, ack-to-ack or ack-to-stop.
4. t134 has a minimum timing for I^2C of 0 ns, while the minimum timing for SMBus is 300 ns.

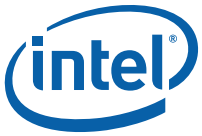


Table 723. AC'97 Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
t_{setup}	ACSDIN[2:0] Setup to Falling Edge of BITCLK	10		ns		Figure 65
t_{hold}	ACSDIN[2:0] Hold from Falling Edge of BITCLK	10		ns		Figure 65
t_{co}	ACSYNC, ACSDOUT valid delay from rising edge of BITCLK		15	ns		Figure 65

Table 724. LPC Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
t150	LAD[3:0] Valid Delay from PCICLK Rising	2	11	ns		Figure 45
t151	LAD[3:0] Output Enable Delay from PCICLK Rising	2		ns		Figure 49
t152	LAD[3:0] Float Delay from PCICLK Rising		28	ns		Figure 47
t153	LAD[3:0] Setup Time to PCICLK Rising	7		ns		Figure 46
t154	LAD[3:0] Hold Time from PCICLK Rising	0		ns		Figure 46
t155	LDRQ[1:0]# Setup Time to PCICLK Rising	12		ns		Figure 46
t156	LDRQ[1:0]# Hold Time from PCICLK Rising	0		ns		Figure 46
t157	LFRAME# Valid Delay from PCICLK Rising	2	12	ns		Figure 45

Table 725. Miscellaneous Timings

Sym	Parameter	Min	Max	Units	Notes	Fig
t160	SERIRQ Setup Time to PCICLK Rising	7		ns		Figure 46
t161	SERIRQ Hold Time from PCICLK Rising	0		ns		Figure 46
t162	RI# Pulse Width	2		RTCCLK		Figure 48
t163	SPKR Valid Delay from CLK14 Rising		200	ns		Figure 45
t164	SERR# Active to NMI Active		200	ns		
t165	IGNNE# Inactive from FERR# Inactive		230	ns		

Table 726. UART Timings

Sym	Parameter	Min	Max	Units	Notes	Fig
t150a	SIU0_TXD, SIU1_TXD Valid Delay from UART_CLK rising	2	13	ns		
t151a	SIU0_DTR#, SIU0_RTS#, SIU1_DTR#, and SIU1_RTS# Valid Delay from PCICLK Rising	2	13	ns		
t152a	SIU0_RXD, SIU1_RXD Setup Time to UART_CLK Rising	7		ns		
t153a	SIU0_RXD, SIU1_RXD Hold Time to UART_CLK Rising	0		ns		
t154a	SIU0_CTS#, SIU0_DSR#, SIU0_DCD#, SIU0_RI#, SIU1_CTS#, SIU1_DSR#, SIU1_DCD#, and SIU1_RI# High Time	100		ns		
t155a	SIU0_CTS#, SIU0_DSR#, SIU0_DCD#, SIU0_RI#, SIU1_CTS#, SIU1_DSR#, SIU1_DCD#, and SIU1_RI# Low Time	100		ns		

Table 727. Power Sequencing and Reset Signal Timings

Sym	Parameter	Min	Max	Units	Notes	Fig
t170	VccRTC active to RTCRST# inactive	5	-	ms		Figure 61
t171	V5RefSus active to VccSus3_3, VccSus1_5 active	0	-	ms	1, 2	Figure 61
t172	VccRTC supply active to VccSus supplies active	0	-	ms	3	Figure 61
t173	VccSus supplies active to RSMRST# inactive	10	-	ms		Figure 61 Figure 62
t174	V5Ref active to Vcc3_3, Vcc1_5, VccHI active	0	-	ms	1, 2	Figure 61
t175	VccSus supplies active to Vcc3_3, Vcc1_5, VccHI supplies active	0	-	ms	3	Figure 61
t176	Vcc3_3, Vcc1_5, VccHI supplies active to PWROK.	99	-	ms		Figure 61 Figure 62 Figure 64
t177	PWROK active to SUS_STAT# inactive	32	38	RTCCLK	4	Figure 62 Figure 64
t178	SUS_STAT# inactive to PXPCIRST# inactive	1	3	RTCCLK		Figure 62 Figure 64
t179	AC_RST# active low pulse width	1		us		
t180	AC_RST# inactive to BIT_CLK startup delay	162.8		ns		

NOTES:
1. The V5Ref supply must power up before or simultaneous with its associated 3.3 V supply, and must power down simultaneous with or after the 3.3 V supply. See the Intel® 6300ESB ICH *Design Guide* for details.
2. The associated 3.3 V and 1.5 V supplies must power up or down simultaneously.
3. The VccSus supplies must **never** be active while the VccRTC supply is inactive.
4. SYSRESET# is not checked for PWROK transitions (t177).

Table 728. Power Management Timings

Sym	Parameter	Min	Max	Units	Notes	Fig
t181	VccSus active to SLP_S5#, SUS_STAT# and PXPCIRST# active		50	ns		Figure 62
t182 t183	RSMRST# inactive to SUSCLK running, SLP_S5# inactive		110	ms	6	Figure 62
t183a	SLPS5# inactive to SLP_S4# inactive	1	2	RTCCLK		Figure 62
t183b	SLPS4# inactive to SLP_S3# inactive	1	2	RTCCLK		Figure 62
t184			50	ns		Figure 62 Figure 64
t187	STPCLK# active to Stop Grant cycle	N/A	N/A		2	Figure 63
t188	Stop Grant cycle to CPUSLP# active	60	63	PCICLK	3	Figure 63 Figure 64
t189	S1 Wake Event to CPUSLP# inactive	1	25	PCICLK	4	Figure 63
t190	CPUSLP# inactive to STPCLK# inactive	3.87	245	µs		Figure 63
t192	CPUSLP# active to SUS_STAT# active	2	4	RTCCLK	1	Figure 64
t193	SUS_STAT# active to PXPCIRST# active	9	21	RTCCLK	1	
t194	PXPCIRST# active to SLP_S3# active	1	2	RTCCLK	1	Figure 64
t194a	SLP_S3# active to SLP_S4# active	1	2	RTCCLK	1	Figure 64
t195	SLP_S4# active to SLP_S5# active	1	2	RTCCLK	1, 5	Figure 64
t196	SLP_S3# active to PWROK inactive	0		ms	4	Figure 64
t197	PWROK inactive to Vcc supplies inactive	20		ns		Figure 64
t198	Wake Event to SLP_S5# inactive	1	10	RTCCLK	1	
t198a	Wake Event to SLP_S4# inactive(S4 Wake)	1	10	RTCCLK	1	
t198b	S3 Wake Event to SLP_S3# inactive(S3 Wake)	0	2	RTCCLK	1	
t198d	SLP_S5# inactive to SLP_S4# inactive	1	2	RTCCLK	1	Figure 64
t198e	SLP_S4# inactive to SLP_S3# inactive	1	2	RTCCLK	1	Figure 64
t220	THRMTRIP# active to SLP_S3#, SLP_S4#, SLP_S5# active		2	PCI CLK		

NOTES:
1. These transitions are clocked off the internal RTC. 1 RTC clock is approximately 32 µs.
2. The Intel® 6300ESB ICH STPCLK# assertion will trigger the processor to send a stop grant acknowledge cycle. The timing for this cycle getting to the Intel® 6300ESB ICH is dependant on the processor and the memory controller.
3. These transitions are clocked off the 33 MHz PCICLK. 1 PCICLK is approximately 30ns.
4. The Intel® 6300ESB ICH has no maximum timing requirement for this transition. It is up to the system designer to determine if the SLP_S3#, SLP_S4# and SLP_S5# signals are used to control the power planes.
5. If the transition to S5 is due to Power Button Override, SLP_S3#, SLP_S4# and SLP_S5# are asserted together similar to timing t194 (PXPCIRST# active to SLP_S3# active).
6. If there is no RTC battery in the system, so VccRTC and the VccSus supplies come up together, the delay from RTCRST# and RSMRST# inactive to SUSCLK toggling may be as much as 2.5 s.
7. This value is programmable in multiples of 1024 PCI CLKs. Maximum is 8192 PCI CLKs (245.6 µs).).

