SCH5636

Desktop Embedded Controller with Fan Control, Hardware Monitoring and PECI

Highlights

- High Performance 32-bit ARC-625D Embedded Controller (EC)
  - 96KB Closely Coupled Instruction ROM
  - 16 KB Single Cycle 32-bit Wide Dual-ported SRAM, Accessible as Closely Coupled Data Memory and Instruction Memory
  - 32 x 32 ⟶ 64 Multiply
  - Maskable Interrupt Aggregator Interface
  - Maskable Hardware Wake-Up Events
  - Idle and Sleep modes
  - JTAG Debug Port
  - MCU Serial Debug Port
  - 5-Channel DMA Interface
- ACPI 2.0 Compliant
- PC2001 Compliant
- LPC Interface
  - Supports LPC Bus frequencies of 19MHz to 33MHz
  - Multiplexed Command, Address and Data Bus
  - Serial IRQ Interface Compatible with Serialized IRQ Support for PCI Systems
  - PME Interface
- 3.3-Volt I/O
- 128-pin QFP RoHS Compliant Package
- Three Programmable 16-bit Timers
- 32-bit Performance Timer
- System Watch Dog Timer (WDT)
- Battery Backed Resources
  - Power-Fail Status Register
  - VBat backed 64 byte memory
- Two EC-based SMBus 2.0 Host Controllers
  - Allows Master or Dual Slave Operation
  - Controllers are Fully Operational on Standby Power
  - DMA-driven I^2C Network Layer
  - I^2C Datalink Compatibility Mode
  - Multi-Master Capable
  - Supports Clock Stretching
  - Programmable Bus Speeds
  - 400 KHz Capable
  - Hardware Bus Access “Fairness” Interface
  - Detects SMBus Time-outs
  - One controller can be multiplexed onto a low voltage SMBus
- PECI Interface 2.0
  - Supports PECI REQUEST# and PECI READY signaling
  - Supports up to 2 CPUs and 4 domains
- Temperature reading from PCH over SMBus
- Temperature reading from AMD-TSI over SMBus
- Temperature Monitor
  - Monitoring up to 2 Remote Thermal Diodes plus an Anti-Parallel Remote Thermal Diode
  - Built-in ADC supports temperature readings from -63 degrees Celsius to +192 degrees Celsius
    - Supports monitoring of discrete diodes (3904 type diodes)
    - Supports monitoring substrate diodes (45nm & 65nm processor diodes)
  - Temperature resolution is 0.125 degrees Celsius
  - Internal Ambient Temperature Measurement
  - Out-of-limit Temperature Event reporting
- Bi-directional PROCHOT# Pin
  - Interrupt generation for PROCHOT Assert events
  - May be used by AMTA and PTTA features to adjust fan control limits
  - May be configured to force fans on full
  - Supports PROCHOT Assertions to external CPU
  - Supports PROCHOT Throttle Events to external CPU
  - Supports Interrupt Event to Host
- Voltage Monitor
  - Monitoring VBAT, VTR,VCC and Vtt power supplies
  - Monitoring of one external voltage
  - Limit comparison on monitored values
- PWM (Pulse width Modulation) Outputs (4)
  - Multiple Clock Rates
  - 16-bit ON and 16-bit OFF Counters

- Fan tachometer Inputs (4)
  - Programmable to monitor standard tachometer outputs or locked rotor alarm outputs
  - Generate tachometer event when speed of fan drops below programmed limit
- Internal clock sources
  - A Ring Oscillator generates 64 MHz clock
  - SIO clocks derived from a 96MHz PLL synchronized to a 14.318MHz clock input
  - Main ring generates 32kHz standby clock when external 32.768KHz clock source is off
- Low Battery Warning
- LED Control
  - Two LEDs to indicate system state
- Programmable Wake-up Event Interface
- General Purpose Input/Output Pins (60 total)
- System Management Interrupt (SMI)
- GLUE Logic
  - 4 Buffered PCI Reset Outputs
  - Power OK Signal Generation
  - Power Sequencing
  - Power Supply Turn On Circuitry
  - Resume Reset Signal Generation
  - Speaker output
  - Intrusion Detection
- 2.88MB Super I/O Floppy Disk Controller
  - Licensed CMOS 765B Floppy Disk Controller
  - Software and Register Compatible with Microchip's Proprietary 82077AA Compatible Core
  - Configurable Open Drain/Push-Pull Output Drivers
  - Supports Vertical Recording Format
  - 16-Byte Data FIFO
  - 100% IBM® Compatibility
  - Detects All Overrun and Underrun Conditions
  - Sophisticated Power Control Circuitry (PCC) Including Multiple Powerdown Modes for Reduced Power Consumption
  - DMA Enable Logic
  - Data Rate and Drive Control Registers
  - 480 Address, Up to Eight IRQ and Four DMA Options
- Enhanced Digital Data Separator
  - 1 Mbps, 500 Kbps, 300 Kbps, 250 Kbps Data Rates
  - Programmable Precompensation Modes
- Keyboard Controller
  - 8042 Software Compatible
  - 8 Bit Microcomputer
  - 2k Bytes of Program ROM
  - 256 Bytes of Data RAM
  - Four Open Drain Outputs Dedicated for Keyboard/Mouse Interface
  - Asynchronous Access to Two Data Registers and One Status Register
  - Supports Interrupt and Polling Access
  - 8 Bit Counter Timer
  - Port 92 Support
  - Fast Gate A20 and KRESET Outputs
- Serial Ports
  - Two Full Function Serial Ports
  - High Speed NS16C550A Compatible UARTs with Send/Receive 16-Byte FIFOs
  - Programmable Baud Rate Generator
  - Modem Control Circuitry
  - Any LPC Address Configurable. 15 IRQ Options
- Multi-Mode™ Parallel Port with ChiProtect™
  - Standard Mode IBM PC/XT®, PC/AT®, and PS/2™ Compatible Bi-directional Parallel Port
  - Enhanced Parallel Port (EPP) Compatible - EPP 1.7 and EPP 1.9 (IEEE 1284 Compliant)
  - IEEE 1284 Compliant Enhanced Capabilities Port (ECP)
  - ChiProtect Circuitry for Protection
  - 960 Addresses, Up to 15 IRQ and Four DMA Options

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include -literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

Table of Contents

1.0 INTRODUCTION

The SCH5636 is a 3.3V PC 2001 compliant Super I/O controller with an LPC interface. All legacy drivers used for Super I/O components are supported making this interface transparent to the supporting software. The LPC bus also supports power management, such as wake-up and sleep modes.

The SCH5636 provides temperature monitoring with auto fan control. The temperature monitor is capable of monitoring up to three external diodes, one internal ambient temperature sensor or retrieving temperatures from external processors that implement the PECI Interface. This device offers programmable automatic fan control support based on one or more of these measured temperatures. There are four pulse width modulation (PWM) outputs with high frequency support as well as four fan tachometer inputs. In addition, there is support for a bi-directional PROCHOT# pin that may be used to generate an interrupt, adjust the programmed temperature limits in the auto fan control logic, or force the PWM outputs on full. The RRCC feature provides a linear relationship of temperature to fan speed.

The Glue Logic includes various power management logic; including generation of RSMRST# and Power OK signal generation. There are also two LEDs to indicate power status. The part also provides a low battery warning circuit.

The SCH5636 provides 60 General Purpose I/O control pins, which offer flexibility to the system designer.

The SCH5636 incorporates the following Super I/O components: a parallel port that is compatible with IBM PC/AT architecture, as well as the IEEE 1284 EPP and ECP; two serial ports that are 16C550A UART compatible; a keyboard/mouse controller that uses an 8042 micro controller; and a floppy disk controller.

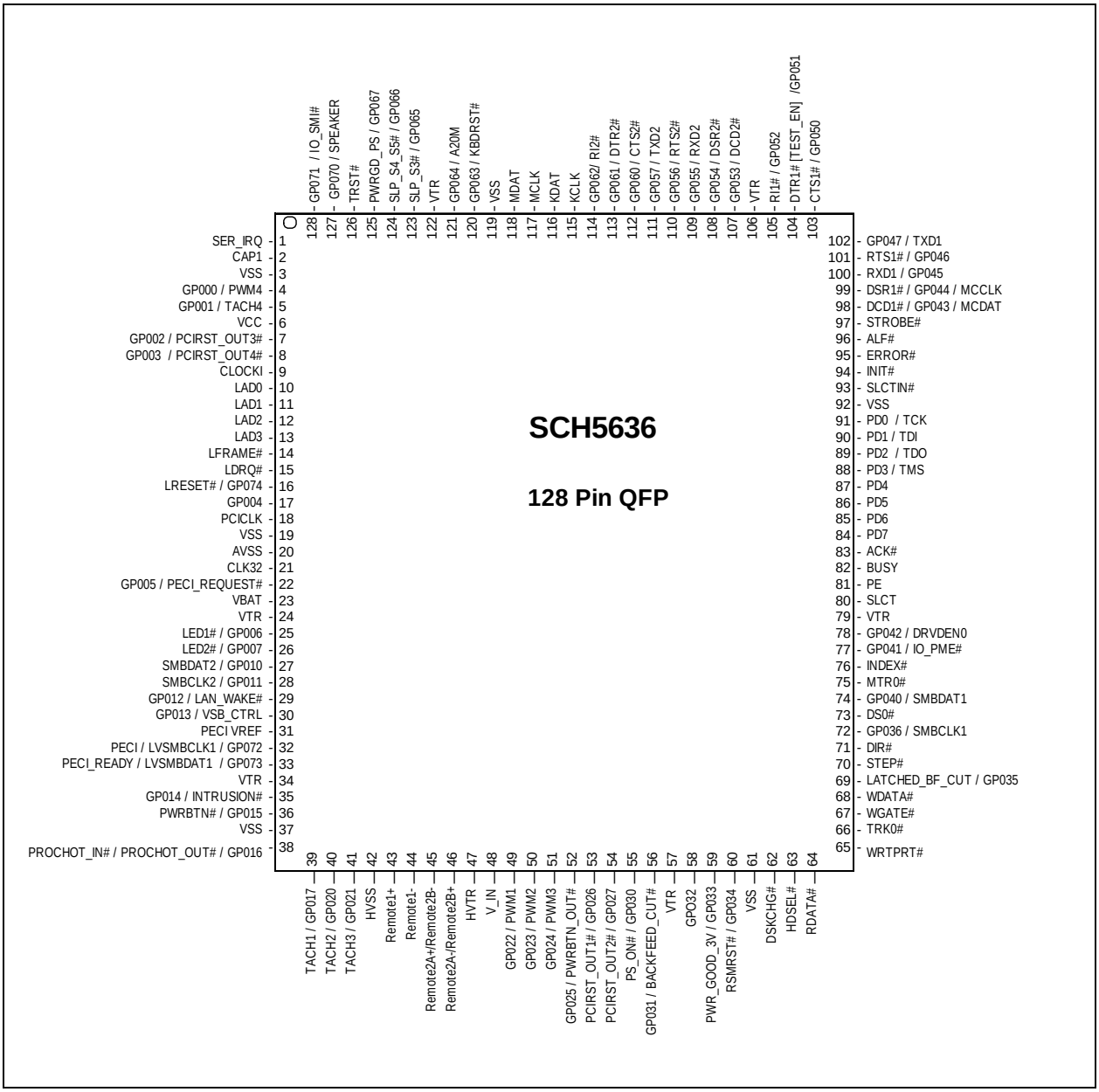
The SCH5636 is ACPI 1.0b/2.0 compatible and supports multiple low power-down modes. It incorporates sophisticated power control circuitry (PCC), which includes keyboard and mouse wake-up events.

The SCH5636 incorporates a high-performance embedded microcontroller. The SCH5636 communicates with the system host using the Intel® Low Pin Count bus.

The SCH5636 is directly powered by two separate suspend supply planes (VBAT and VTR) and senses a third runtime power plane (VCC) to provide “instant on” and system power management functions. The SCH5636 also contains an integrated VTR Reset Generator and a system power management interface that supports low-power states and can drive state changes as a result of hardware wake events as defined by the SCH5636 wake interface.

2.0 PIN LAYOUT

FIGURE 2-1: SCH5636 PIN DIAGRAM



2.1 Changes from SCH5617

The following table lists pinout changes from the SCH5617. Because GPIO names have changed for all GPIOs, GPIO name changes have not been listed in the DIFFERENCE column. In addition, there is no distinction between Host and 8051 GPIOs; all GPIO pins can be accessed by either the EC directly or by the Host indirectly through the EMI.