22.5 Timing Diagrams and Test Conditions

22.5.1 PCI-X

Figure 39. PCI-X 3.3V Clock

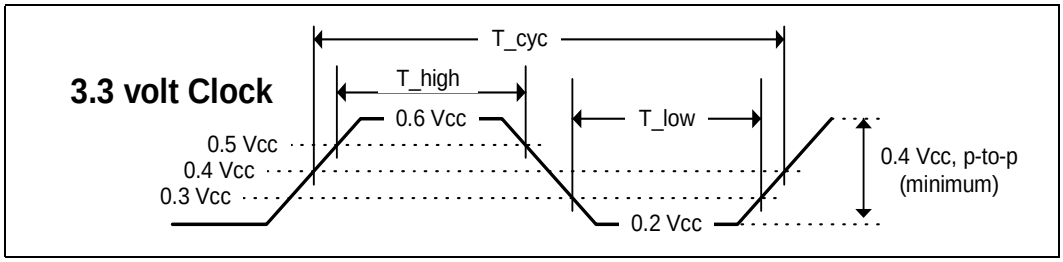


Figure 40. Clock Uncertainty (PXPCLK[0:4])

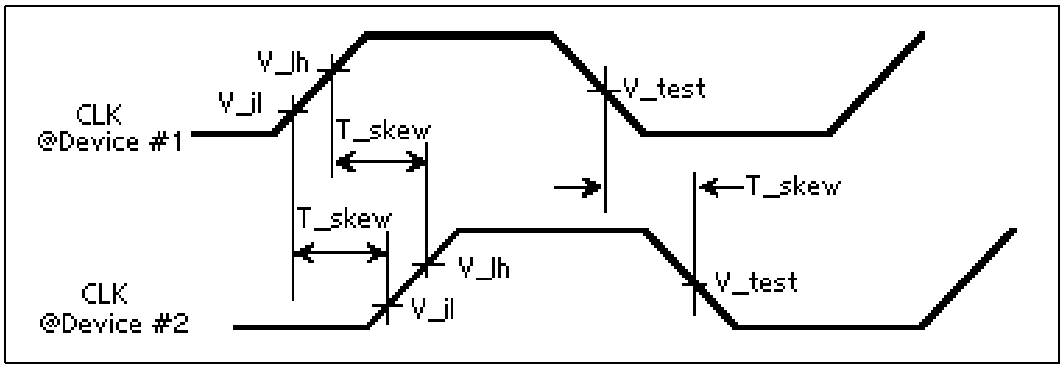


Table 729. Clock Uncertainty Parameters

Symbol	Parameter	Units
Vtest-clk	0.4Vcc	V
Tskew	0.4(Max)	ns

Figure 41. PCI-X Output Timing

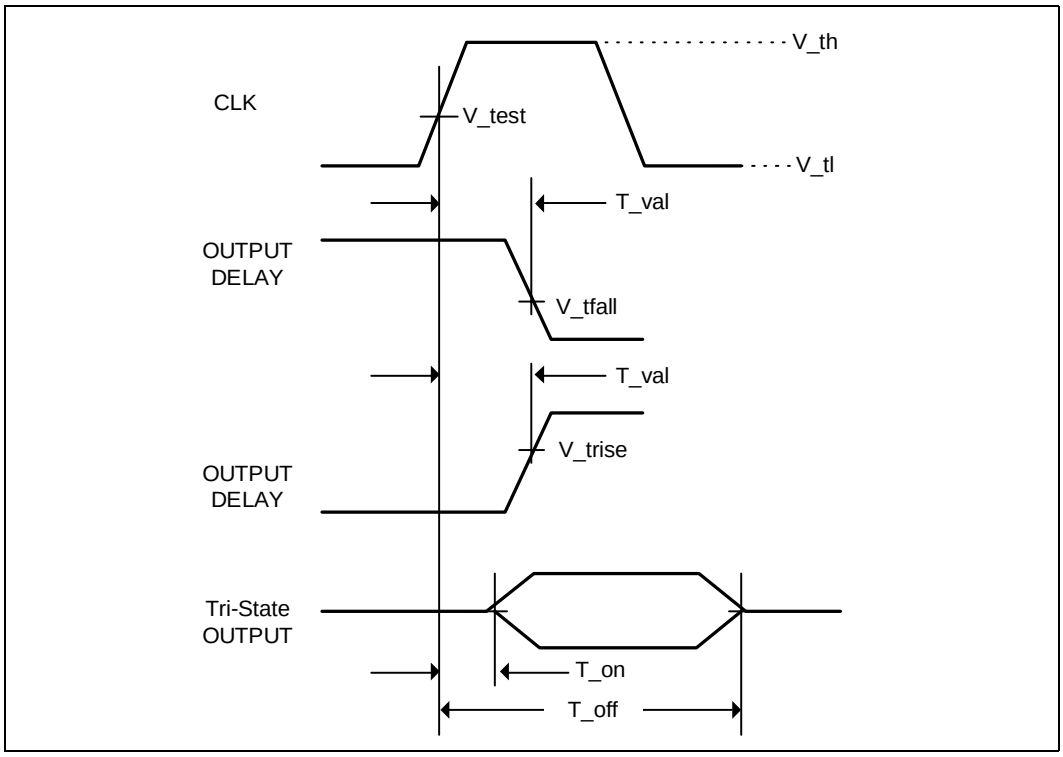


Figure 42. PCI-X Input Timing

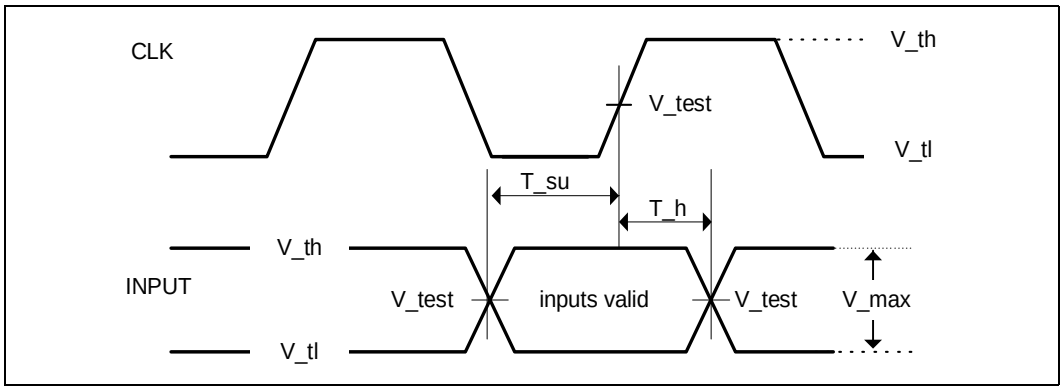
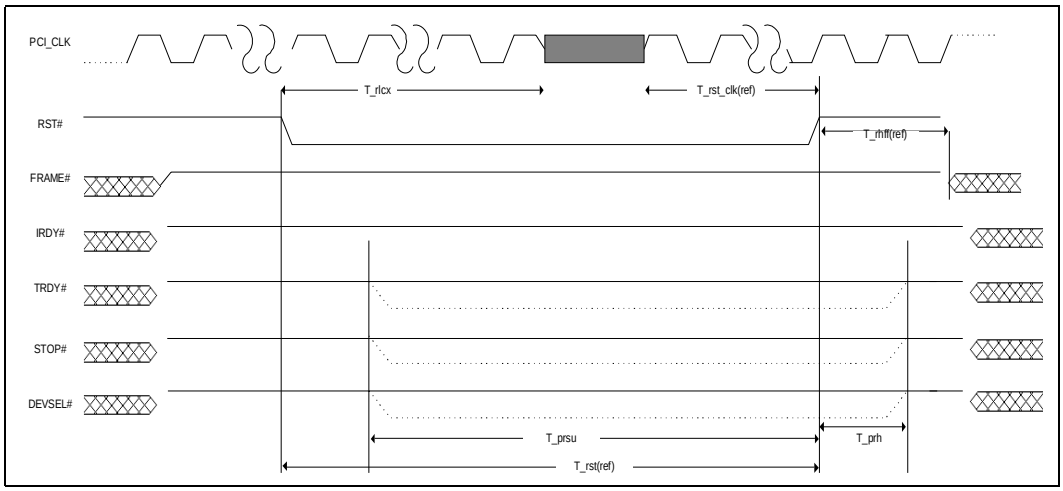


Table 730. PCI-X Measurement Condition Parameters

Symbol	Value	Units	Notes
Vth	0.6 Vcc	V	1
Vtl	0.25 Vcc	V	1
Vtest	0.4 Vcc	V	
Vrise	0.285 Vcc	V	2
Vtfall	0.615 Vcc	V	2
Vmax	0.4 Vcc	V	1
Input Signal Slew Rate	1.5	V/ns	3

NOTES:
1. The test for the 3.3V environment is done with 0.1*V_{cc} of overdrive. V_{max} specifies the maximum peak-to-peak waveform allowed for measuring input timing. Production testing is permitted to use different voltage values but must correlate results back to these parameters.
2. V_{trise} and V_{tfall} are reference voltages for timing measurements only.
3. Input signal slew rate in PCI-X mode is measured between V_{il} and V_{ih}.

Figure 43. PCI-X RST# Timing for switching to PCI-X Mode Pull-ups



22.5.2 System Clocks and General Timing

Figure 44. Clock Timing

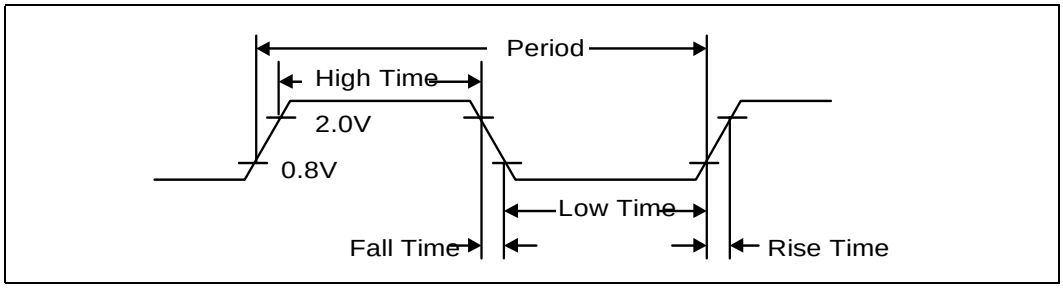


Figure 45. Valid Delay from Rising Clock Edge

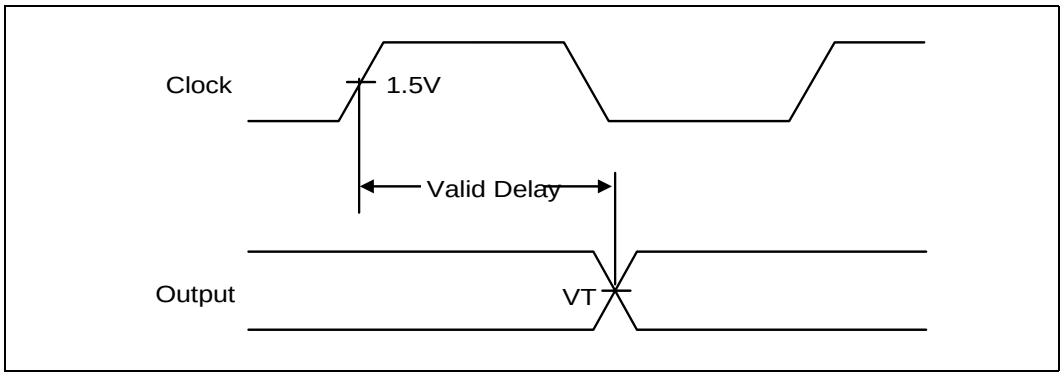


Figure 46. Setup and Hold Times

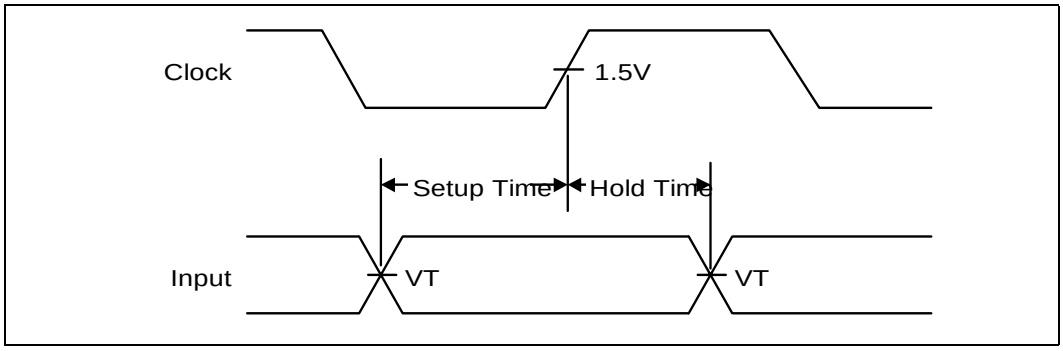


Figure 47. Float Delay

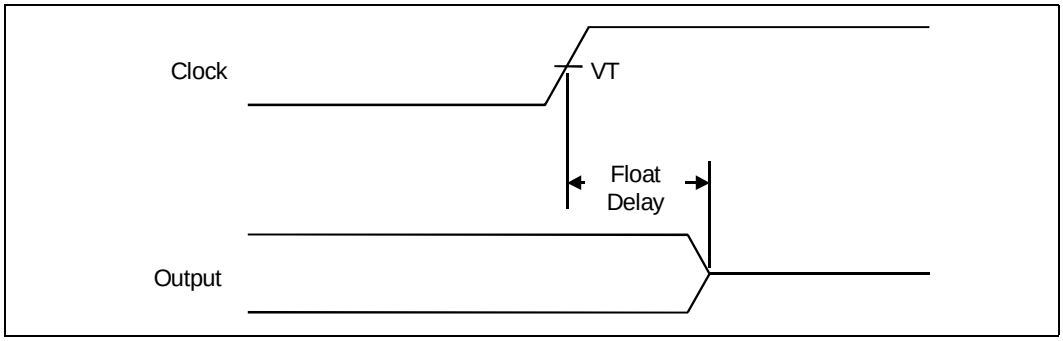


Figure 48. Pulse Width

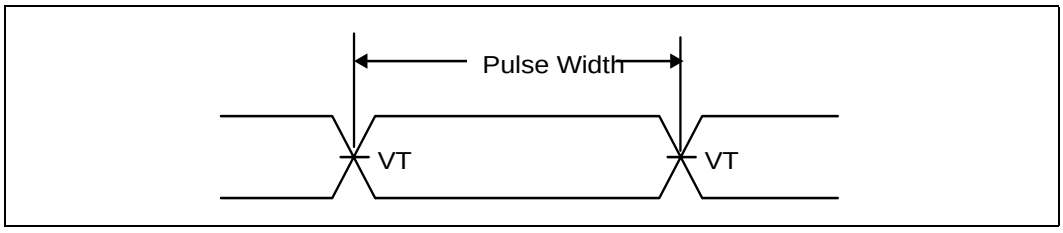
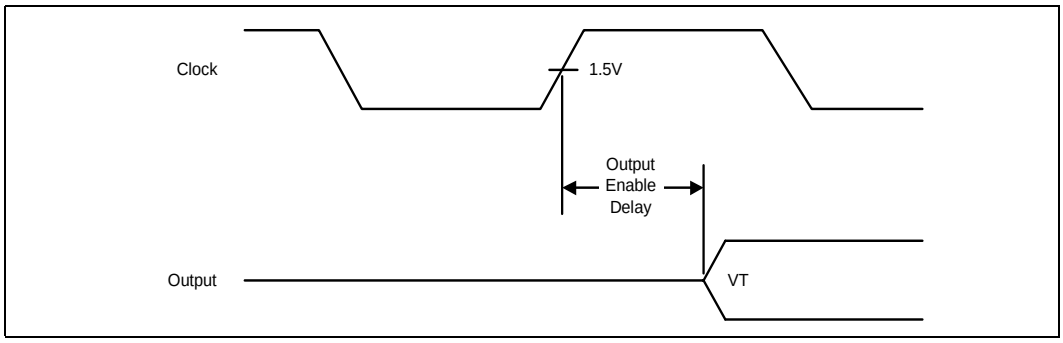


Figure 49. Output Enable Delay



22.5.3 IDE and Ultra ATA Timing

Figure 50. IDE PIO Mode

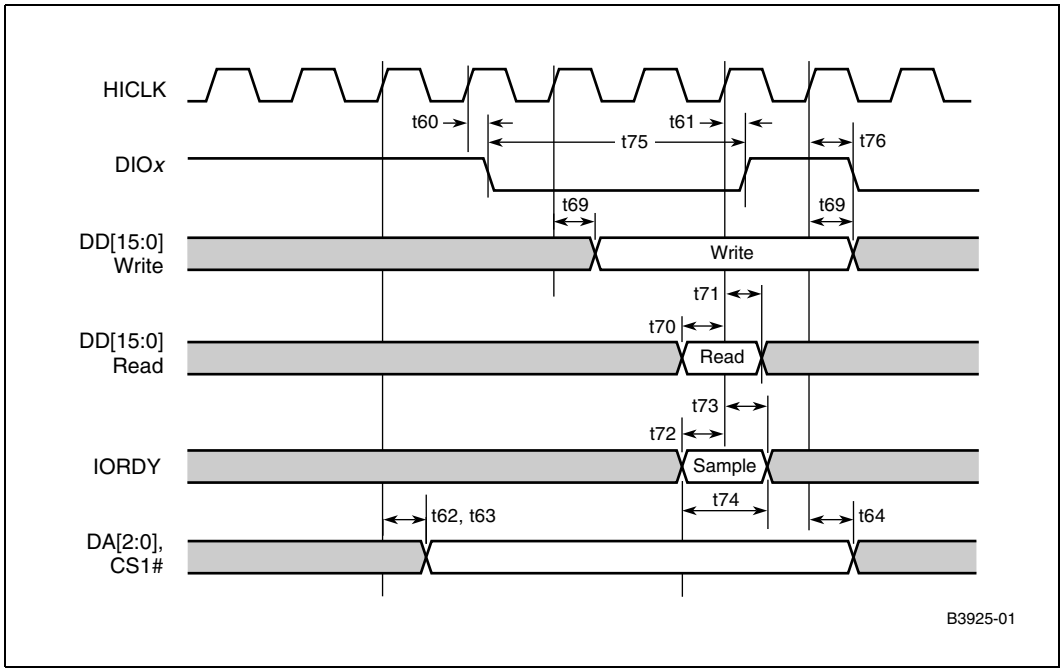


Figure 51. IDE Multiword DMA

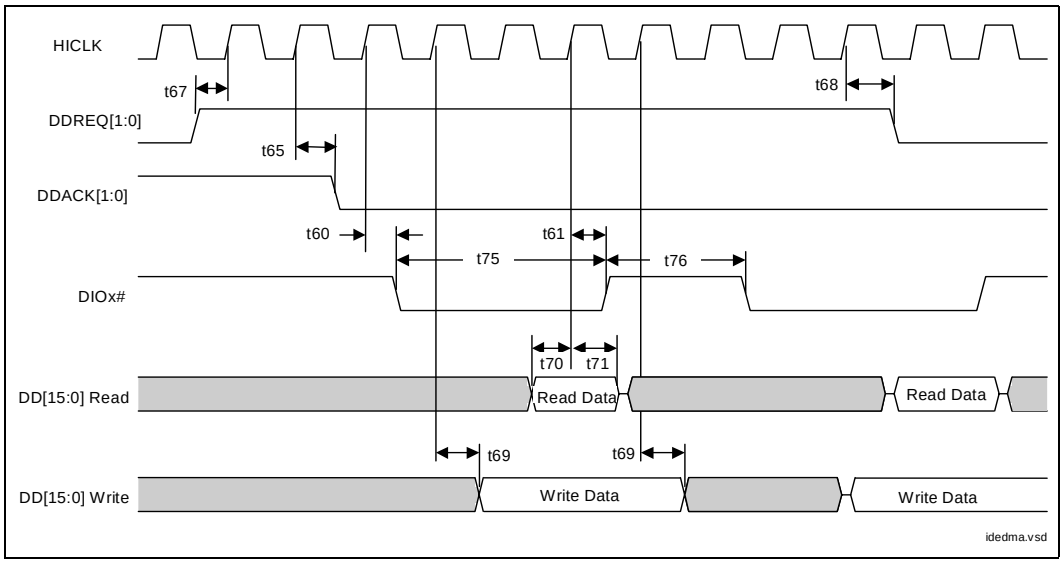


Figure 52. Ultra ATA Mode (Drive Initiating a Burst Read)