TABLE 2-1: PIN CHANGES FROM THE SCH5617

Pin #	SCH5636 Pin Name	SCH5617 Pin Name	Difference
1	SER_IRQ	SER_IRQ	
2	CAP1	CAP1	
3	VSS	VSS	
4	GP000 / PWM4	GP8051_1 / SMB_DATA_5V	SMB_DATA_5V eliminated
5	GP001) / TACH4	GP8051_3 / SMB_DATA_2P5V	SMB_DATA_2P5V eliminated
6	VCC	VCC	
7	GP002 / PCIRST_OUT3#	GP8051_2 / SMB_CLK_5V	SMB_CLK_5V eliminated
8	GP003 / PCIRST_OUT4#	GP8051_4 / SMB_CLK_5V	SMB_CLK_2P5V eliminated
9	CLOCKI	CLOCKI	
10	LAD0	LAD0	
11	LAD1	LAD1	
12	LAD2	LAD2	
13	LAD3	LAD3	
14	LFRAME#	LFRAME#	
15	LDRQ#	LDRQ#	
16	LRESET# / GP074	LRESET#	
17	SPI_DO / GP004	LPCPD#	LPCPD# eliminated; SPI and a GPIO added
18	PCICLK	PCICLK	
19	VSS	VSS	
20	AVSS	AVSS	
21	CLK32	LATCHED_BF_CUT/GP23/ GP8051_18	LATCHED_BF_CUT and GPIO eliminated (LATCHED_BF_CUT moved to pin 69) 32K clock added
22	GP005 / PECI_REQUEST#	TEST	.TEST eliminated, PECI_REQUEST and GPIO added
23	VBAT	VBAT	
24	VTR	VTR	
25	LED1 / GP006	LED3 / GP60	
26	LED2 / GP007	LED2 / GP60	
27	SMBDAT2 / GP010	SDAT_1 / GP42/ IO_SMI#	SMB Isolation eliminated, IO_SMI# eliminated (moved to pin 128); SMB Dat added
28	SMBCLK2 / GP011	SDAT/ GP35 / LED1	SMB Isolation eliminated; SMB Clock added LED1 eliminated
29	SPI_DI / GP012	SCLK_1 / GP26	SMB Isolation eliminated; SPI added

TABLE 2-1: PIN CHANGES FROM THE SCH5617 (CONTINUED)

Pin #	SCH5636 Pin Name	SCH5617 Pin Name	Difference
30	SPI_CK / GP013	SCLK / GP25	SMB Isolation eliminated; SPI added
31	PECI VREF	PECI VREF	
32	PECI / LVSMBCLK1 / GP072	PECI	LVSMBCLK1 added
33	PECI_READY / LVSMBDAT1 / GP073	PECI READY	LVSMBCLK1 added
34	VTR	VTR	
35	GP014 / INTRUSION	PROCHOT_OUT	PROCHOT_OUT eliminated (PROCHOT_IN#, pin 38, now bidirectional); GPIO added, INTRUSION added;
36	GP015	GP8051_47 / GP33	
37	VSS	VSS	
38	PROCHOT_IN# / PROCHOT_OUT# / GP016	PROCHOT_I#	PROCHOT function now bidirectional
39	TACH1 / GP017	TACH1	GPIO added
40	TACH2 / GP020	TACH2	GPIO added
41	TACH3 / GP021	TACH3	GPIO added
42	HVSS	HVSS	
43	Remote1+	Remote1+	
44	Remote1-	Remote1-	
45	Remote2+	Remote2+	
46	Remote2-	Remote2-	
47	HVTR	HVTR	
48	V_IN	VIN1	
49	GP022 / PWM1	PWM1	GPIO added
50	GP023 / PWM2	PWM2	GPIO added
51	GP024 / PWM3	PWM3	GPIO added
52	GP025 / WDT_OUT#	IDE_RSTDRV# / GP75	WDT_OUT# added IDE_RSTDRV# deleted
53	PCIRST_OUT1# / GP026	PCI_RST_SYS# / GP76	
54	PCIRST_OUT2# / GP027	PCI_RST_SLOTS# / GP77	
55	PS_ON#/ / GP030	PS_ON# / GP80	
56	GP031 / BACKFEED_CUT#	BACKFEED_CUT# / GP81	
57	VTR	VTR	
58	SPI_CS# / GPO32	GP82	SPI added
59	PWR_GOOD_3V / GP033	PWR_GOOD_3V / GP83	
60	RSMRST# / GP034	RSMRST# / GP84	
61	VSS	VSS	
62	DSKCHG#	DSKCHG#	
63	HDSEL#	HDSEL#	
64	RDATA#	RDATA#	
65	WRTPRT#	WRTPRT#	
66	TRK0#	TRK0#	
67	WGATE#	WGATE#	

TABLE 2-1: PIN CHANGES FROM THE SCH5617 (CONTINUED)

Pin #	SCH5636 Pin Name	SCH5617 Pin Name	Difference
68	WDATA#	WDATA#	
69	LATCHED_BF_CUT / GP035	CAP2	CAP2 eliminated
70	STEP#	STEP#	
71	DIR#	DIR#	
72	GP036 / SMBCLK1	GP22 / P12 / MRT1# / SCSI#	P12 / MTR1# / SCSI# eliminated
73	DS0#	DS0#	
74	GP040 / SMBDAT1	GP21/ P16 / DS1#	P16 / DS1# eliminated
75	MTR0#	MTR0#	
76	INDEX#	INDEX#	
77	GP041 / IO_PME#	GP41 / IO_PME#	
78	GP042 / DRVDEN0	GP40 / DRVDEN0	
79	VTR	VTR	
80	SLCT	SLCT/KDAT	KDAT eliminated
81	PE	PE/KCLK	KCLK eliminated
82	BUSY	BUSY/FALE1	Flash debug function eliminated
83	ACK#	ACK#/FALE0	Flash debug function eliminated
84	PD7	PD7/FD7_FA7	Flash debug function eliminated
85	PD6	PD6/FD6_FA6	Flash debug function eliminated
86	PD5	PD5/FD5_FA5	Flash debug function eliminated
87	PD4	PD4/FD4_FA4	Flash debug function eliminated
88	PD3 / TMS	PD3/FD3_FA3	Flash debug function eliminated; JTAG function added
89	PD2 / TDO	PD2/FD2_FA2	Flash debug function eliminated; JTAG function added
90	PD1 / TDI	PD1/FD1_FA1	Flash debug function eliminated; JTAG function added
91	PD0 / TCK	PD0/FD0_FA0	Flash debug function eliminated; JTAG function added
92	VSS	VSS	
93	SLCTIN#	SLCTIN# / FWR#	Flash debug function eliminated
94	INIT#	INIT# / FRD#	Flash debug function eliminated
95	ERROR#	ERROR#/FPGM	Flash debug function eliminated
96	ALF#	ALF#/MCLK/ FCS#	Flash debug function eliminated, MCLK eliminated
97	STROBE#	STROBE#/MDAT	MDAT eliminated
98	DCD1# / GP043 / MCDAT	DCD1#/GP8051_10	MCU Debug added
99	DSR1# / GP044 / MCCLK	DSR1#/GP8051_11	MCU Debug added

TABLE 2-1: PIN CHANGES FROM THE SCH5617 (CONTINUED)

Pin #	SCH5636 Pin Name	SCH5617 Pin Name	Difference
100	RXD1 / GP045	RXD1/GP8051_12	
101	RTS1# / GP046	RTS1# [SYSOPT]	
102	GP047 / TXD1	GP8051_14 / TXD1	
103	CTS1# / GP050	CTS1# / GP8051_15	
104	DTR1# [TEST_EN] / GP051	DTR1# [FLASH_EN] / GP8051_16	FLASH_EN strap eliminated, TEST_EN strap added
105	RI1# / GP052	RI1# / GP8051_17	
106	VTR	VTR	
107	GP053 / DCD2#	DCD2# / GP8051_9	
108	GP054 / DSR2#	DSR2# / GP8051_8	
109	GP055 / RXD2	GP52 / RXD2	
110	GP056 / RTS2#	GP55 / RTS2# / DDRC	DDRC removed
111	GP057 / TXD2	GP53 / TXD2	
112	GP060 / CTS2#	CTS2# / GP8051_7	
113	GP 061 / DTR2#	GP57 / DTR2#	
114	GP062 / RI2#	RI2# / GP8051_6	
115	KCLK	KCLK	
116	KDAT	KDAT	
117	MCLK	MCLK	
118	MDAT	MDAT	
119	VSS	VSS	
120	GP063 / KBDRST#	GP36 / KBDRST#	
121	GP064 / A20M	GP37 / A20M	
122	VTR	VTR	
123	SLP_S3# / GP065	SLP_S3# / GP10	
124	SLP_S4_S5# / GP066	SLP_S4_S5# / GP11	
125	PWRGD_PS / GP067	PWRGD_PS	
126	TRST#	GP31 / SECONDARY_HD#	SECONDARY_HD# eliminated GPIO eliminated; JTAG function added
127	GP070 / SPEAKER	/ GP14 / HD_LED#	HD_LED# eliminated
128	GP071 / IO_SMI#	GP33 / PRIMARY_HD#	PRIMARY_HD# removed IO_SMI# moved to this pin

3.0 SIGNAL DESCRIPTIONS

In the following table, each row with multiple entries represents a pin with multiple configuration options. The first entry in the row is the default configuration after VTR power on. Except as noted, pins that default to GPIOs default to inputs. All pins that default to open drain outputs are tri-stated on VTR power on.

Note:	No GPIO pin should be left floating in a system. If a GPIO pin is not in use, it should be either tied high, tied low, or pulled to either power or ground through a resistor.

TABLE 3-1: SIGNAL DESCRIPTIONS

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
LPC INTERFACE (9)				
SER_IRQ	PCI_IO	Serial IRQ pin used with the PCI_CLK pin to transfer interrupts to the host.	GATE	Note 3-9 Note 3-19
LAD0	PCI_IO	Active high LPC signals used for multiplexed command, address and data bus.	GATE	Note 3-9
LAD1	PCI_IO	Active high LPC signals used for multiplexed command, address and data bus.	GATE	Note 3-9
LAD2	PCI_IO	Active high LPC signals used for multiplexed command, address and data bus.	GATE	Note 3-9
LAD3	PCI_IO	Active high LPC signals used for multiplexed command, address and data bus.	GATE	Note 3-9
LFRAME#	PCI_I	Active low signal indicates start of new cycle and termination of broken cycle.	GATE	Note 3-9
LDRQ#	PCI_IO	Active low signal used for encoded DMA/Bus Master request for the LPC interface.	GATE	Note 3-9
LRESET#	I	Active low signal used as LPC Interface Reset. PCICLK must be stable for at least 1ms before de-assertion of LRESET#	NO GATE	Note 3-19
GP074	IO4	GPIO	NO GATE	
PCICLK	PCI_CLK	PCI clock input.	GATE	Note 3-9 Note 3-19
HARDWARE MONITOR (13)				
PROCHOT_IN#	PECI_I	PROCHOT Input	GATE	Note 3-12 Note 3-17
PROCHOT_OUT#	OD_PH	PROCHOT Output	GATE	
GP016	LVIOD24	Low voltage GPIO	NO GATE	
TACH1	I_M	Input for monitoring a fan tachometer	NO GATE	Note 3-12
GP017	IO4	GPIO	NO GATE	
TACH2	I_M	Input for monitoring a fan tachometer	NO GATE	Note 3-12
GP020	IO4	GPIO	NO GATE	
TACH3	I_M	Input for monitoring a fan tachometer.	NO GATE	Note 3-12
GP021	IO4	GPIO	NO GATE	

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
Remote1+	I_{AN}	This is the positive input (current source) from the remote thermal diode. This serves as the positive input into the A/D.	NO GATE	Note 3-12
Remote1-	I_{AN}	This is the negative Analog input (current sink) from the remote thermal diode. This serves as the negative input into the A/D.	NO GATE	Note 3-12
Remote2A+/Remote2B-	I_{AN}	This is the positive input (current source) from the remote thermal diode. This serves as the positive input into the A/D This is also the negative input for an anti-parallel remote thermal diode.	NO GATE	Note 3-12
Remote2A-/Remote2B+	I_{AN}	This is the negative Analog input (current sink) from the remote thermal diode. This serves as the negative input into the A/D. This is also the positive input for an anti-parallel remote thermal diode.	NO GATE	Note 3-12
V_IN	I_{AN}	Voltage input to A/D. Requires external resistor divider network.	NO GATE	Note 3-12
GP022 PWM1	IO4 OD4	GPIO PWM Output 1 for controlling speed of fan.	NO GATE NO GATE	Note 3-2 Note 3-12
GP023 PWM2	IO4 OD4	GPIO PWM Output 2 for controlling speed of fan.	NO GATE NO GATE	Note 3-2 Note 3-12
GP024 PWM3	IO4 OD4	GPIO PWM Output 3 for controlling speed of fan.	NO GATE NO GATE	Note 3-2 Note 3-12
GP014 / INTRUDER#	IO4	GPIO Intruder detect. Battery-backed.	NO GATE	Note 3-2 Note 3-12 Note 3-15
PARALLEL PORT (17)				
SLCT	I	This high active input from the printer indicates that it has power on. Bit 4 of the Printer Status Register reads the SLCT input.	GATE	Note 3-19 Note 3-7
PE	I	Another status output from the printer, a high indicating that the printer is out of paper. Bit 5 of the Printer Status Register reads the PE input	GATE	Note 3-19 Note 3-7
BUSY	I	A low active output from the printer indicating that it has received the data and is ready to accept new data. Bit 6 of the Printer Status Register reads the ACK# input.	GATE	Note 3-19 Note 3-7