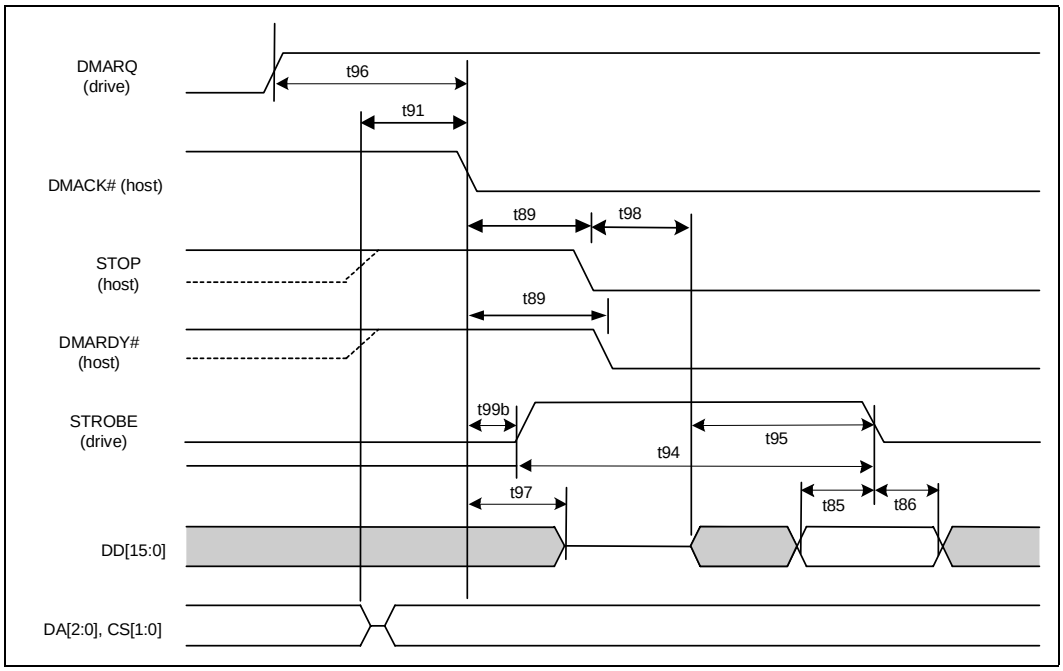


Figure 53. Ultra ATA Mode (Sustained Burst)

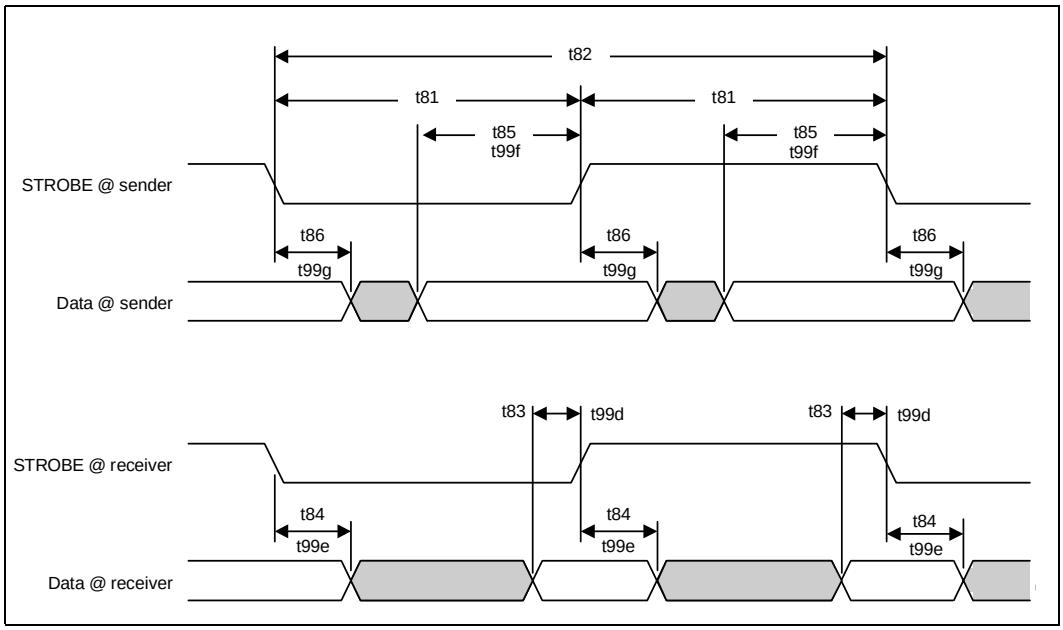


Figure 54. Ultra ATA Mode (Pausing a DMA Burst)

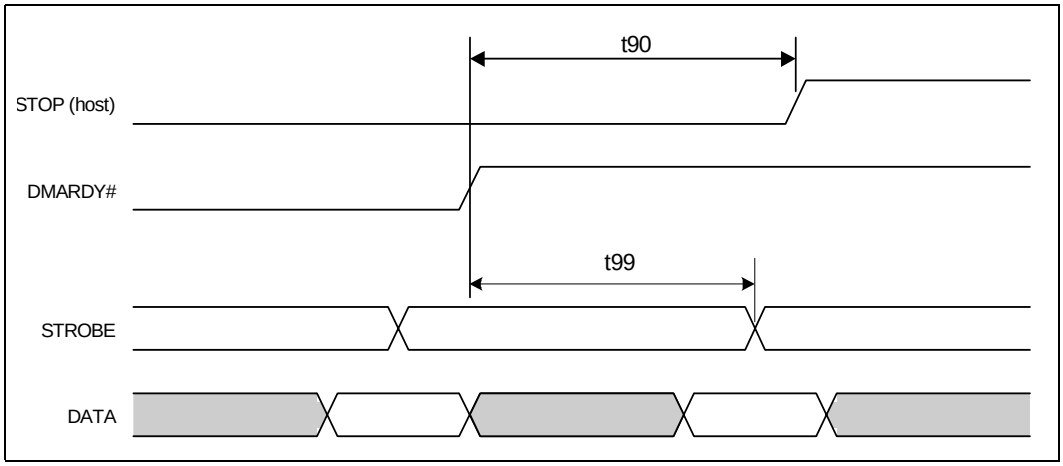
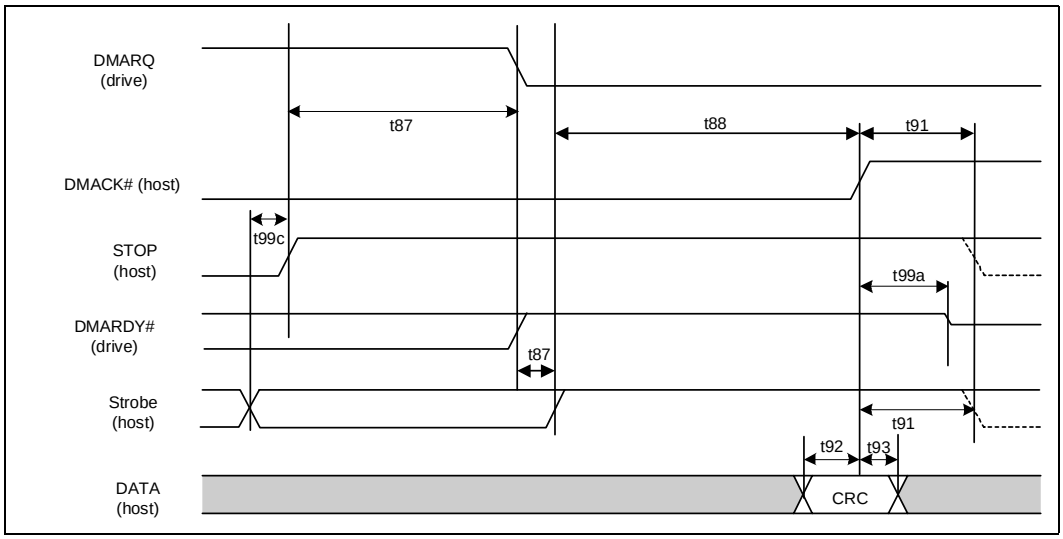


Figure 55. Ultra ATA Mode (Terminating a DMA Burst)



22.5.4 USB

Figure 56. USB Rise and Fall Times

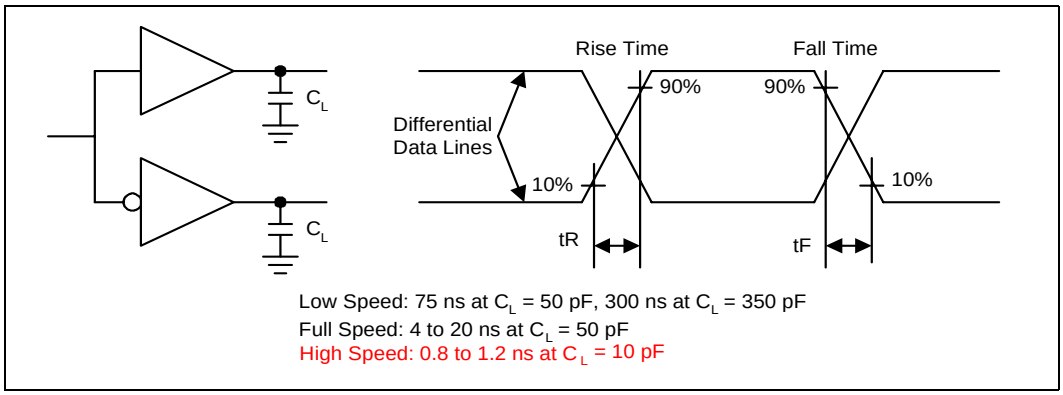


Figure 57. USB Jitter

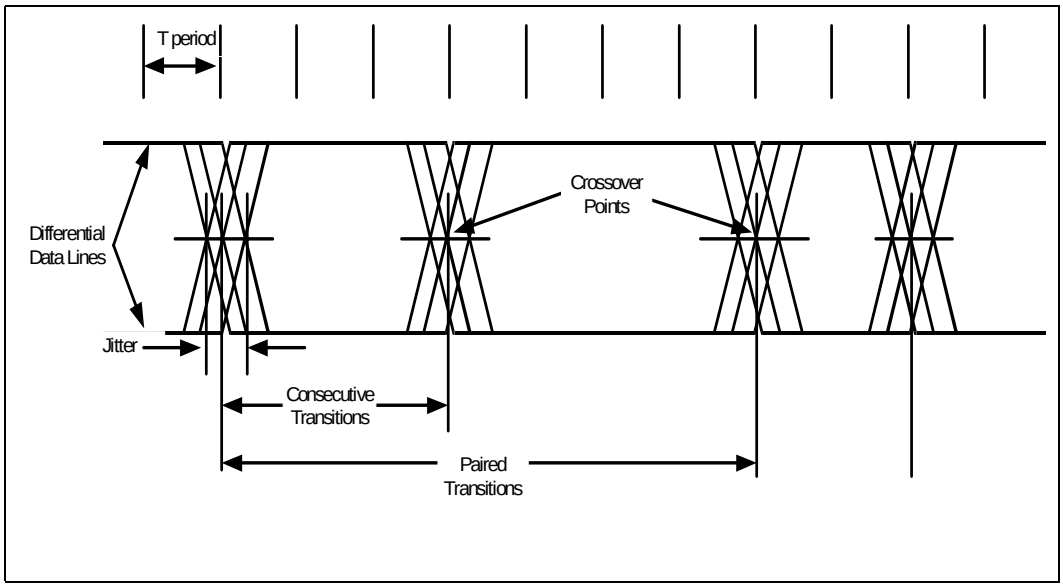
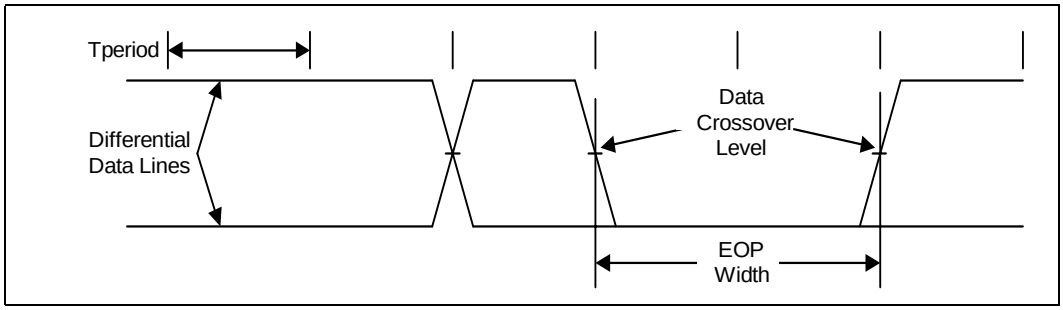


Figure 58. USB EOP Width



22.5.5 SMBus

Figure 59. SMBus Transaction

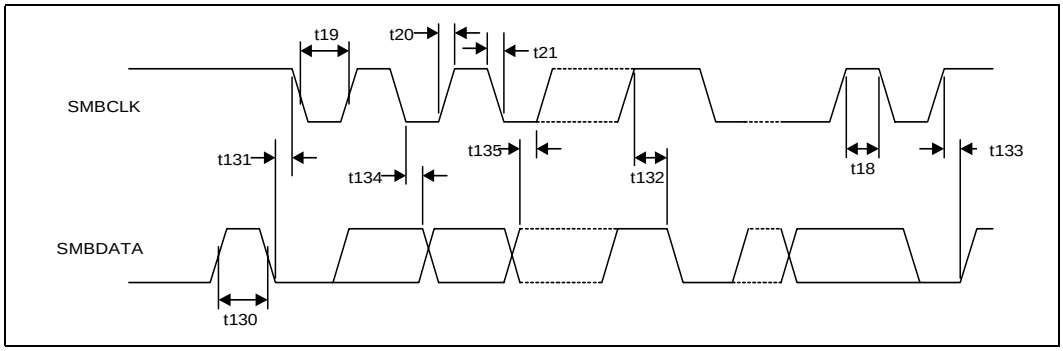
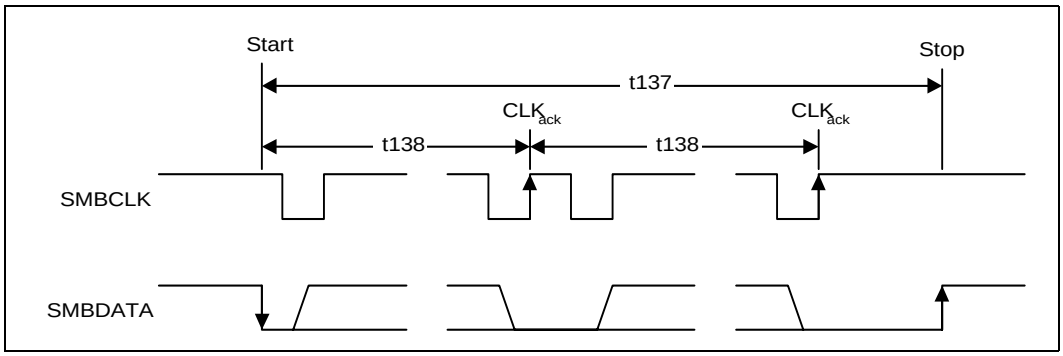