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
ACK#	I	A low active output from the printer indicating that it has received the data and is ready to accept new data. Bit 6 of the Printer Status Register reads the ACK# input.	GATE	Note 3-19 Note 3-7
PD7	IOP14	Port Data 7	GATE	Note 3-2 Note 3-3
PD6	IOP14	Port Data 6	GATE	Note 3-2 Note 3-3
PD5	IOP14	Port Data 5	GATE	Note 3-2 Note 3-3
PD4	IOP14	Port Data 4	GATE	Note 3-2 Note 3-3
PD3 TMS	IOP14 I	Port Data 3 JTAG Mode Select Input	GATE NO GATE	Note 3-2 Note 3-3 Note 3-23
PD2 TDO	IOP14 O14	Port Data 2 JTAG Data Output	GATE NO GATE	Note 3-2 Note 3-3 Note 3-23
PD1 TDI	IOP14 I	Port Data 1 JTAG Data Input	GATE NO GATE	Note 3-2 Note 3-3 Note 3-23
PD0 TCK	IOP14 I	Port Data 0 JTAG Clock Input	GATE NO GATE	Note 3-2 Note 3-3 Note 3-23
SLCTIN#	OD16	This active low output selects the printer. This is the complement of bit 3 of the Printer Control Register.	GATE	Note 3-2 Note 3-3 Note 3-7
INIT#	OD16	This output is bit 2 of the printer control register. This is used to initiate the printer when low.	GATE	Note 3-2 Note 3-3 Note 3-7
ERROR#	I	A low on this input from the printer indicates that there is a error condition at the printer. Bit 3 of the Printer Status register reads the ERR# input.	GATE	Note 3-19 Note 3-7
ALF#	OD8	This output goes low to cause the printer to automatically feed one line after each line is printed. The ALF# output is the complement of bit 1 of the Printer Control Register.	GATE	Note 3-2 Note 3-3 Note 3-7
STROBE#	OD8	An active low pulse on this output is used to strobe the printer data into the printer. The STROBE# output is the complement of bit 0 of the Printer Control Register.	GATE	Note 3-2 Note 3-3 Note 3-7

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
UART 1 (8)				
DCD1#	I	Active low Data Carrier Detect input for the serial port. Handshake signal which notifies the UART that carrier signal is detected by the modem. The CPU can monitor the status of DCD# signal by reading bit 7 of Modem Status Register (MSR). A DCD# signal state change from low to high after the last MSR read will set MSR bit 3 to a 1. If bit 3 of Interrupt Enable Register is set, the interrupt is generated when DCD #changes state.	GATE	Note 3-2 Note 3-4 Note 3-9
GP043	IO4	GPIO	NO GATE	
MCDAT	O4	MCU Debug data output	NO GATE	
DSR1#	I	Active low Data Set Ready input for the serial port. Handshake signal which notifies the UART that the modem is ready to establish the communication link. The CPU can monitor the status of DSR# signal by reading bit 5 of Modem Status Register (MSR). A DSR# signal state change from low to high after the last MSR read will set MSR bit 1 to a 1. If bit 3 of Interrupt Enable Register is set, the interrupt is generated when DSR# changes state.	GATE	Note 3-2 Note 3-4 Note 3-9
GP044	IO4	GPIO	NO GATE	
MCCLK	O4	MCU Debug clock output	NO GATE	
RXD1	I	Receiver serial data input for port 1	GATE	Note 3-2 Note 3-4 Note 3-9
GP045	IO4	GPIO	NO GATE	
RTS1# [SYSOPT]	O4	Active low Request to Send output for the Serial Port. Handshake output signal notifies modem that the UART is ready to transmit data. This signal can be programmed by writing to bit 1 of the Modem Control Register (MCR). The hardware reset will reset the RTS# signal to inactive mode (high). RTS# is forced inactive during loop mode operation. Defaults to tri-state on VTR power on. SYSOPT Strap. Used to determine the configuration base address. See Note 3-22.	GATE	Note 3-2 Note 3-4 Note 3-22
GP046	IO4	GPIO	NO GATE	
GP047	IO8	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-9
TXD1	O8	Transmit serial data output for port 1.	GATE	

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
CTS1#	I	Active low Clear to Send input for the serial port. Handshake signal which notifies the UART that the modem is ready to receive data. The CPU can monitor the status of CTS# signal by reading bit 4 of Modem Status Register (MSR). A CTS# signal state change from low to high after the last MSR read will set MSR bit 0 to a 1. If bit 3 of the Interrupt Enable Register is set, the interrupt is generated when CTS# changes state. The CTS# signal has no effect on the transmitter.	GATE	Note 3-2 Note 3-4 Note 3-9
GP050	IO4	GPIO	NO GATE	
DTR1# [Test_EN]	O4	Active low Data Terminal ready output for the Serial Port. Handshake output signal notifies modem that the UART is ready to transmit data. This signal can be programmed by writing to bit 1 of the Modem Control Register (MCR). Defaults to tri-state on VTR power on. Test Enable Strap. Used to enable test functions. Firmware samples the Test_Enable before de-asserting RSMRST# to enable the test function. Should be grounded for normal use.	GATE	Note 3-2 Note 3-4 Note 3-9, Note 3-11
GP051	IO4	GPIO	NO GATE	
RI1#	I	Active low Ring Indicator input for the serial port. Handshake signal which notifies the UART that the telephone ring signal is detected by the modem. The CPU can monitor the status of RI# signal by reading bit 6 of Modem Status Register (MSR). A RI# signal state change from low to high after the last MSR read will set MSR bit 2 to a 1. If bit 3 of Interrupt Enable Register is set, the interrupt is generated when nRI changes state.	NO GATE	Note 3-2 Note 3-4 Note 3-9
GP052	IO4	GPIO	NO GATE	
UART 2 (8)				
GP053	IO4	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-9
DCD2#	I	Data Carrier Detect 2 Input. See DCD1# pin description above.	GATE	
GP054	IO4	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-9
DSR2#	I	Data Set Ready 2 Input. See DSR1# pin description above.	GATE	
GP055	IO4	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-9
RXD2	I	Receive Serial Data 2 Input. See RXD1 pin description above.	GATE	

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
GP056 RTS2#	IO4 O4	GPIO Request to Send 2 Output. See RTS1# pin description above.	NO GATE GATE	Note 3-2 Note 3-4
GP057 TXD2	IO8 O8	GPIO Transmit Serial Data 2 Output. See TXD1 pin description above.	NO GATE GATE	Note 3-2 Note 3-4 Note 3-9
GP060 CTS2#	IO4 I	GPIO Clear to Send 2 Input. See CTS1# pin description above.	NO GATE GATE	Note 3-2 Note 3-4 Note 3-9
GP061 DTR2#	IO4 O4	GPIO Data Terminal Ready Output. See DTR1# pin description above.	NO GATE GATE	Note 3-2 Note 3-4 Note 3-9
GP062 RI2#	IO4 I	GPIO Ring Indicator 2 Input. See RI1# pin description above.	NO GATE NO GATE	Note 3-2 Note 3-4 Note 3-9
CLOCK PINS (2)				
CLK32	I	32.768KHz input clock	NO GATE	
CLOCKI	I	14.318 MHz Input clock	NO GATE	
FDD INTERFACE (13)				
DSKCHG#	I	This input senses that the drive door is open or that the diskette has possibly been changed since the last drive selection. This input is inverted and read via bit 7 of I/O address 3F7H. The DSKCHG# bit also depends upon the state of the Force Disk Change bits in the Force Disk Change register (see Runtime Registers section).	GATE	Note 3-19
HDSEL#	OD12	Head Select Output. This high current output selects the floppy disk side for reading or writing. A logic “1” on this pin means side 0 will be accessed, while a logic “0” means side 1 will be accessed. Can be configured as an Push-Pull Output.	GATE	Note 3-2
RDATA#	I	Raw serial bit stream from the disk drive, low active. Each falling edge represents a flux transition of the encoded data.	GATE	Note 3-19
WRTPRT#	I	This active low Schmitt Trigger input senses from the disk drive that a disk is write protected. Any write command is ignored. The WRPRT# bit also depends upon the state of the Force Write Protect bit in the FDD Option register (see the Configuration Registers section).	GATE	Note 3-19
TRK0#	I	This active low Schmitt Trigger input senses from the disk drive that the head is positioned over the outermost track.	GATE	Note 3-19

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
WGATE#	OD12	Write Gate Output. This active low high current driver allows current to flow through the write head. It becomes active just prior to writing to the diskette. Can be configured as a Push-Pull Output.	GATE	Note 3-2
WDATA#	OD12	Write Disk Data Output. This active low high current driver provides the encoded data to the disk drive. Each falling edge causes a flux transition on the media. Can be configured as a Push-Pull Output.	GATE	Note 3-2
STEP#	OD12	Step Pulse Output. This active low high current driver issues a low pulse for each track-to-track movement of the head. Can be configured as a Push-Pull Output.	GATE	Note 3-2
DIR#	OD12	Step Direction Output. This high current low active output determines the direction of the head movement. A logic “1” on this pin means outward motion, while a logic “0” means inward motion. Can be configured as a Push-Pull Output.	GATE	Note 3-2
DS0#	OD12	Drive Select 0 Output. Can be configured as a Push-Pull Output.	GATE	Note 3-19
MTR0#	OD12	Motor On 0 Output. Can be configured as a Push-Pull Output.	GATE	Note 3-19
INDEX#	I	This active low Schmitt Trigger input senses from the disk drive that the head is positioned over the beginning of a track, as marked by an index hole.	GATE	Note 3-19
GP042 DRVDEN0	IO12 O12	GPIO Drive Density Select 0 Output.	NO GATE GATE	Note 3-2 Note 3-4
MISCELLANEOUS (24)				
GP003 PCIRST_OUT4	IO8 O8	GPIO Buffered PCI RESET#	NO GATE NO GATE	Note 3-2 Note 3-4
GP000 PWM6	IO8 O8	GPIO PWM Output 6 for controlling speed of fan	NO GATE NO GATE	Note 3-2 Note 3-4
GP002 PCIRST_OUT3#	IO8 O8	GPIO Buffered PCI RESET#	NO GATE NO GATE	Note 3-2 Note 3-4
GP001 TACH4	IO8 I	GPIO Input for monitoring a fan tachometer	NO GATE NO GATE	Note 3-2 Note 3-4
PWRBTN# GP015	I IO4	Power Button input GPIO	NO GATE NO GATE	Note 3-2 Note 3-4 Note 3-14

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
GP041 IO_PME#	IO4 OD4	GPIO Power Management Event output. This active low Power Management Event signal allows this device to request wakeup.	NO GATE NO GATE	Note 3-2 Note 3-4
SMBDAT2 GP010	IOD4 IO4	SMBus Data GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
SMBCLK2 GP011	IOD4 IO4	SMBus Clock GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
GP040 SMBDAT1	IO4 IOD4	GPIO SMBus Data	NO GATE NO GATE	Note 3-2 Note 3-4
GP036 SMBCLK1	IO4 IOD4	GPIO SMBus Clock	NO GATE NO GATE	Note 3-2 Note 3-4
LED1 GP006	O8 IO8	Active Low LED Output – ON, OFF, Blink. Blinks at 1Hz rate with a 50% duty cycle. When RSMRST# is asserted the blink rate can range from 0.33Hz to 1.0Hz due to Ring Oscillator variations GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
LED2 GP007	O8 IO8	Active Low LED Output – ON, OFF, Blink. Blinks at 1Hz rate with a 50% duty cycle. When RSMRST# is asserted the blink rate can range from 0.33Hz to 1.0Hz due to Ring Oscillator variations. GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
WDT_OUT# GP025	IO8	GPIO Must be configures as an open-drain output when used as WDT_OUT#	NO GATE	Note 3-2 Note 3-4
LATCHED_BF_CUT GP035	O8 IO8	Latched Backfeed Cut GPIO	NO GATE NO GATE	Note 3-2 Note 3-4 Note 3-20
GP031 BACKFEED_CUT#	IO8 OD8	GPIO Backfeed Cut	NO GATE NO GATE	Note 3-2 Note 3-4
PCIRST_OUT1# GP026	O4 IO4	Buffered PCI RESET# GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
PCIRST_OUT2# GP027	O4 IO4	Buffered PCI RESET#. GPIO	NO GATE NO GATE	Note 3-2 Note 3-4
PS_ON#/ GP030	OD4 IO4	Power Supply Turn-On (Open Drain Output) GPIO	NO GATE NO GATE	Note 3-2 Note 3-4 Note 3-21
PWR_GOOD_3V GP033	O4 IO4	Power Good Output GPIO	NO GATE NO GATE	Note 3-2 Note 3-4 Note 3-20
RSMRST# GP034	O4 IO4	Resume Reset Output GPIO	NO GATE NO GATE	Note 3-2 \Note 3-4 Note 3-20

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
SLP_S3#	I	Sleep S3 power plane control Input. An active low indicates that the system is in the Suspend to RAM state.	NO GATE	Note 3-2 \Note 3-4 Note 3-13
GP065	IO4	GPIO	NO GATE	
SLP_S4_S5#	I	Sleep S5 power plane control Input. An active low indicates that the system is in the Soft Off state. General Purpose I/O	NO GATE	Note 3-2 \Note 3-4 Note 3-13
GP066	IO4	GPIO	NO GATE	
PWRGD_ PS	I	Active high Power Good Input from power supply. (Input to VTR powered logic)	NO GATE	Note 3-2 \Note 3-4 Note 3-13
GP067	IO4	GPIO	NO GATE	
GP070	IO4	GPIO	NO GATE	Note 3-2 \Note 3-4
SPEAKER	O4	SPEAKER Output	NO GATE	
GP071	IO4	GPIO	NO GATE	Note 3-2 \Note 3-4
IO_SMI#	OD4	System Management Interrupt	NO GATE	
TRST#	I	JTAG Reset input Must be tied to ground during normal use.	NO GATE	
SPI INTERFACE (4)				
SPI_DI	I	Serial Peripheral Interface input.	NO GATE	
GP012	IO4	GPIO	NO GATE	
SPI_DO	IO16	Serial Peripheral Interface Output Forced low on VTR power on	NO GATE	
GP004	IO16	GPIO	NO GATE	
SPI_CK	O16	Serial Peripheral Interface Clock Forced low on VTR power on	NO GATE	
GP013	IO16	GPIO	NO GATE	
SPI_CS#	O8	Serial Peripheral Interface Chip select. Forced high on VTR power on.	NO GATE	
GP032	IO8	GPIO	NO GATE	
KEYBOARD AND MOUSE(6)				
KCLK	IOD12	Keyboard Clock I/O	NO GATE	Note 3-19
KDAT	IOD12	Keyboard Data I/O	NO GATE	Note 3-19
MCLK	IOD12	Mouse Clock I/O	NO GATE	Note 3-19
MDAT	IOD12	Mouse Data I/O	NO GATE	Note 3-19
GP063	IO4	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-5
KBDRST#	OD4	Keyboard Reset Open-Drain Output.	GATE	
GP064	IO4	GPIO	NO GATE	Note 3-2 Note 3-4 Note 3-5
A20M	OD4	Gate A20 Open-Drain Output.	GATE	
PECI INTERFACE (4)				
PECI V_{REF}	PECI_ VREF	PECI IO Voltage Source. 0.95V to 1.26V		