Figure 60. SMBus Timeout



22.5.6 Power and Reset

Figure 61. Power Sequencing and Reset Signal Timings

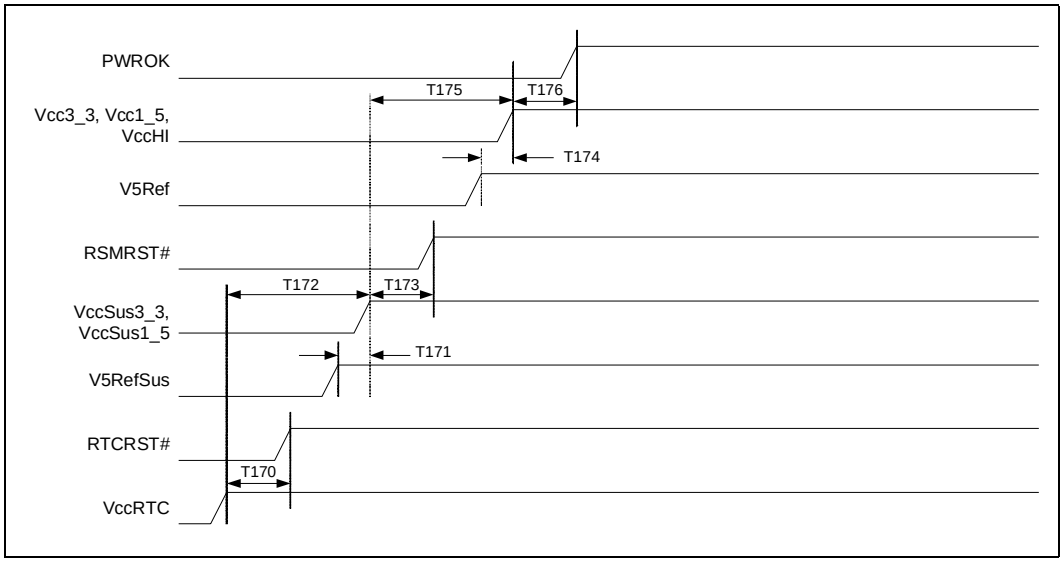


Figure 62. G3 (Mechanical Off) to S0 Timings

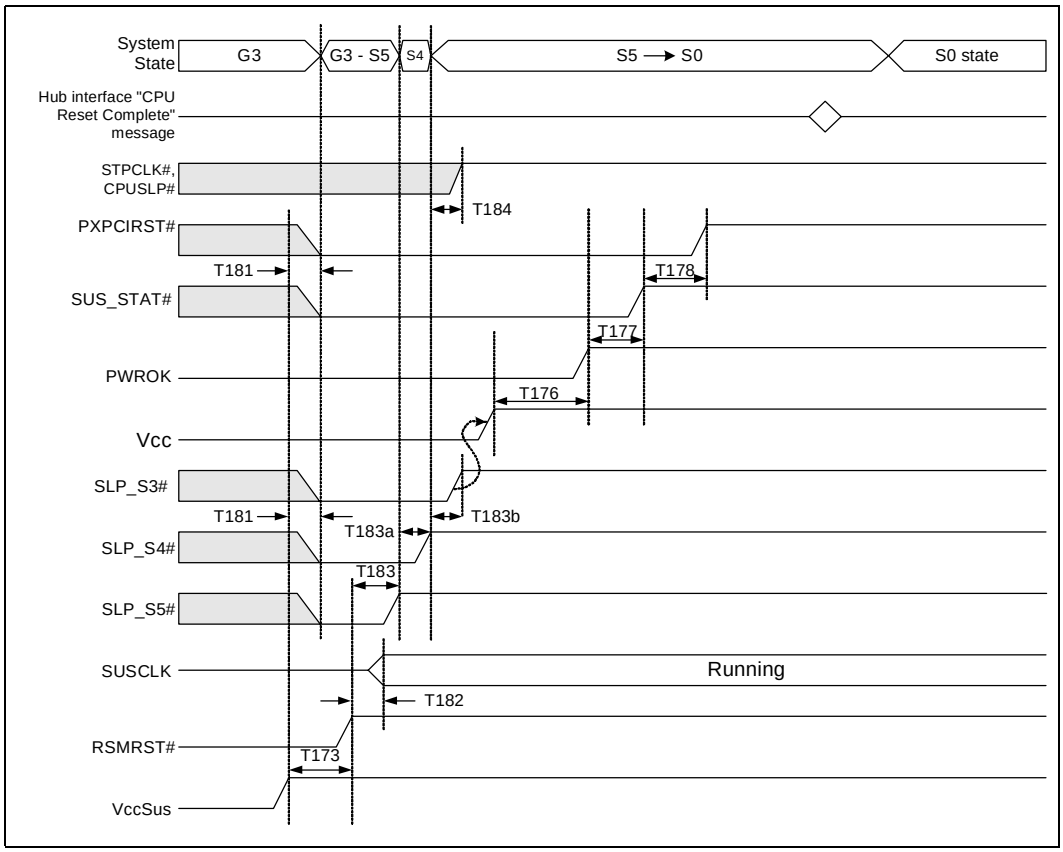


Figure 63. S0 to S1 to S0 Timing

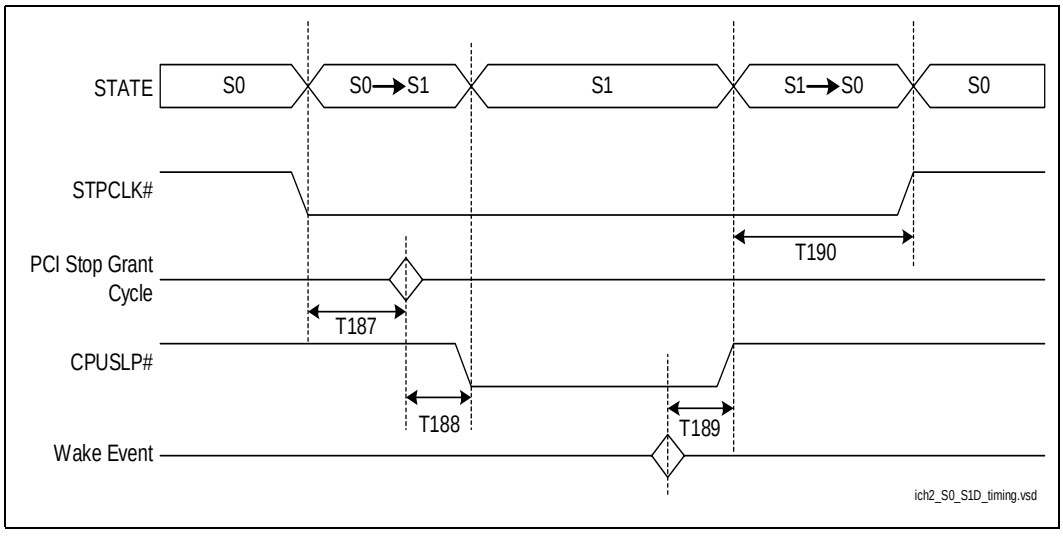
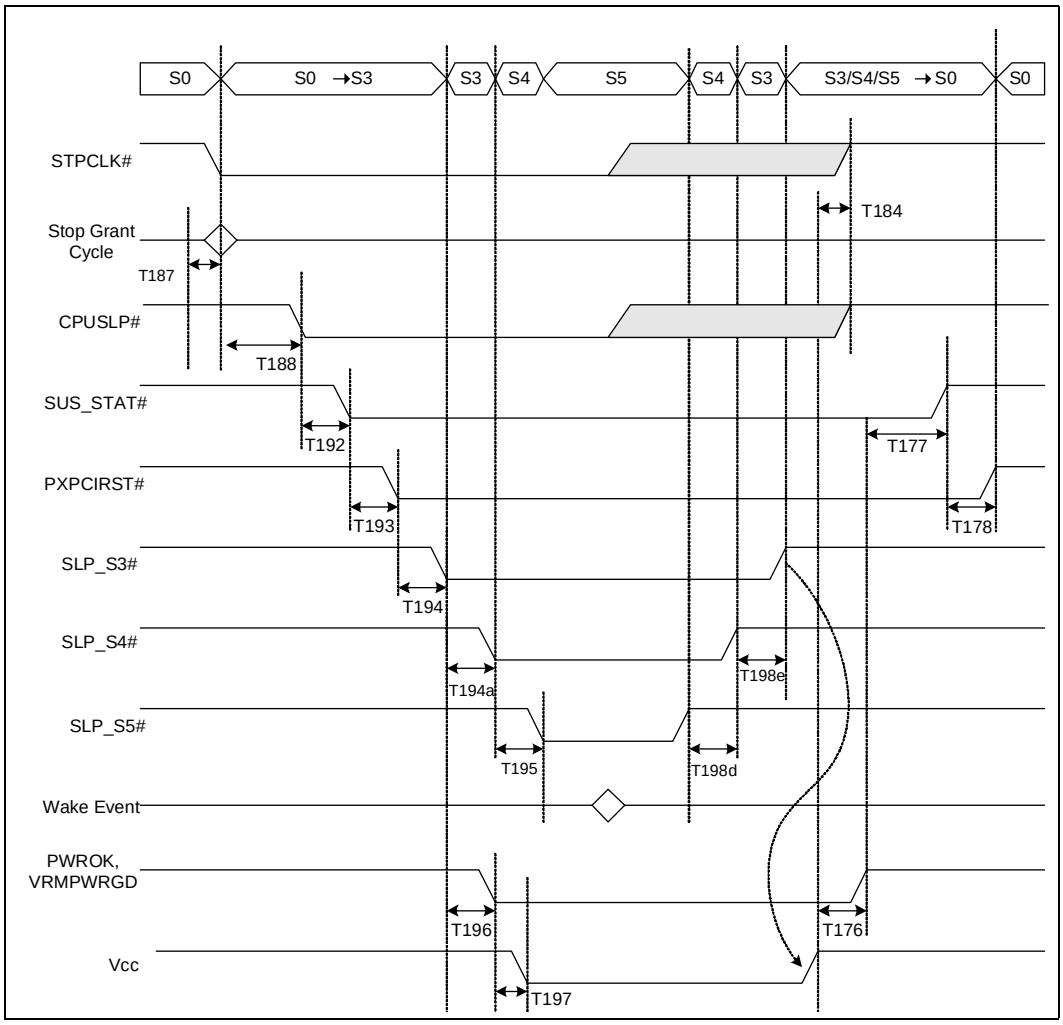
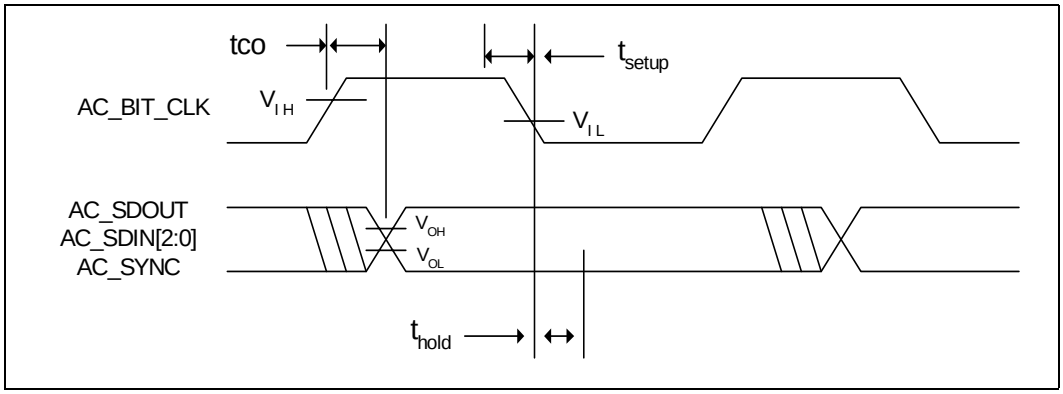


Figure 64. S0 to S5 to S0 Timings



22.5.7 AC'97 and Miscellaneous

Figure 65. AC'97 Data Input and Output Timings



Testability 23

23.1 Test Mode Description

The Intel® 6300ESB ICH supports two types of test modes, a tri-state test mode and a XOR Chain test mode. Driving RTCRST# low for a specific number of PCI clocks while PWROK is high will activate a particular test mode as described in Table 731.