TABLE 3-1: SIGNAL DESCRIPTIONS (CONTINUED)

Name Note 3-1	Buffer Type per Function	Description	Signal Affected by VCC=0 Note 3-8	Notes
PECI LVSMBCLK1 GP072	PECI_IO LVIOD4 LVIOD4	PECI Data IO Low voltage SMBus clock. GPIO All functions on this pin are at the PECI voltage.	GATE NO GATE NO GATE	Note 3-18
PECI_READY LVSMBDAT1 GP073	PECI_I LVIOD4 LVIO4	PECI Ready to read Low voltage SMBus data. GPIO All functions on this pin are at the PECI voltage.	GATE NO GATE NO GATE	Note 3-16 Note 3-18
GP005 PECI_REQUEST#	IO4 OD4	GPIO PECI Request. This pin is open-drain and should be pulled up to VCC.	NO GATE GATE	Note 3-2 Note 3-4
POWER AND GROUND (18)				
VCC	pwr	VCC power indication	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VTR	pwr	3.3V Supply Voltage	-	-
VBAT	pwr	Battery Input Voltage	-	-
HVTR	pwr	Hardware Monitor 3.3V Supply Voltage	-	Note 3-6
VSS	gnd	Digital Ground	-	-
VSS	gnd	Digital Ground	-	-
VSS	gnd	Digital Ground	-	-
VSS	gnd	Digital Ground	-	-
VSS	gnd	Digital Ground	-	-
VSS	gnd	Digital Ground	-	-
AVSS	agnd	Analog Ground	-	-
HVSS	gnd	Hardware Monitor Ground	-	-
CAP1	regulator	Regulator filter 4.7µF ±20% Capacitor to ground (ESR ≤ 2 ohms)	-	-

Note 3-1 Names in parentheses are software functions.

Note 3-2 Output Buffers are 5V tolerant in open drain mode only.

Note 3-3 When PWRGD_PS is 0 the buffers are tri-stated.

Note 3-4 If a pin is used as a push pull output, it is not 5V tolerant and should not be pulled up to 5V.

Note 3-5 External pullups must be placed on the KBDRST# and A20M pins. These pins are GPIOs that are inputs after an initial power-up (VTR POR). If the KBDRST# and A20M functions are to be used, the system must ensure that these pins are high.

Note 3-6 HVTR must be connected to the suspend power well (VTR).

Note 3-7 Refer to Parallel Port description for use of this pin in ECP and EPP mode.

Note 3-8 **NO GATE** indicates that the pin is not protected, or affected by V_{CC} = 0 operation. **GATE** indicates that the pin is protected as an input, or set to a **HI-Z** state as an output. The **GATE** function applies when V_{CC} is below the trip point in the reset generator, when the pin PWRGD_PS is low (de-asserted) or when the pin SLP_S3# is low (asserted).

Note 3-9 The **GATE** function also applies to this pin when pin LRESET# is low (asserted).

Note 3-10 Buffer is active when GPIO selected.

Note 3-11 Footprints for a Pull-Up and a Pull-Down resister should be placed near pin 104 to strap the Test_EN function. A pull-up to VTR will enable the test interface on VTR POR; a 0 will disable the interface.

Note 3-12 These function are on HVTR power supply.

Note 3-13 When this pin is configured as a GPIO, the alternate function input is forced high.

Note 3-14 The PWRBTN# function is always enabled, even when the pin is configured as a GPIO output. It is not affected by the GPIO configuration parameters, including polarity.

Note 3-15 The INTRUDER# function is always enabled, even when the pin is configured as a GPIO output. It is not affected by the GPIO configuration parameters, including polarity.

Note 3-16 The PECI_Ready Signal should ONLY be connected to signals that are on the VTT rail. Connect to VTT, PCI Reset or PCI Clockrun.

Note 3-17 When this pin is configured for PROCHOT_OUT# or GPIO, the pad type must be set to Open Drain.

Note 3-18 Both the pin mux control and the pad type control (Push-Pull for PECI and Open Drain for LVSMB) must be configured to switch this pin between PECI and LVSMBUS. When this pin is configured for PECI and push-pull, the pad drive strength is weak, as defined for PECI; when this pin is configured for Low Voltage SMBus and open-drain, the drive strength is as defined by the buffer type.

Note 3-19 This Pin is 5V tolerant.

Note 3-20 This Pin is forced low and glitch-protected until VTR is stable and internal power supplies have stabilized.

Note 3-21 This Pin is held in tri-state and glitch-protected until VTR is stable and internal power supplies have stabilized.

Note 3-22 Footprints for a Pull-Up and a Pull-Down resister should be placed near pin 101 to strap the SYSOPT function. A 0 will set the configuration base address to 002Eh; a pull-up to VCC will set the configuration base address to 004Eh.

Note 3-23 The JTAG signal becomes active when the TRST# pin is high.

3.1 Buffer Type Description

Note: The buffer type values are specified at VCC=3.3V.

TABLE 3-2: BUFFER TYPES

Buffer Type	Description
I	Input with Schmitt Trigger with 400mV hysteresis, TTL Compatible.
I_M	Input - Hardware Monitoring Block.
I_{AN}	Analog Input, Hardware Monitoring Block.
IO4	Input/Output, 4mA sink, 4mA source.
O4	Output, 4mA sink, 4A source.
OD4	Output (Open Drain), 4mA sink
IO8	Input/Output, 8mA sink, 8mA source.
O8	Output, 8mA sink, 8mA source.
OD8	Output (Open Drain), 8mA sink
IO12	Input/Output, 12mA sink, 12mA source.
O12	Output, 12mA sink, 12mA source.
OD12	Output (Open Drain), 12mA sink.
IOD12	Input/Output (Open Drain), 12mA sink.
IOP14	Input/Output, 14mA sink, 14mA source. Backdrive protected.
IO16	Input/Output, 16mA sink, 16mA source
O16	Output, 16mA sink, 16mA source
OD16	Output (Open Drain), 16mA sink.
IO24	Input/Output, 24mA sink, 24mA source.
LVIOD4	Input/Output (Open Drain), V_{REF} = 1.2V, I_{OL} = 4ma. See DC Electrical Characteristics section.
LVIO4	Input/Output, V_{REF} = 1.2V, I_{OL} = 4ma. See DC Electrical Characteristics section.
LVIOD24	Input/Output (Open Drain), V_{REF} = 1.2V, I_{OL} = 24ma. See DC Electrical Characteristics section.
IOD_PH	Input/Output (Open Drain), V_{REF} = 1.2V, I_{OL} = 24ma. See DC Electrical Characteristics section.
OD_PH	Output (Open Drain), V_{REF} = 1.2V, I_{OL} = 24ma. See DC Electrical Characteristics section.
PCI_I	Input. These pins meet the PCI 3.3V AC and DC Characteristics. (Note 3-24
PCI_O	Output. These pins meet the PCI 3.3V AC and DC Characteristics. (Note 3-24)
PCI_IO	Input/Output These pins meet the PCI 3.3V AC and DC Characteristics. (Note 3-24)
PCI_CLK	Clock Input. These pins meet the PCI 3.3V AC and DC Characteristics and timing. (Note 3-25)
PECI_I	Input. These pins are at the PECI V_{REF} level
PECI_IO	Input/Output These pins are at the PECI V_{REF} level

Note 3-24 See the “PCI Local Bus Specification,” Revision 2.1, Section 4.2.2.

Note 3-25 See the “PCI Local Bus Specification,” Revision 2.1, Section 4.2.2 and 4.2.3.

3.2 Backdrive Protection

All pins except pins with buffer type PCI_IO and IAN are backdrive protected. The signals are listed in the specification. The backdrive leakage maximum at 5V is 10µA.

3.3 5V Tolerance

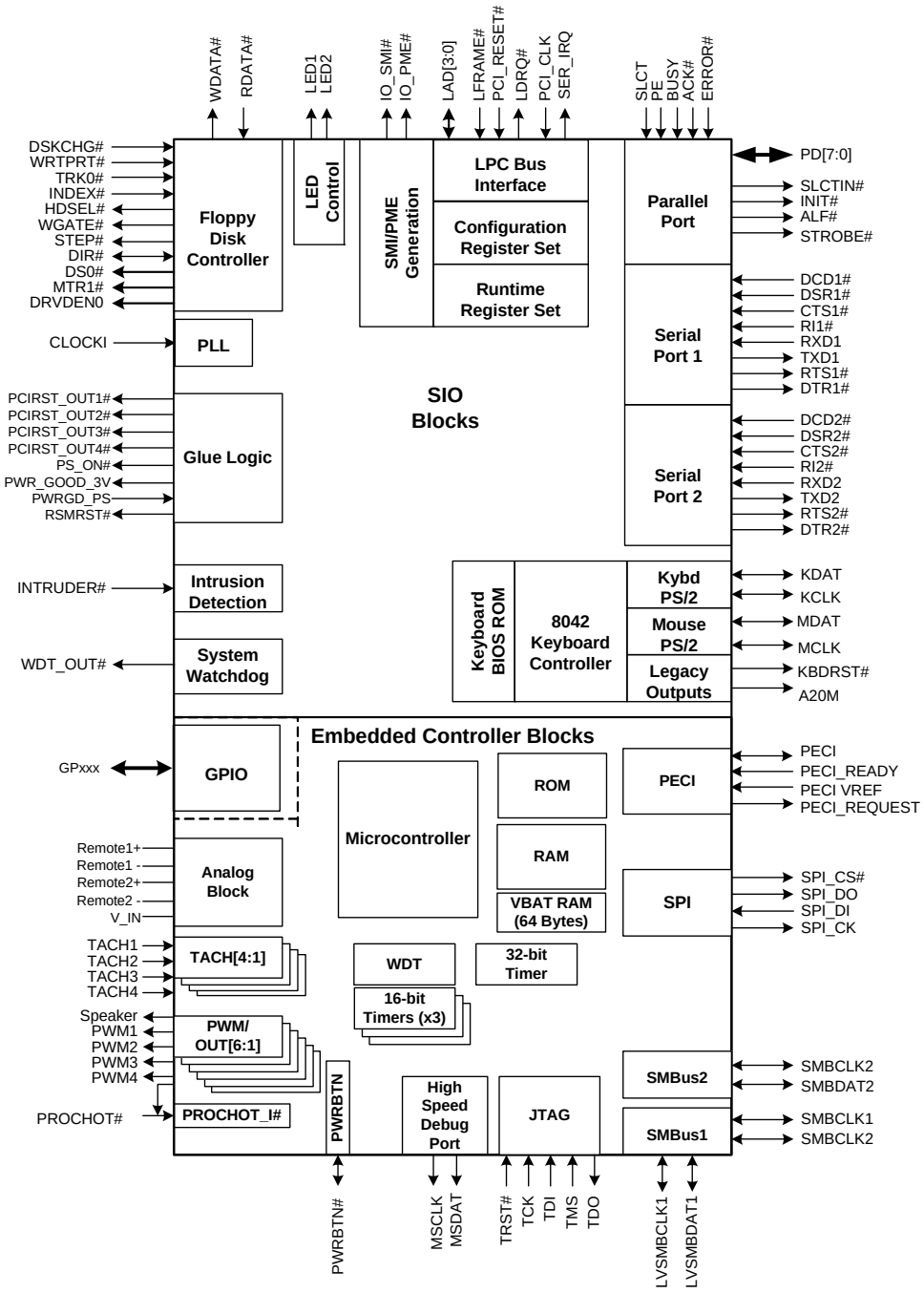
5V tolerant leakage is maximum 10µA. The maximum leakage at 5.5V is 16µA.

4.0 BUS HIERARCHY

4.1 Block Diagram

Figure 4-1 shows, in graphic form, the inter connectivity of devices on the SCH5636, including the EC, the principal lower buses and most of the peripherals.

FIGURE 4-1: SCH5636 BLOCK DIAGRAM

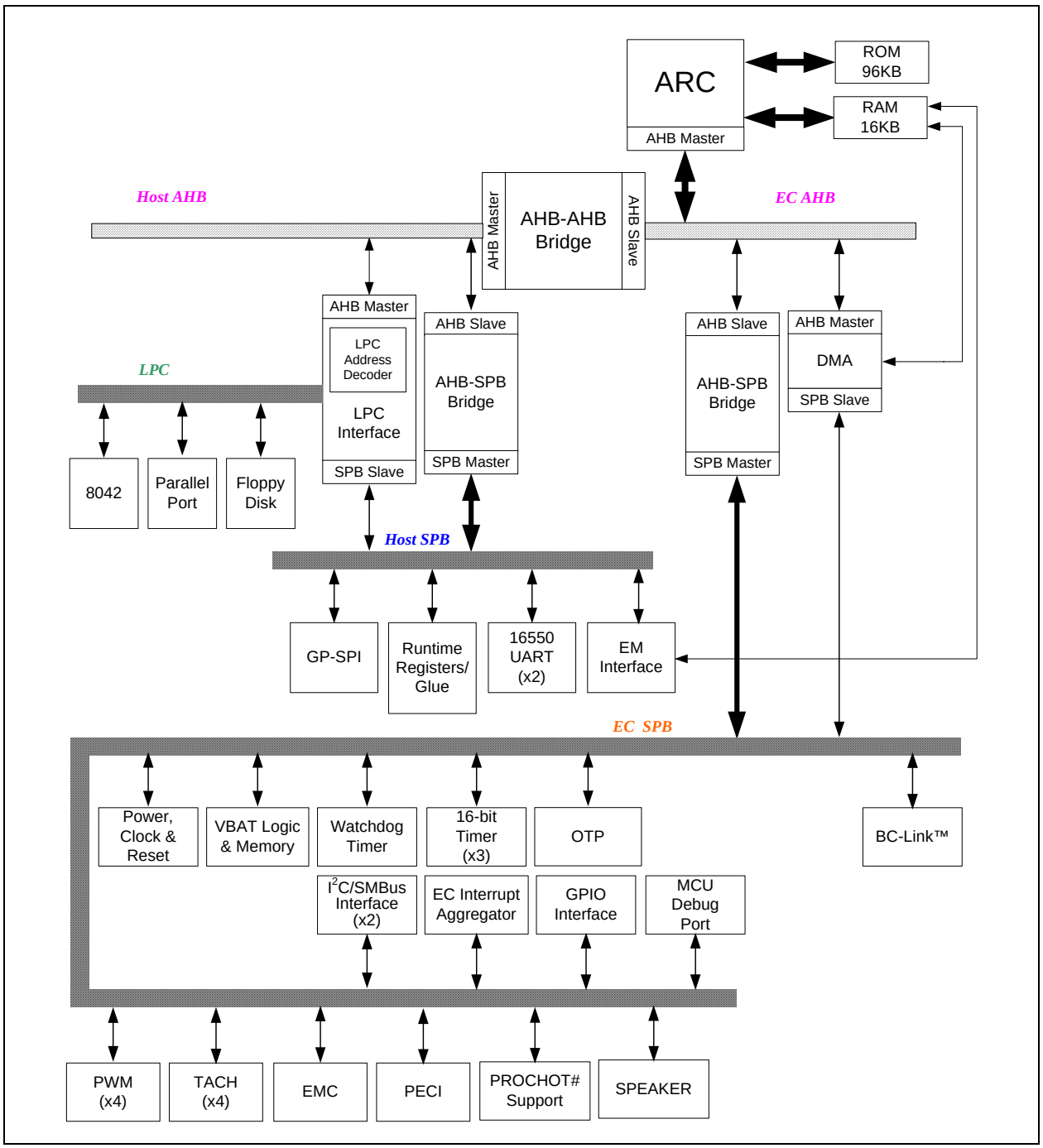


See pinout diagram for multiplexing of these interface functions onto pins.

4.2 Address Space

The ARC EC has a 24-bit address space. Addresses in the lower half of the range, 0h through 7F_FFFFh, can be used for both instruction access and data access. The address range 0h through 1_7FFFh is mapped to the 96K ROM. Addresses in the range 80_0000h through 80_3FFFh are mapped to the Closely Coupled Data Memory. These memories are shown in Figure 4-3). Addresses greater or equal to 80_1000h are propagated through the AHB interface on the ARC processor. References, by either the EC or the Host via the LPC bus, to addresses that are not mapped to any device register or memory will cause a bus error; see Section 4.3.3, "AHB Bus Errors," on page 28.

FIGURE 4-2: SCH5636 BUS HIERARCHY



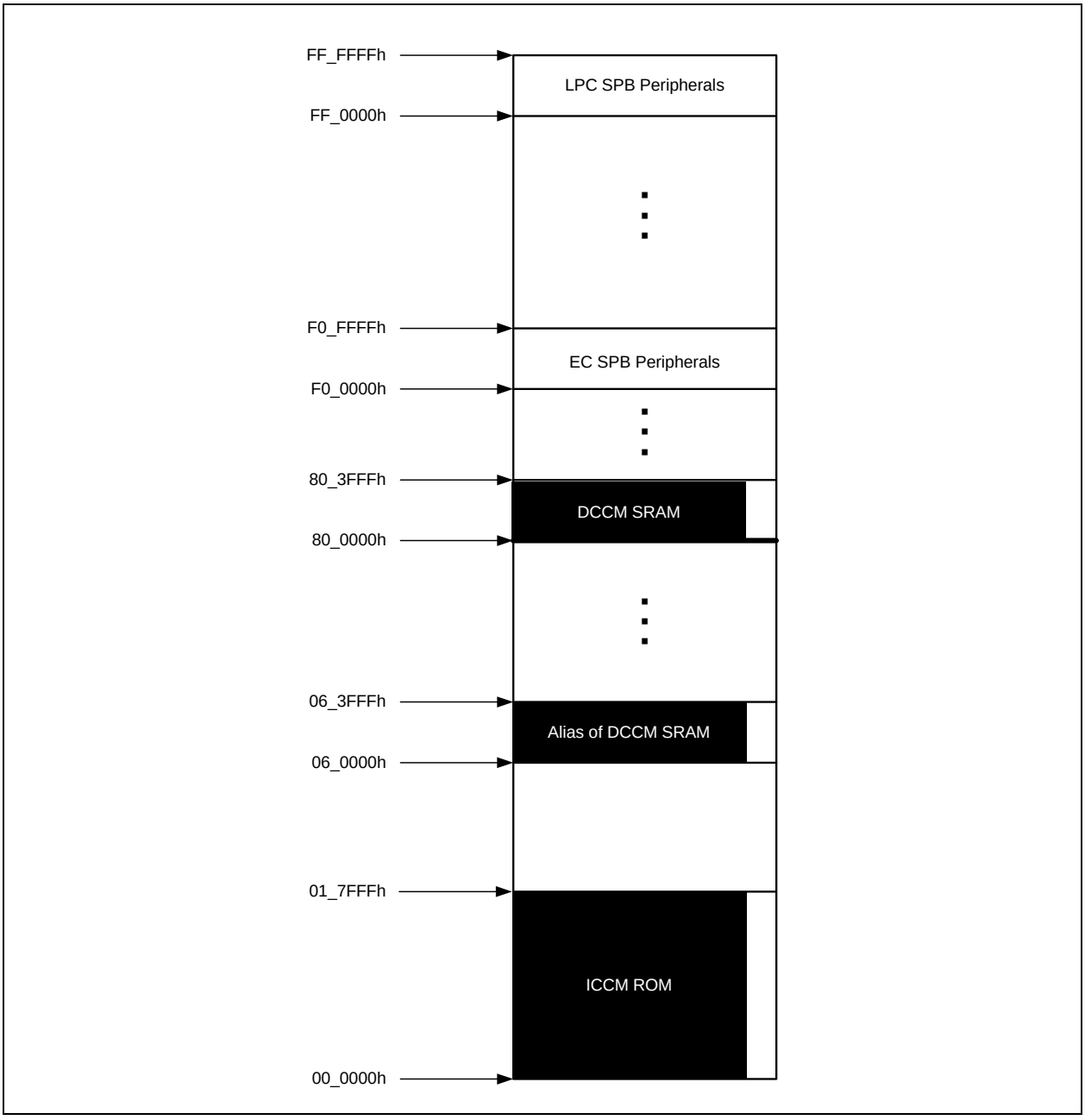
4.2.1 ARC ADDRESS SPACE

The ARC Address Space is illustrated in Figure 4-3, "SCH5636 EC Memory Map". EC Instruction Memory occupies the lower half of the address space, 00_0000h through 7F_FFFFh. The ARC processor can only execute instructions that are located in the Instruction Memory portion of the address space. Only a portion of the Instruction space is populated.

The upper half of the address space is used solely for data references. The region contains a general-purpose EC Data Memory SRAM. as well as the address space of the two SPB peripheral buses.

The contents of the Instruction memory can be read and written by the ARC processor through regular load and store instructions. However, there may be a one cycle instruction fetch penalty whenever a data load or store instruction to Instruction memory is executed.

FIGURE 4-3: SCH5636 EC MEMORY MAP



4.2.2 AHB ADDRESS SPACE

The components on the AHB subsystem and the two SPB bridges define a set of addresses that are accessible by the EC. This address space is shown in Table 4-1, "SCH5636 Peripheral Address Space". As shown in the table, the Host can access much of the address space, but not all.

TABLE 4-1: SCH5636 PERIPHERAL ADDRESS SPACE

Address Range	Device	Accessible by EC	Accessible by Host
F0_0000h - F0_FFFFh	EC SPB Bridge	Yes	No
FF_0000h - FF_FFFFh	Host SPB Bridge	Yes	Yes (limited by LPC interface map)

The 64KB address space of an SPB Bridge is divided into 1KB Frames. Peripherals are grouped into Logical Devices; each Logical Device corresponds to a 1KB Frame. Logical Devices addressable by the Host are listed in Table 4-2, "Host Logical Devices on SCH5636". The 1KB Address Frame of a Host Logical Device is divided into four subregions, as described in Section 4.3.2, "Address Framing," on page 28.

TABLE 4-2: HOST LOGICAL DEVICES ON SCH5636

Logical Device Number	Logical Devices	AHB Address of Frame	Runtime Registers	Configuration Registers	EC-only Registers
0h	**EM Interface**	**FF_0000h**	yes	no	yes
1h	**Keyboard Controller**	**FF_0400h**	yes Note 4-1	yes	no
7h	**UART1**	**FF_1C00h**	yes	yes	no
8h	**UART2**	**FF_2000h**	yes	yes	no
Ah	**Runtime Registers**	**FF_2800h**	yes	yes	yes
Bh	**Floppy Disk Controller**	**FF_2C00h**	yes Note 4-1	yes	no
Ch	**LPC Interface**	**FF_3000h**	no	yes	yes
Fh	**SPI**	**FF_3C00h**	yes	no	no
11h	**Parallel Port**	**FF_4400h**	yes Note 4-1	yes	no
3Fh	**Global Configuration**	**FF_FC00h**	no	no	yes

Note 4-1 These devices have local BARs. There is no BAR for them in the LPC Configuration space. The Configuration registers for the devices are located on the AHB bus at the frames specified.

The 64KB address space of the EC SPB Bridge is divided into 1KB Frames. Peripherals are grouped into Logical Devices; each Logical Device corresponds to a 1KB Frame. These EC Logical Devices are listed in Table 4-3, "EC Logical Devices on SCH5636". Multiple instantiations of the same block can exist in a single 1KB Frame. Each instantiation is separated by 128 bytes.

Note 1: All VBAT powered registers are in a Single 1KB Frame separated by 128 bytes (see Table 4-3, Logical Device Number 33h.)

2: Although the EC SPB Bridge address space uses the terminology of Logical Device; there is no host access to these blocks.

TABLE 4-3: EC LOGICAL DEVICES ON SCH5636

Logical Device Number	AHB Address for Base of Frame	Logical Devices	Notes
1h	F0_0400h	Watchdog Timer	
3h	F0_0C00h	16-bit Timer	
5h	F0_1400h	BC Bus Master	
6h	F0_1800h	SMBus	
9h	F0_2400h	DMA	
Ah - 15h	F0_2800h - F0_4FFFh	Reserved	
16h	F0_5800h	PWM	
17h	F0_5C00h	Reserved	
18h	F0_6000h	TACH	
19h	F0_6400h	PECI	
1Ah	F0_6800h	EMC3	
1Bh	F0_6C00h	PROCHOT# Monitor	
1Ch - 22h	F0_7000h - F0_8BFFh	Reserved	
23h	F0_8C00h	MCU Debug Port	
24h - 2Fh	F0_9000h - F0_BCFFh	Reserved	
30h	F0_C000h	ARC Interrupts	
31h	F0_C400h	GPIOs	
32h	F0_C800h	Power, Clock & Reset (VTR PWR'ed)	
33h	F0_CC00h	Power, Clock & Reset (VBAT PWR'ed)	
	F0_CD00h	VBAT Backed Memory	
34h - 3Eh	F0_D000h - F0_FBFFh	Reserved	
3Fh	F0_FC00h	EC Test and Debug	

4.3 AHB Buses

Addresses and internal buses in the SCH5636 are compatible with ARM Limited's *Advanced Microprocessor Bus Architecture* (AMBA), as specified in *AMBA™ Specification (Rev 2.0),* 1999.

As seen in FIGURE 4-1: SCH5636 Block Diagram on page 23, there are two separate AHB buses, the EC AHB and the Host AHB. The EC AHB has two masters and two slaves, while the Host AHB has two masters and one slave. The bus connections are summarized in Table 4-4, "SCH5636 AHB Buses" and can be seen in Figure 4-1.

TABLE 4-4: SCH5636 AHB BUSES

AHB Bus	Master Interfaces	Slave Interfaces
EC AHB	EC DMA Interface	EC SPB bridge AHB-AHB bridge
Host AHB	LPC Interface AHB-AHB bridge	Host SPB bridge

The AHB-to-AHB bridge is a one-way device: addresses generated on the EC AHB can be propagated to the Host AHB bus, but addresses on the Host AHB bus cannot be propagated to the EC AHB bus. This is the reason that the Host is restricted from accessing the address range F0_0000h to F0_FFFFh (the address range of the EC SPB bridge). The bridge maps address from FD_0000h through FF_FFFFh.