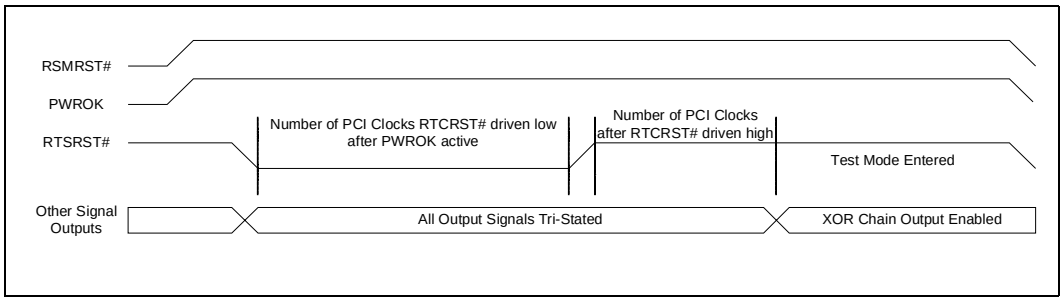
Note: RTCRST# can be driven low any time after PXPCIRST# is inactive.

Table 731. Test Mode Selection

Number of PCI Clocks RTCRST# driven low after PWROK active	Number of PCI Clocks after RTCRST# driven high	Test Mode
<4	N/A	No Test Mode Selected
4	3	XOR Chain 1
5	3	XOR Chain 2
6	3	XOR Chain 3
7	3	XOR Chain 4
8	3	All “Z”
9 - 13	N/A	Reserved. DO NOT ATTEMPT
14	3	Long XOR
15 - 42	N/A	Reserved. DO NOT ATTEMPT
43 - 51	N/A	No Test Mode Selected
52	3	XOR Chain 6
53	3	XOR Chain 4 Bandgap
59	3	XOR Chain 5
60	3	XOR Chain 7
>60	N/A	No Test Mode Selected

Figure 66 illustrates entry into a test mode. A particular test mode is entered upon the rising edge of the RTCRST# after being asserted for a specific number of PCI clocks while PWROK is active. To change test modes, the same sequence should be followed again. To restore the Intel® 6300ESB ICH to normal operation, execute the sequence with RTCRST# being asserted so that no test mode is selected as specified in Table 731.

Figure 66. Test Mode Entry (XOR Chain Example)



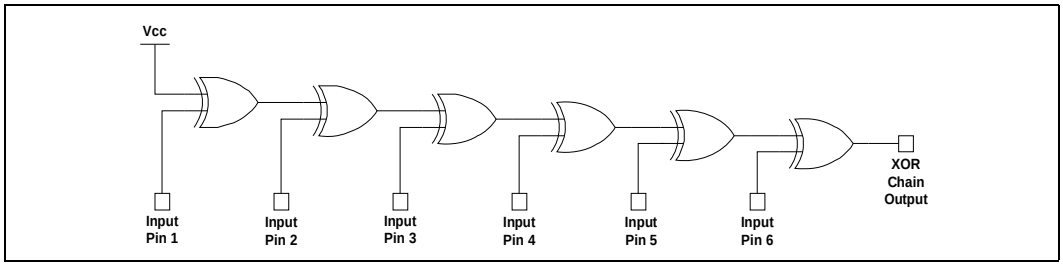
23.2 Tri-State Mode

When in the tri-state mode, all outputs and bi-directional pin are tri-stated, including the XOR Chain outputs.

23.3 XOR Chain Mode

In the Intel® 6300ESB ICH, provisions for Automated Test Equipment (ATE) board level testing are implemented with XOR Chains. The Intel® 6300ESB ICH signals are grouped into seven independent XOR chains which are enabled individually. When an XOR chain is enabled, all output and bi-directional buffers within that chain are tri-stated, except for the XOR chain output. Every signal in the enabled XOR chain (except for the XOR chain's output) functions as an input. All output and bi-directional buffers for pins not in the selected XOR chain are tri-stated. Figure 67 is a schematic example of XOR chain circuitry.

Figure 67. Example XOR Chain Circuitry



23.3.1 XOR Chain Testability Algorithm Example

XOR chain testing allows motherboard manufacturers to check component connectivity (e.g., opens and shorts to V_{CC} or GND). An example algorithm to do this is shown in Table 732.

Table 732. XOR Test Pattern Example

Vector	Input Pin 1	Input Pin 2	Input Pin 3	Input Pin 4	Input Pin 5	Input Pin 6	XOR Output
1	0	0	0	0	0	0	**1**
2	1	0	0	0	0	0	**0**
3	1	1	0	0	0	0	**1**
4	1	1	1	0	0	0	**0**
5	1	1	1	1	0	0	**1**
6	1	1	1	1	1	0	**0**
7	1	1	1	1	1	1	**1**

In this example, Vector 1 applies all “0s” to the chain inputs. The outputs being non-inverting, will consistently produce a “1” at the XOR output on a good board. One short to Vcc (or open floating to Vcc) will result in a “0” at the chain output, signaling a defect.

Likewise, applying Vector 7 (all “1s”) to the chain inputs (given that there are an even number of input signals in the chain), will consistently produce a “1” at the XOR chain output on a good board. One short to Vss (or open floating to Vss) will result in a “0” at the chain output, signaling a defect. It is important to note that the number of inputs pulled to “1” will affect the expected chain output value. If the number of chain inputs pulled to “1” is even, then expect “1” at the output. If the number of chain inputs pulled to “1” is odd, expect “0” at the output.

Continuing with the example in Table 732, as the input pins are driven to “1” across the chain in sequence, the XOR Output will toggle between “0” and “1.” Any break in the toggling sequence (e.g., “1011”) will identify the location of the short or open.

Table 733. XOR Chain #1 (RTCRST# asserted for 4 PCI clocks while PWROK active) (Sheet 1 of 2)

Pin Name	Ball #
PLOCK#	J6
C/BE[3]#	H5
PCICLK	J5
GPIO[33] / PXIRQ[0]#	G1
GPIO[34] / PXIRQ[1]#	H1
GPIO[35] / PXIRQ[2]#	H2
GPIO[36] / PXIRQ[3]#	J3
PXREQ[1]#	L6
GPIO[0] / PXREQ[2]#	J1
PXGNT0#	K1
PXIRDY#	R4
PXFRAME#	R6

Table 733. XOR Chain #1 (RTCRST# asserted for 4 PCI clocks while PWROK active) (Sheet 2 of 2)

Pin Name	Ball #
PXSTOP#	T2
PXTRDY#	T5
PXSERR#	U1
PXC/BE[1]#	U2
PXPAR	U4
PXPERR#	U6
PXC/BE[0]#	AA1
RASERR#	AE1
PXREQ64#	AE5
PXACK64#	AF4
OUTPUT IRQ[14]	AC23

Table 734. XOR Chain #2 (RTCRST# asserted for 5 PCI clocks while PWROK active) (Sheet 1 of 3)

Pin Name	Ball #
PDD[6]	AH24
PDD[4]	AJ25
PDD[7]	AC20
PDD[11]	AH25
PDD[5]	AD21
PDD[8]	AF23
PDD[9]	AB20
PDD[13]	AJ26
PDD[2]	AH26
PDD[3]	AD22
PDD[10]	AC22
PDDREQ	AF24
PDD[12]	AE23
PDD[14]	AE24
PDD[1]	AG25
PDD[15]	AH27
PDD[0]	AF25
PDIOR# (/PDWSTB / PRDMARDY#)	AB21
PIORDY (/PDRSTB / PWDMARDY#)	AF26
PDDACK#	AB22
PDIOW# (/PDSTOP)	AB23
IRQ[14]	AC23
PDA[1]	AG27
PDA[0]	AG28
PDCS1#	AF27
PDA[2]	AA23
PDCS3#	AD25
REQ[2]#	B11
REQ[3]#	D12
GNT[3]#	G13
GNT[2]#	C11
GPIO[2] / PIRQ[E]#	B10
GPIO[3] / PIRQ[F]#	A10

Table 734. XOR Chain #2 (RTCRST# asserted for 5 PCI clocks while PWROK active) (Sheet 2 of 3)

Pin Name	Ball #
PIRQ[C]#	A9
PIRQ[A]#	E12
GPIO[5] / PIRQ[H]#	A8
PIRQ[D]#	B8
PIRQ[B]#	A6
GPIO[4] / PIRQ[G]#	D10
REQ[1]#	G12
GNT[1]#	D9
AD[18]	B7
REQ[0]#	D8
AD[28]	H12
AD[15]	C7
GNT[0]#	E10
AD[22]	B6
AD[30]	G11
AD[20]	A5
AD[16]	A4
AD[4]	F10
AD[24]	E8
AD[0]	H11
STOP#	B5
AD[11]	B4
AD[26]	E7
AD[6]	D6
TRDY#	F9
FRAME#	G10
AD[7]	B3
AD[9]	D5
AD[2]	C3
PAR	G9
AD[5]	D4
AD[13]	F6
AD[1]	G8
SERR#	E4
C/BE[0]#	C2

Table 734. XOR Chain #2 (RTCRST# asserted for 5 PCI clocks while PWROK active) (Sheet 3 of 3)

Pin Name	Ball #
C/BE[1]#	G7
AD[8]	H8
AD[3]	F5
IRDY#	E3
PERR#	D2
AD[14]	D1
AD[12]	G6
AD[10]	E2
AD[23]	H7
C/BE[2]#	F4
DEVSEL#	G5
AD[17]	E1
AD[19]	F2
AD[21]	K8
AD[25]	K7
AD[27]	F1
AD[29]	H4
AD[31]	G2
OUTPUT FERR#	AA29

Table 735. XOR Chain #3 (RTCRST# asserted for 6 PCI clocks while PWROK active) (Sheet 1 of 4)