Both the EC and the LPC interface can have at most one outstanding AHB bus request at one time. On the Host AHB, the LPC interface always has priority over the AHB-AHB bridge. Both AHB buses support byte, halfword and word AHB transfers. AHB bus locking or early burst termination are not supported.

4.3.1 BUS CLOCKING

The Host Bus Clock runs by default at the system clock rate of 64.52 MHz but can be configured to run at a reduced clock rate. The bus clock and the bus arbiter can be shut down when there are no transactions active on the bus. The bus clock and arbiter will be restarted as soon as an address is acquired from the LPC bus, or when an EC AHB bus transaction is mapped, via the AHB-to-AHB bridge, to the address space of the Host AHB. The EC AHB clock can be programmed to run at any of the available rates between 8.1Mhz MHz and 32.2MHz. See Section 5.0, "Power, Clocks and Resets," on page 30. The EC bus clock is shut down when the EC is idle.

If the Host Bus clock is idle when an LPC transaction arrives at the LPC interface, the LPC interface will restart the Host bus clock and arbiter early enough so that as soon as an I/O address is translated to an AHB address the I/O transaction can be placed on the AHB without delay. If the I/O address is not claimed by the SCH5636 then the LPC interface will drop its bus request. If the EC is not requesting the Host AHB at the same time, the Host AHB bus clock will again shut down.

4.3.2 ADDRESS FRAMING

The EC can directly address all peripherals on the SCH5636. The Host, by contrast, is restricted in what it can address. The Host accesses the SCH5636 through the LPC bus using I/O cycles, (see Section 8.0, "Host Interface," on page 75). These cycles are mapped into the SCH5636 address space accessible by the LPC interface.

The mapping function forces some of the address bits to preset values, as shown in Table 4-5, "LPC to SCH5636 Address Mapping". This mapping has the effect of creating four contiguous 256-byte regions: LPC I/O, EC-only, LPC DMA and LPC Configuration. Together, the four regions create a 1024 byte “frame” for each logical device that is accessible to the Host. There is therefore a maximum of 64 Logical Devices, each with a 1KB frame, located on the SCH5636.

LPC I/O cycles for Runtime Registers are mapped into the first 256 bytes of the 1024 byte frame. Configuration Registers, which are accessed through a Configuration portal typically located at addresses 2Eh and 2Fh in the LPC I/O space, are restricted to the highest 256 bytes of the 1024 byte frame.

DMA FIFO addresses are restricted to offsets between 200h and 2FFh within a Logical Device frame. DMA FIFO addresses are not used on the SCH5636. DMA LPC accesses to legacy devices (parallel port and floppy disk controller) are directly to the legacy devices.

Because the EC does not require the mapping mechanism required for translating LPC Runtime Registers, Configuration Registers and DMA channels, it accesses each 1KB frame uniformly. All Logical Devices located on the EC-only AHB (those in the address range F0_0000h through F0_FFFFh) have a flat, 1KB frame. Each 1KB logical device frame on the EC side may be further divided into 8 128-byte instance blocks.

For details on LPC address mapping to the SCH5636 address space, see Section 8.0, "Host Interface," on page 75.

TABLE 4-5: LPC TO SCH5636 ADDRESS MAPPING

Type of Access	Address Bit Mapping
LPC I/O Access Runtime Registers	Address bits[9:8] = 00b Address bits[23:10] set from map
No LPC Access EC-only registers	Address bits[9:8] = 01b
LPC I/O Access through Configuration Access Port Configuration Registers	Address bits[9:8] = 11b Address bits[23:10] set from map
LPC DMA Access	Address bits[9:8] = 10b Address bits[23:10] set from map

4.3.3 AHB BUS ERRORS

AHB bus requests by both the Host, through the LPC bus, and the EC, can be terminated with an AHB bus error. The handling of bus errors by the EC is described in the *ARCompact™ Instruction Set Architecture: Programmer’s Reference*.

Bus errors may be caused by:

- EC I/O requests that lie outside the range of the Data Closely Coupled Memory, the EC SPB or the AHB-AHB bridge
- I/O requests to either the Host SPB or the EC SPB that map to a non-existent Logical Device
- I/O requests to an invalid register address within a valid Logical Device on either the EC SPB or the Host SPB

4.4 Peripheral Buses (SPB)

The Microchip Peripheral Bus (SPB) is a byte-addressable bus with a 16-bit address space and a 32-bit data path. All accesses must be aligned to the data-size boundaries. The SPB supports 32-bit, 16-bit and 8-bit accesses. All peripheral accesses are 32-bits wide, so that if a peripheral cannot transfer more than 8 bits of a register in one I/O access, register addresses should all be on 32-bit boundaries even though the upper bits (bits 31 through 8) are always zero. The SPB will not assemble a 32-bit word out of multiple 8-bit accesses.

SPB transfers take two cycles, so that a read or write on the AHB to a register on the SPB will take a total of three cycles. The SPB contains an 8-bit word address, four byte lane strobes and up to 64 logical device select strobes. Data on the SPB read and write buses are always 32-bit aligned, with the byte strobes indicating which byte lane or lanes should be active during a transaction. An SPB peripheral must steer bytes, based on the byte strobes, to the correct byte lane.

There are two SPB bridges forwarded to two SPB (busses):

Host SPB

The Host SPB address range is FF_0000h through FF_FFFFh

EC SPB

The EC SPB address range is F0_0000h through F0_FFFFh

If an AHB master attempts to access an address within an SPB bridge address space that is not assigned to a Logical Device, the master's access will be terminated with a bus error. In addition, an access to an invalid address within a valid Logical Device will be terminated with a bus error.

4.5 Register Table Conventions

4.5.1 REGISTER SIZE

Any register that is specified as 8-bits must be accessed by the EC using 8-bit load and store instructions. 16-bit or 32-bit loads and stores may have unexpected results. JTAG Debugger access should indirect using peek_poke_arc macros described in AN #16.1, MEC5035 JTAG Design and Layout Guide. These registers may be on any 8-bit address. The Host can only access 8-bit registers.

Any register that is specified as 16-bits must be accessed by the EC using 16-bit load and store instructions. 32-bit loads and stores may have unexpected results. JTAG Debugger access should indirect using peek_poke_arc macros described in AN #16.1, MEC5035 JTAG Design and Layout Guide. These addresses are on 16-bit aligned addresses.

Any register that is specified as 32-bits may be accessed by the EC using 32-bit loads and stores. Unless otherwise noted, bytes within the register can also be accessed using 8-bit or 16-bit loads and stores. These registers are on 32-bit aligned addresses.

Any bit not explicitly defined is reserved.

4.5.2 REGISTER TYPE

A register type of "R" means that all bits within the register are read-only. All stores to these registers are ignored.

A register type of "W" means that all bits within the register are write-only. Any read from one of these registers returns 0.

A register type of "R/W" means that at least one bit in the register is read/write.

A register type of "R/WC" means that all bits in the register are either read-only or read/write-clear. A read/write-clear bit is cleared to 0 when written with a '1b'. A write of a '0b' to the bit has no effect. Unless otherwise noted, the bit is not cleared on a write of '1b' if hardware is simultaneously setting the bit to '1b'.

A register type of "R/WS" means that all bits in the register are either read-only or read/write-set. A read/write-clear bit is set to 1 when written with a '1b'. A write of a '0b' to the bit has no effect.

4.5.3 REGISTER FIELDS

A field labeled "Reserved" always returns 0. Writes to a reserved field are ignored.

A field labeled "MCHP Reserved" is reserved for Microchip use. Reads should be ignored and any write should be 0. Writes of a '1b' to a field labeled "MCHP Reserved" may have unpredictable results.

5.0 POWER, CLOCKS AND RESETS

5.1 General Description

The Power, Clocks and Resets chapter includes descriptions of the SCH5636 Power configuration, clock sources and clock domains, and the various reset signals. This section also describes the Power Management Interface.

5.2 References

1. Advanced Configuration and Power Interface Specification, Revision 1.0b, February 2, 1999.
2. Intel® 82801DBM I/O Controller Hub 4 Mobile (ICH4-M), Data Sheet, Order Number: 252337-001, Intel Corp., January 2003.

5.3 Interrupts

When either the VBAT_POR or the WDT status bits in the Power-Fail and Reset Status Register is asserted '1', the PFR interrupt status bit in the GIRQ23 Source register in the EC Interrupt Aggregatorr is set to '1'. The PFR may be used to generate an EC interrupt, or the interrupt may be masked, as described in Section 7.0, "EC Interrupt Aggregator," on page 63.

5.4 Power Configuration

The SCH5636 is influenced by three separate power planes, VBAT, VTR, and VCC, as described in Table 5-1:

TABLE 5-1: TYPICAL SCH5636 POWER SUPPLIES VS. ACPI POWER STATES

Supply Name	ACPI Power State						Description
	S0 (FULL ON)	S1 (POS)	S3 (STR)	S4 (STD)	S5 (Soft Off)	G3 (MECH OFF)	
VBAT	ON	ON	ON	ON	ON	ON	SCH5636 V_{BAT} Well Supply (assuming a Type 2 configuration as described in Section 5.4.4, "Power Supply Configurations," on page 31)
VTR	ON	ON	ON	ON/OFF	ON/OFF	OFF	SCH5636 Suspend Supply. (Note 5-1)
VCC	ON	ON	OFF	OFF	OFF	OFF	SCH5636 Runtime Supply

Note 5-1 VTR availability in S4 - S5 may depend, for example, on whether AC power is available.

The VBAT and VTR power planes provide power directly to the SCH5636 through the VBAT and VTR pins. The SCH5636 senses the VCC power state using the PWRGD_PS and VCC input pins. The VBAT, VTR, and VCC power sequencing requirements are as follows:

1. VCC power can be applied simultaneously with or after VTR power.
2. VTR power can be applied simultaneously with or after VBAT.

5.4.1 1.2V REGULATOR

The 1.2V regulator generates the SCH5636 core power well. As illustrated in FIGURE 5-2: Reset Interface Block Diagram on page 35, the input to the 1.2V Regulator is VTR, the output is VTR_1.2. The 1.2V Regulator is not used when VTR is inactive, as described in Section 5.4.2, "Power Mux" below.

The stability of the 1.2V Regulator amplifier depends on an external capacitor, VR_CAP. The capacitor requirement is defined in Table 3-1, "Signal Descriptions," on page 11. The choice of capacitor can be either ceramic or low ESR tantalum. Ceramics are the recommended choice due to their superior AC performance (below 100mΩ ESR), but X5R dielectrics should be used to prevent greater than 20% capacitance variation over temperature and voltage. Low ESR tantalum capacitors will work but care should be taken because the ESR can vary 2x at low temperatures.

5.4.2 POWER MUX

To ensure the highest reliability and lowest possible power consumption, the Power Mux switches between the 1.2V Regulator and a level-shifted VBAT voltage to produce the 1.2V internal supply for VBAT-backed logic.

Power Mux switching depends on the voltage level of the 1.2V Regulator and the VTR supply. As illustrated in Figure 5-2, the Power Mux selects the 1.2V Regulator after the VTR and the VTR_1.2 power supplies exceed preset voltage thresholds. The Power Mux selects the VBAT supply as soon as either the VTR or VTR_1.2 power supplies drop below these thresholds (see FIGURE 5-4: VTR Power-Down Timing on page 36).

Note that the Power Mux only switches 1.2 volts. To ensure minimum VBAT power consumption for 3.3V VBAT powered outputs when VTR is fully powered, supply switching from VBAT to VTR must be done externally.

5.4.3 VCC POWER GOOD

VCC Power Good is defined by the PWRGD_PS input pin. PWRGD_PS is also synchronized to the 64.52 MHz Ring Oscillator.

The PWRGD_PS input must always be driven to a ‘1’ or a ‘0,’ even when VCC is 0 V. The minimum PWRGD_PS pulse width (high and low) is shown below in Table 5-2.

TABLE 5-2: PWRGD_PS INPUT TIMING

Parameters	Symbol	MIN	TYP	MAX	Units	Notes
PWRGD_PS Pulse Width	t_{VPGPW}	100	–	-	ns	

5.4.4 POWER SUPPLY CONFIGURATIONS

There are two acceptable types of SCH5636 power supply configuration that fundamentally differ based on the need for a backup battery connection to VBAT.

5.4.4.1 Type 1

Type 1 configurations do not use a VBAT backup battery connection. Power supply requirements for Type 1 configurations are as follows: VBAT is tied to VTR and VTR is connected to the suspend supply.

In this configuration some internal components that utilize the VBAT power plane are switched internally to VTR using a Power Mux when VTR power is applied according to the internal VTRGD signal (see also Section 5.4.2, "Power Mux," on page 31).

5.4.4.2 Type 2

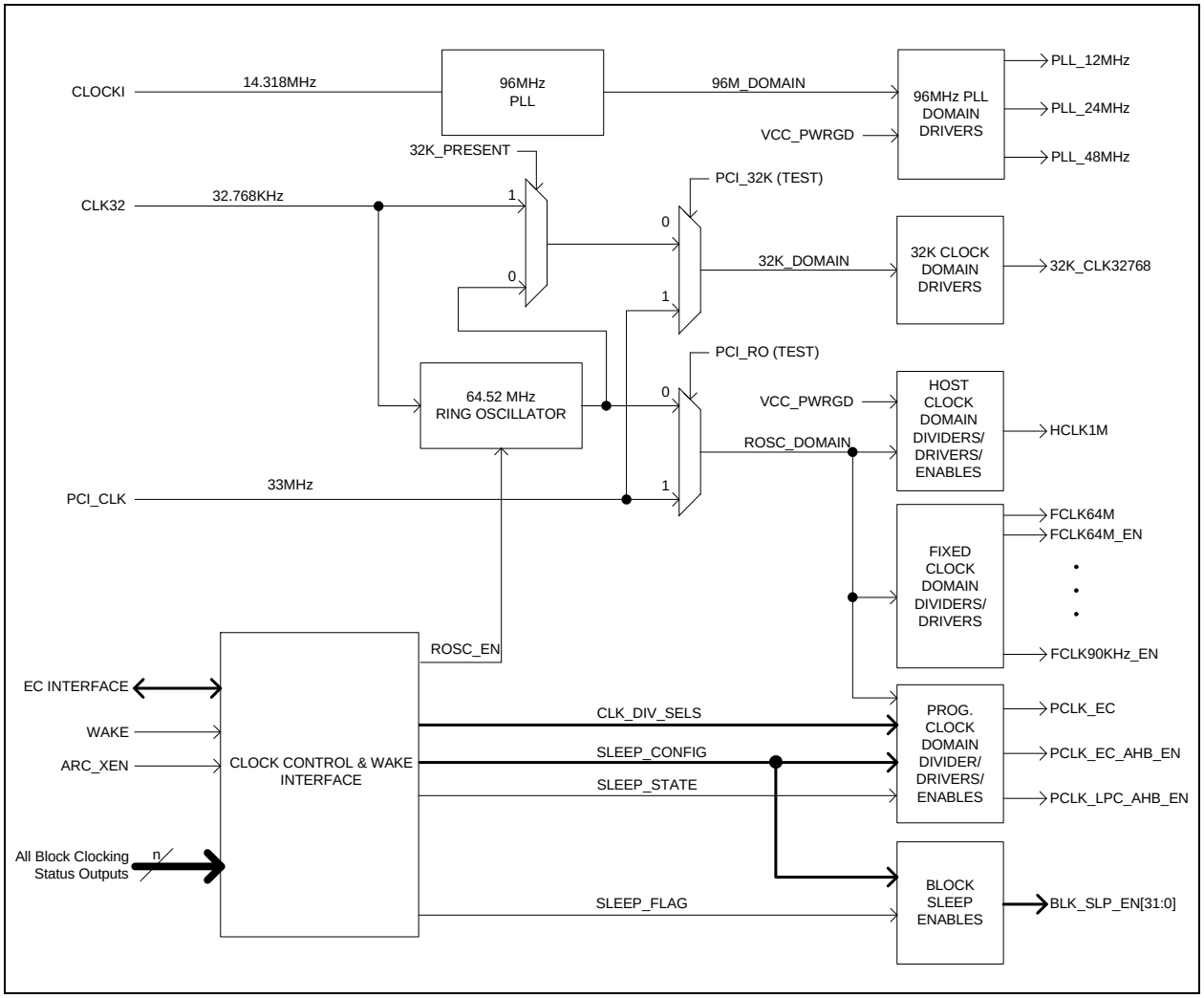
Type 2 configurations use an un-switched VBAT backup battery connection. Power supply requirements for Type 2 configurations are as follows: VBAT is connected to an un-switched backup battery and VTR is connected to the suspend supply.

In this configuration some internal components that utilize the VBAT power plane are switched internally to VTR using a Power Mux when VTR power is applied as described in Section 5.4.2, "Power Mux," on page 31.

5.5 Clock Sources

The SCH5636 includes four clock sources as illustrated in Figure 5-1, "Clock Generator Block Diagram": the 64.52 MHz Ring Oscillator, 32.768 KHz Clock, the PCI Clock and the 14.318MHz clock input.

FIGURE 5-1: CLOCK GENERATOR BLOCK DIAGRAM



5.5.1 64.52 MHZ RING OSCILLATOR

The SCH5636 Clock Sources includes a high-accuracy, low power, low start-up latency 64.52 MHz Ring Oscillator. The 64.52 MHz Ring Oscillator is always enabled when VTR power is applied except when the 64.52 MHz Ring Oscillator is stopped by hardware as described in Section 5.8, "Power Management Interface," on page 38. The 64.52 MHz Ring Oscillator start-up time t_{ADJ} is shown in Table 5-3.

Without correction or when the 32.768 KHz Clock is not running, the accuracy of the 64.52 MHz Ring Oscillator is ±50% (min.). When the 64.52 MHz Ring Oscillator is enabled and the 32.768 KHz Clock is running, the accuracy of the 64.52 MHz Ring Oscillator is automatically corrected by hardware from ±50% to ±2% using a free-running iterative algorithm (see a description of the FREQ_LOCK bit in the PCR Status Register and the SAA bit in the Clock Control Register). The t_{ADJ} time is shown in Table 5-3.

The 64.52 MHz Ring Oscillator is reset by VTRGD as described in Section 5.7.1, "VTRGD," on page 35.

TABLE 5-3: 64.52 MHZ RING OSCILLATOR TIMING PARAMETERS

Parameters	Symbol	MIN	TYP	MAX	Units
Adjustment Delay to ±2% Accuracy	t_{ADJ}	0.03	-	4 (Note 5-2)	ms
64.52 MHz Ring Oscillator Start-up Delay (±50% accuracy)	t_{SU}			6	µs

Note 5-2 This time only applies if the external 32KHz clock input is available.

5.5.2 32.768 KHZ CLOCK

The CLK32 clock input provides a stable timebase for the 64.52 MHz Ring Oscillator and a clock source for the 32K Clock Domain.

The 32.768KHz CLK32 clock is a single-ended input. The system core logic will typically provide an accurate 32.768KHz clock whenever RSMRST# is de-asserted.

The CLK32 input is glitch-free on VTR rise.

5.5.3 PCI CLOCK

The PCICLK clock input is generated by the system and provides the clock source for the PCI Clock domain. The PCI Clock input is typically 33MHz. The PCI Clock input can only be active when VCC power is active.

5.5.4 14.318MHZ CLOCK

The CLOCKI clock input is 14.318MHz and is provided by the system. It is used as the clock source for the 96MHz PLL used for generating clocks for legacy I/O blocks.

The CLOCKI input is only available when VCC power is active.

5.6 Clock Domains

5.6.1 EC BUS CLOCK

The EC Bus Clock is a programmable clock that is defined by the EC Clock Divider Register, as described in Section 5.9.1 on page 44, and derived from the 64.52MHz ring oscillator.

5.6.2 HOST BUS CLOCK

The Host Bus Clock is a programmable clock that is defined by the LPC_AHB Clock Divider Register, as described in Section 5.9.2 on page 45, and derived from the 64.52MHz ring oscillator.

5.6.3 FIXED CLOCK DOMAIN

The Fixed Clock Domain represents non-programmable clocks that are derived from the 64.52 MHz Ring Oscillator. The Fixed Clock Domain outputs include a main system clock, MCLK, as well as a set of clock enables that are used by individual blocks when the blocks requires a clock that is slower than the main system clock. The fixed clock domain clocks are summarized in Table 5-4, "All Fixed Clock Domains":

TABLE 5-4: ALL FIXED CLOCK DOMAINS

Symbol	Type (CLOCK or ENABLE)	Frequency
MCLK	CLOCK	64.52 MHz
MCLK_DIV2_EN	ENABLE	32.26 MHz
MCLK_DIV4_EN	ENABLE	16.13 MHz
MCLK_DIV8_EN	ENABLE	8.06 MHz
MCLK_DIV16_EN	ENABLE	4.03 MHz
MCLK_DIV32_EN	ENABLE	2.02 MHz
MCLK_DIV64_EN	ENABLE	1.01 MHz
MCLK_DIV128_EN	ENABLE	504 KHz
MCLK_DIV717_EN	ENABLE	90 KHz

5.6.4 32K CLOCK DOMAIN

The 32K Clock Domain represents all of the clocking derived from the 32.768 KHz Clock.

If the 64MHz clock is running when the 32KHz clock input is not available, the 32KHz clock is derived from the main ring. If the 64MHz clock is not running when the 32KHz clock input is not available, all clocks will be disabled. The 64MHz clock will restart when an external wake event is received.

The 32K Clock Domain is synchronized to the 64.52 MHz Ring Oscillator. Synchronization is disabled:

- When the 64.52 MHz Ring Oscillator is disabled (e.g., in a sleep state as defined in Section 5.5.1, "64.52 MHz Ring Oscillator" or during a test mode), and
- When VTRGD is not asserted (see Section 5.7.1, "VTRGD," on page 35).

In addition to its use in synchronization with the main ring oscillator, the 32K Clock Domain is also used for the Watchdog Timer, for the PWRBTN# debounce circuitry and for the LED glue logic.

5.6.5 96MHZ PLL DOMAIN

The 96MHz PLL is used to provide clocks for Legacy I/O blocks. It is synchronized to the 14.318MHz CLOCKI input and is only operational when the VCC power rail is active. It is divided down to 24MHz to supply clocks for the Floppy Disk Controller, 12MHz for the 8042 Keyboard Controller and Parallel Port, and 1.8432MHz for the UARTs.

Operation of the 96MHz PLL is described in Section 5.8.4, "96MHz PLL Control," on page 43.

5.6.6 PCI CLOCK DOMAIN

The 33MHz PCICLK input provides the clock for the PCI Clock domain. This clock is used in the LPC Host Interface and in the Parallel Port and Floppy Disk Controller legacy blocks.

5.6.7 CLOCK DOMAINS VS. ACPI POWER STATES

Table 5-5, "Typical SCH5636 Clocks vs. ACPI Power States" shows the relationship between ACPI power states and SCH5636 clock domains:

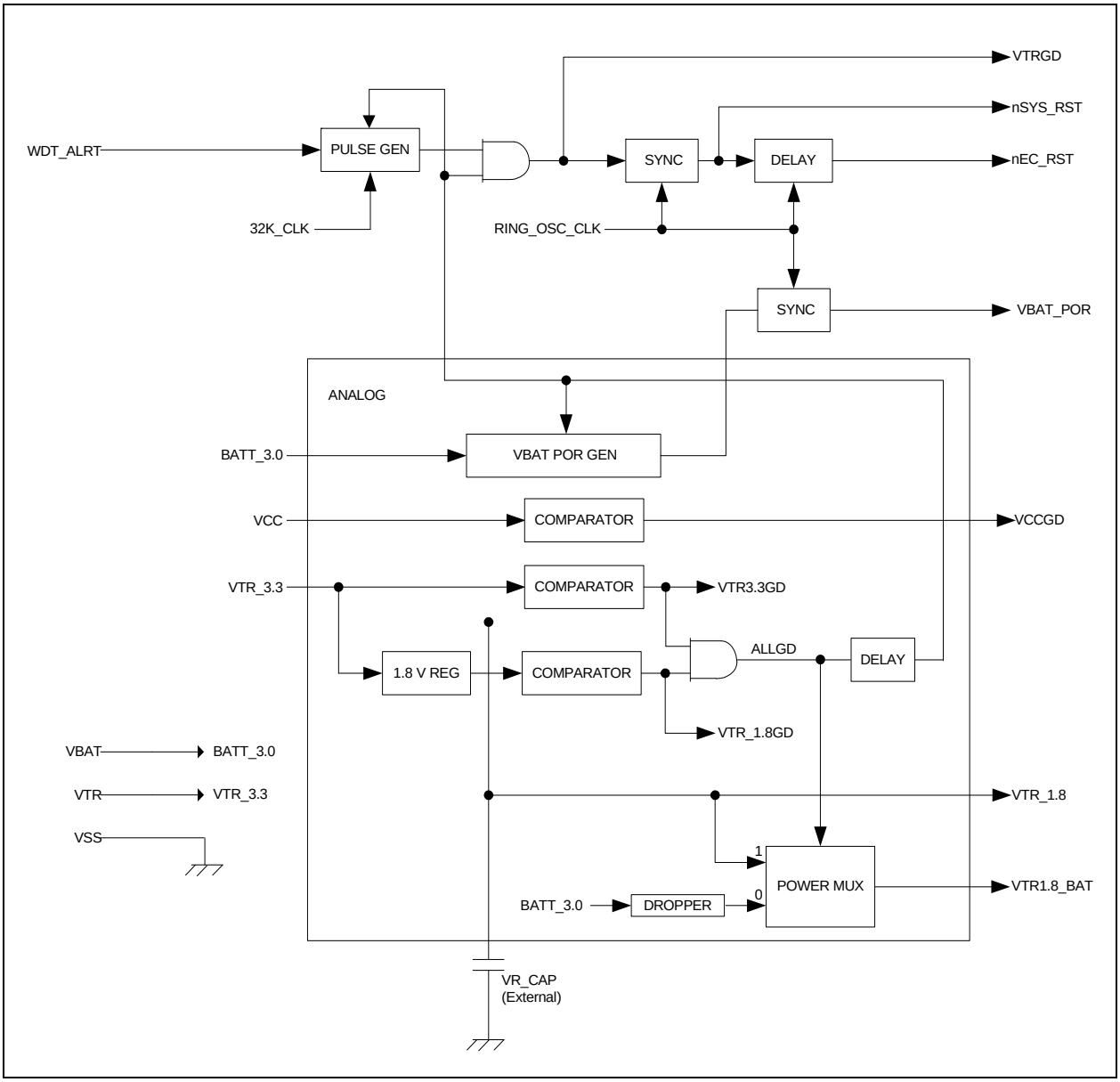
TABLE 5-5: TYPICAL SCH5636 CLOCKS VS. ACPI POWER STATES

Clock Name	ACPI Power State						Description
	S0 (FULL ON)	S1 (POS)	S3 (STR)	S4 (STD)	S5 (Soft Off)	G3 (MECH OFF)	
32K Clock Domain	ON	ON	ON	ON	ON	OFF	This clock domain is generated from the 32K CLK32 clock input, and derived from the 64.52MHz ring oscillator when RSMRST# is asserted.
96MHz PLL Domain	ON	ON	OFF	OFF	OFF	OFF	The 96MHz PLL Domain is gated by the SCH5636 runtime supply (VCC) as described in Section 5.8.4, "96MHz PLL Control," on page 43.
PCI Clock Domain	ON/OFF	ON/OFF	OFF	OFF	OFF	OFF	33MHz LPC Bus clock input powered by the SCH5636 runtime supply (VCC).
EC Bus Clock, Host Bus Clock, Fixed Clock Domain	ON/OFF	ON/OFF	ON/OFF	OFF/ON	OFF/ON	OFF	These clocks are powered by the SCH5636 suspend supply (VTR) but may start and stop as described in Section 5.8, "Power Management Interface," on page 38. (see also Note 5-1)

5.7 Reset Interface

The primary function of the Reset Interface (Figure 5-2) is to generate reset signaling, including, VTRGD, VCCGD, VBAT_POR, nSYS_RST, nEC_RST and nWDT_RST.