Pin Name	Ball #
IRQ[15]	V23
VRMPWRGD	Y26
A20GATE	AB29
RCIN#	V26
THRMTRIP#	AA28
FERR#	AA29
A20M#	W27
INTR	Y28
NMI	Y29
IGNNE#	U24
INIT#	W28

Table 735. XOR Chain #3 (RTCRST# asserted for 6 PCI clocks while PWROK active) (Sheet 2 of 4)

Pin Name	Ball #
STPCLK#	W29
SMI#	V29
CPUSLP#	V28
HL6	T26
HL5	U28
HL7	R24
HL4	U29
HLCOMP	T27
HI_STB/HI_STBS	T28
HI_STB#/HI_STBF	T29
HL3	R28
HL2	R29
HL1	P28
HL0	P29
HL10	N28
HL8	N29
HL9	N25
HL11	M28
NC	M29
NC	L29
NC	L28
NC	M25
NC	K29
CLK14	L26
GPIO[32] / WDT _TOUT#	M24
SPKR	K27
SATALED#	J28
GPIO[40]	H29
GPIO[18]	J27
GPIO[19]	L24
GPIO[20]	H28
GPIO[21]	K25
GPIO[23]	L22
GPIO[42]	G29
GPIO[38]	F29
GPIO[43]	K24
GPIO[41]	H26
NC	G28

Table 735. XOR Chain #3 (RTCRST# asserted for 6 PCI clocks while PWROK active) (Sheet 3 of 4)

Pin Name	Ball #
NC	J23
NC	H25
NC	F28
NC	G27
NC	F27
NC	H24
NC	G25
AC_BIT_CLK	G19
AC_SYNC	F20
AC_SDOUT	E20
NC	G18
GPIO[6]	C21
GPIO[7]	B22
NC	A22
NC	B21
NC	A21
NC	D19
GPIO[39]	E19
GPIO[37]	D20
NC	B20
SERIRQ	G17
SIU1_RXD	D18
SIU1_TXD	C19
SIU1_CTS#	A20
SIU1_DSR#	B19
SIU1_DCD#	A19
SIU1_RI#	D17
SIU1_DTR#	B18
SIU1_RTS#	G16
UART_CLK	A18
SIU0_RXD	B17
SIU0_TXD	A17
SIU0_CTS#	D16
SIU0_DSR#	E17
SIU0_DCD#	B16
SIU0_RI#	A16
SIU0_DTR#	E15
SIU0_RTS#	F16
LAD[0]	C15

Table 735. XOR Chain #3 (RTCRST# asserted for 6 PCI clocks while PWROK active) (Sheet 4 of 4)

Pin Name	Ball #
LAD[1]	B15
LDRQ[0]#	F14
LAD[2]	A15
LAD[3]	A14
LDRQ[1]#	G14
LFRAME#	B14
THRM#	D14
NC	B13
NC	A13
NC	F13
NC	B12
NC	A11
NC	E13
OUTPUT IRQ[14]	AC23

Table 736. XOR Chain #4 (RTCRST# asserted for 7 PCI clocks while PWROK active) (Sheet 1 of 2)

Pin Name	Ball #
PXPCIRST#	AH8
GPIO[25]	AH9
GPIO[8]	AE12
GPIO[12]	AF11
NC	AD13
GPIO[27]	AG10
GPIO[13]	AC14
PME#	AJ8
RI#	AH10
GPIO[28]	AJ9
SLP_S4#	AD14
PWRBTN#	AJ10
SYSRESET#	AH11
SLP_S5#	AJ11
SMLINK[1]	AH12
GPIO[24]	AE14
SUSCLK	AJ12
SUS_STAT#	AD15
SMLINK[0]	AH13
SLP_S3#	AG14
SMBDATA	AJ13
SMBCLK	AH14
GPIO[11] / SMBALERT#	AF15
NC	AJ14
NC	AJ15
OC[0]#	F24
OC[2]#	G23
OC[1]#	F23
OC[3]#	E24
AC_SDIN0	F22
AC_RST#	E23
AC_SDIN2	E22
AC_SDIN1	E21

Table 736. XOR Chain #4 (RTCRST# asserted for 7 PCI clocks while PWROK active) (Sheet 2 of 2)

Pin Name	Ball #
GPIO[56]	C22
GPIO[57]	D22
OUTPUT FERR#	AA29

Table 737. XOR Chain #5 (RTCRST# asserted for 59 PCI clocks while PWROK active) (Sheet 1 of 4)

Pin Name	Ball #
PXPCLKO[0]	J2
PXPCLKO[1]	L8
PXPCLKO[2]	K6
PXPCLKO[3]	K4
PXPCLKO[4]	L5
PXAD[31]	K2
PXAD[29]	M7
PXAD[28]	L3
PXREQ[0]#	L2
PXAD[30]	M5
PXAD[27]	L1
PXAD[26]	M2
PXAD[25]	M1
PXC/BE[3]#	N6
PXAD[24]	N4
PXAD[23]	N3
PXAD[20]	N2
PXAD[22]	N1
PXAD[21]	P7
PXAD[19]	P5
PXAD[18]	P2
PXAD[17]	P1
PXAD[16]	R1
PXC/BE[2]#	R2
PXPLOCK#	T1
PXDEVSEL#	T3

Table 737. XOR Chain #5 (RTCRST# asserted for 59 PCI clocks while PWROK active) (Sheet 2 of 4)

Pin Name	Ball #
PXAD[15]	T7
PXAD[12]	V1
PXAD[11]	V2
PXAD[13]	V4
PXAD[8]	W1
PXAD[9]	W2
PXAD[14]	Y1
PXM66EN	W4
PXAD[6]	Y2
PXAD[2]	V7
PXAD[4]	AA2
PXAD[3]	Y4
PXAD[47]	AB1
PXAD[10]	AA4
PXAD[45]	AB2
PXAD[1]	W7
PXAD[43]	Y6
PXAD[46]	AC1
PXAD[42]	AB3
PXAD[5]	W8
GPIO[16] / PXGNT[2]#	AC2
GPIO[17] / PXGNT[3]#	Y7
PXAD[41]	AD1
PXAD[7]	AA6
PXAD[40]	AD2
GPIO[1] / PXREQ[3]#	Y8
PXGNT1#	AB5
PXPCLKI	AC4
PXPCICLK	AC5
PCIXSBRST#	AA7
PXAD[44]	AB7
PXAD[35]	AE2
PXAD[38]	AD4
PXAD[32]	AF1
PXAD[34]	AE3

Table 737. XOR Chain #5 (RTCRST# asserted for 59 PCI clocks while PWROK active) (Sheet 3 of 4)

Pin Name	Ball #
PXAD[39]	AC6
PXAD[0]	AF2
PXAD[54]	AB9
PXAD[33]	AE4
PXAD[37]	AD6
PXC/BE[5]#	AG2
PXAD[36]	AD7
PXAD[63]	AH3
PXPAR64	AD8
PXAD[50]	AC10
PXC/BE[4]#	AE7
PXC/BE[7]#	AF6
PXAD[60]	AG5
PXAD[62]	AH4
PXAD[59]	AH5
PXAD[48]	AD10
PXC/BE[6]#	AF7
PXAD[51]	AE9
PXAD[61]	AJ4
PXAD[57]	AH6
PXAD[58]	AJ5
PXAD[56]	AJ6
PXAD[53]	AF9
PXAD[49]	AE10
PXAD[55]	AH7
PXAD[52]	AJ7
SDD[7]	AF28
SDD[9]	AE26
SDD[5]	AF29
SDD[8]	Y23
SDD[3]	AD26
SDD[6]	W22
SDD[10]	AB24
SDD[12]	AE28
SDD[11]	AC26

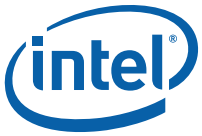


Table 737. XOR Chain #5 (RTCRST# asserted for 59 PCI clocks while PWROK active) (Sheet 4 of 4)

Pin Name	Ball #
SDD[1]	AE29
SDD[4]	AA24
SDD[13]	AB25
SDDREQ	AD28
SDD[14]	AD27
SDDACK#	Y24
SIORDY / (SDRSTB/ SWDMARDY#)	W24
SDA[1]	AD29
SDD[15]	AC28
SDD[2]	AB27
SDA[2]	AC29
SDIOW# / (SDSTOP)	Y25
SDD[0]	AA26
SDCS1#	AB28
SDIOR# / (SDWSTB/ PRDMARDY#)	W25
SDCS3#	AA27
SDA[0]	V24
OUTPUT IRQ[15]	V23

Table 738. XOR Chain #6 (RTCRST# asserted for 52 PCI clocks while PWROK active)

Pin Name	Ball #
INTRUDER#	AG23
RTCX1	AE21
OUTPUT IRQ[15]	V23

Table 739. XOR Chain #7 (RTCRST# asserted for 60 PCI clocks while PWROK active)

Pin Name	Ball #
SATACLKN	AJ17
SATACLKP	AH17
SATA[0]RXN	AJ19
SATA[0]RXP	AH19
SATA[0]TXN	AG18
SATA[0]TXP	AF18
SATA[1]RXN	AJ21
SATA[1]RXP	AH21
SATA[1]TXN	AG20
SATA[1]TXP	AF20
SATARBIASN	AJ23
SATARBIASP	AH23
OUTPUT LDRQ[1]#	G14

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

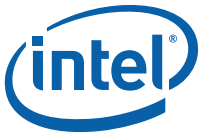
P

R

S

T

U



V

W

November 2007
Order Number: 300641-004US