FIGURE 5-2: RESET INTERFACE BLOCK DIAGRAM



5.7.1 VTRGD

VTRGD (Figure 5-3) is the reset signal for the 64.52 MHz Ring Oscillator and the source for nSYS_RST and nEC_RST.

As shown in Figure 5-3, Figure 5-4 and in Table 5-6, VTRGD is asserted following a delay after the VTR and VTR_1.2 power supplies exceed preset voltage thresholds. VTRGD is de-asserted as soon as either the VTR or VTR_1.2 power supplies drop below these thresholds (see FIGURE 5-4: VTR Power-Down Timing on page 36).

FIGURE 5-3: VTR POWER-UP TIMING

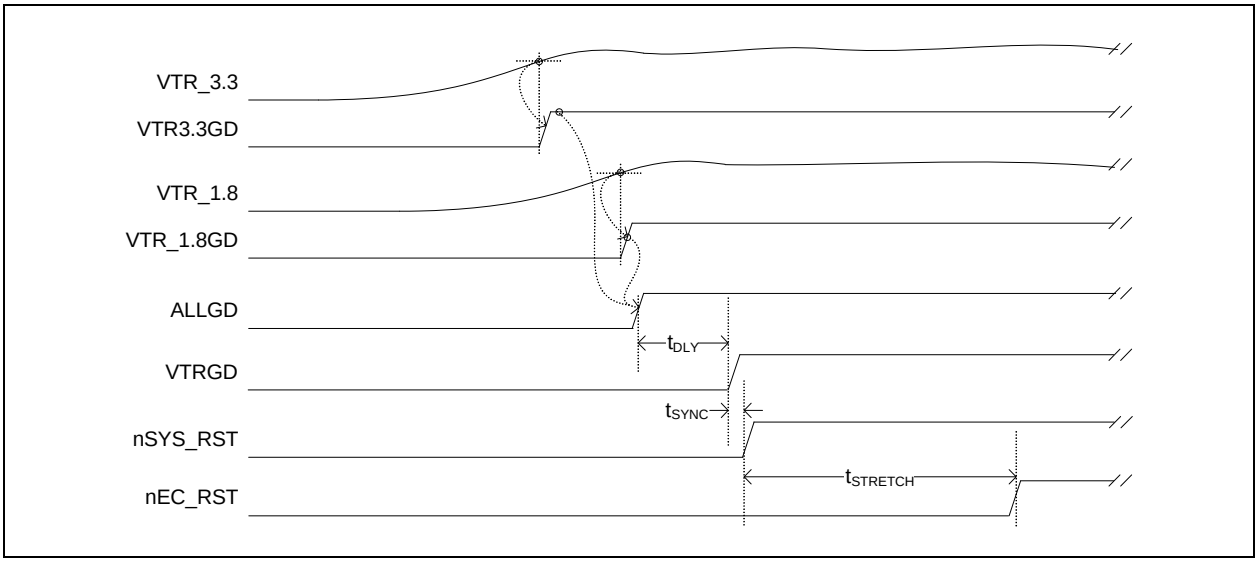
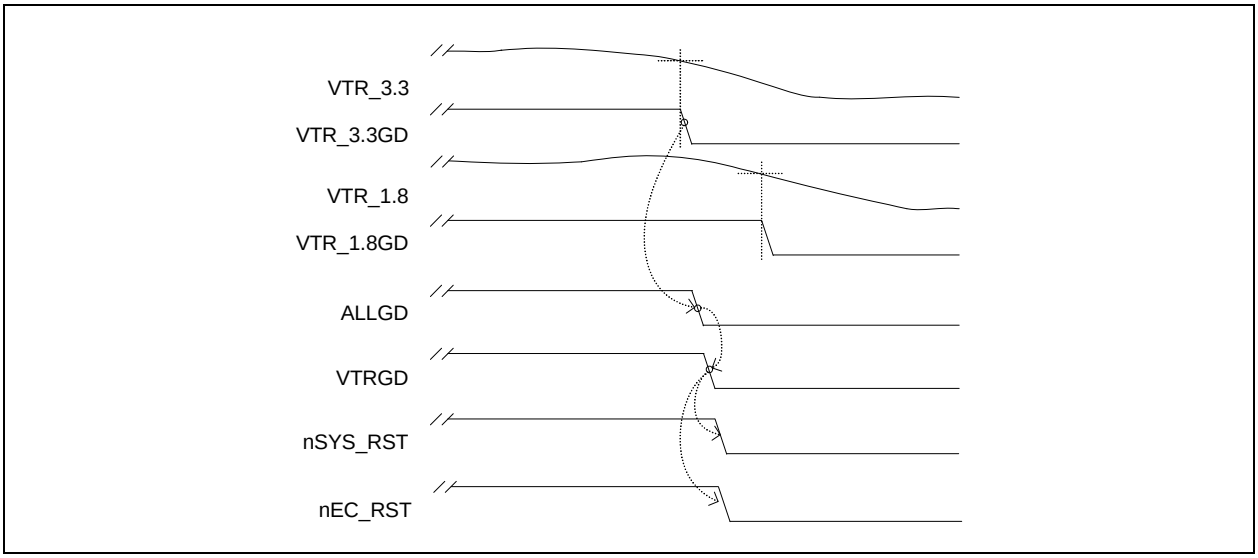


TABLE 5-6: VTR POWER-UP TIMING

Parameters	Symbol	MIN	TYP	MAX	Units	Notes
VTRGD Delay Time	t_{DLY}		360		µs	
nSYS_RST Delay Time	t_{SYNC}	2	–	3	64.52 MHz Ring Oscillator Clocks	Note 5-3
nEC_RST Delay Time	$t_{STRETCH}$	10	20	40	ms	

Note 5-3 this interval is determined using a Fixed Clock Domain from the 64.52 MHz Ring Oscillator.

FIGURE 5-4: VTR POWER-DOWN TIMING



5.7.2 VCCGD

VCCGD is asserted as soon as the VCC input exceeds a preset threshold. VCCGD is de-asserted as soon as the VCC power supply drops below the threshold. VCCGD is used to control the 96MHz PLL and is monitored in the VCC_Low bit in PCR Status Register.

5.7.3 VBAT_POR

VBAT_POR is a pulse that is asserted at the rising edge of VTRGD (Figure 5-3) if the VBAT voltage is below a nominal 1.25V. VBAT_POR is also asserted as a level if, while VTRGD is not asserted (‘0’), the coin cell is replaced with a new cell that delivers at least a nominal 1.25V. In this latter case VBAT_POR is de-asserted after VTRGD is asserted. No action is taken if the coin cell is replaced, or if the VBAT voltage falls below 1.25 V nominal, while VTRGD is asserted

VBAT_POR is used as described throughout this specification to reset registers and functional device blocks. VBAT_POR events are registered in the Power-Fail and Reset Status Register.

5.7.4 NSYS_RST

nSYS_RST is VTRGD synchronized to the 64.52 MHz Ring Oscillator. Note that VTRGD and nSYS_RST have the same logical sense (uninverted); however, because of nomenclature, the asserted states are opposite. Note that VTRGD is defined in Section 5.7.1, "VTRGD," on page 35.

nSYS_RST is de-asserted as defined in Figure 5-3, "VTR Power-Up Timing" and in Table 5-6. nSYS_RST is asserted as soon as either the VTR or VTR_1.2 power supplies drop below preset voltage thresholds (see FIGURE 5-4: VTR Power-Down Timing on page 36).

All VTR-powered blocks are reset on nSYS_RST except for the 64.52 MHz Ring Oscillator and the Embedded Controller.

5.7.5 NEC_RST

nEC_RST is a delayed version of nSYS_RST that is used to reset the Embedded Controller.

nEC_RST is de-asserted as defined in Figure 5-3, "VTR Power-Up Timing" and in Table 5-6. Like nSYS_RST, nEC_RST is asserted as soon as either the VTR or VTR_1.2 power supplies drop below preset voltage thresholds (see FIGURE 5-4: VTR Power-Down Timing on page 36).

nEC_RST is also asserted on nWDT_RST.

5.7.6 NWDT_RST

The nWDT_RST is asserted either when nSYS_RST is asserted, or when the Watchdog Timer (EC Logical Device 1h) asserts the WDT_ALRT signal when the watchdog times out.

The nWDT_RST reset signal asserts nEC_RST, and also resets blocks attached to the EC AHB bus except the Watchdog block itself. Table 5-7, "Logical Devices Reset on nWDT_RST" lists the blocks that are attached to the EC AHB bus and are reset on nWDT_RST:

TABLE 5-7: LOGICAL DEVICES RESET ON NWDT_RST

EC Logical Device Number	AHB Address for Base of Frame	Logical Devices
3h	**F0_0C00h**	16-bit Timer
5h	**F0_1400h**	BC Bus Master
6h	**F0_1800h**	SMBus
9h	**F0_2400h**	DMA
16h	**F0_5800h**	PWM
18h	**F0_6000h**	TACH
19h	**F0_6400h**	PECI
1Ah	**F0_6800h**	EMC3
1Bh	**F0_6C00h**	PROCHOT# Monitor
23h	**F0_8C00h**	MCU Debug Port

TABLE 5-7: LOGICAL DEVICES RESET ON NWDT_RST (CONTINUED)

EC Logical Device Number	AHB Address for Base of Frame	Logical Devices
30h	**F0_C000h**	ARC Interrupts
31h	**F0_C400h**	GPIOs (see Note 5-4)

Note 5-4 Two GPIOs, GP063/KBDRST# and GP064/A20M, are not reset on nWDT_RST. They are only reset on nSYS_RST. All other pins with GPIO functions are reset to their VTR POR state on nWDT_RST.

5.7.7 VCC RESET

The VCC RESET signal is generated when the VCC voltage is below a threshold, described in the definition of Section 5.7.2, "VCCGD" and in the description of the VCC_Low bit in PCR Status Register. VCC Reset is part of nSIO_RESET and is also used to reset some individual registers as noted.

5.7.8 PCI RESET#

The PCI RESET signal is generated whenever the LRESET# pin is low (the status of the LRESET# pin is indicated by the LRESET# bit in the PCR Status Register). PCI RESET# is part of nSIO_RESET.

5.7.9 NSIO_RESET

nSIO_RESET is a signal that is asserted when any one of VTR POR, VCC RESET or PCIRESET# is asserted.

In all blocks listed in Table 5-8, "Logical Devices Reset on nSIO_RESET", all registers within the block are reset on nSIO_RESET except as noted in the block description.

TABLE 5-8: LOGICAL DEVICES RESET ON NSIO_RESET

LPC Logical Device Number	AHB Address for Base of Frame	Logical Devices
1h	**FF_0400h**	8042 Keyboard Controller
7h	**FF_1C00h**	UART 1
8h	**FF_2000h**	UART 2
Bh	**FF_2C00h**	Floppy Disk Controller
Ch	**FF_3000h**	LPC Interface
11h	**FF_3C00h**	Parallel Port

5.8 Power Management Interface

5.8.1 EC SLEEP CONTROL

If it is not required for program execution, the Embedded Controller can save power by entering a sleep state, in which all clocks within the processor are gated except the processor's interrupt controller. The ARC_CLK_DISABLE signal is asserted whenever the processor is in its sleep state. As soon as an interrupt is asserted, the clocks are ungated, the ARC_CLK_DISABLE signal is de-asserted, and the processor wakes up and responds to the interrupt.

The following sequence should be used In order to enter the processor sleep mode:

- Enable Interrupts. Interrupts must be enabled or the ARC processor will not restart after an interrupt or wake event.
- Issue the `SLEEP` instruction

Once this instruction sequence is completed, the ARC processor core will halt and gate its clocks. The Ring Oscillator is not affected and all logical devices will continue to operate, if enabled. As soon as an interrupt occurs, the ARC processor will ungate its clocks and respond to the interrupt. The ARC interrupt mechanism is described in Section 7.0, "EC Interrupt Aggregator," on page 63.

5.8.2 GENERIC BLOCK CLOCKING MODEL

The generic Block Clocking model defines how clocks are distributed to logical devices on both the Host AHB bus and the EC AHB bus, as well as the mechanism that is used to save power by gating the clocks to each block.

FIGURE 5-5: GENERIC BLOCK CLOCKING MODEL ILLUSTRATION

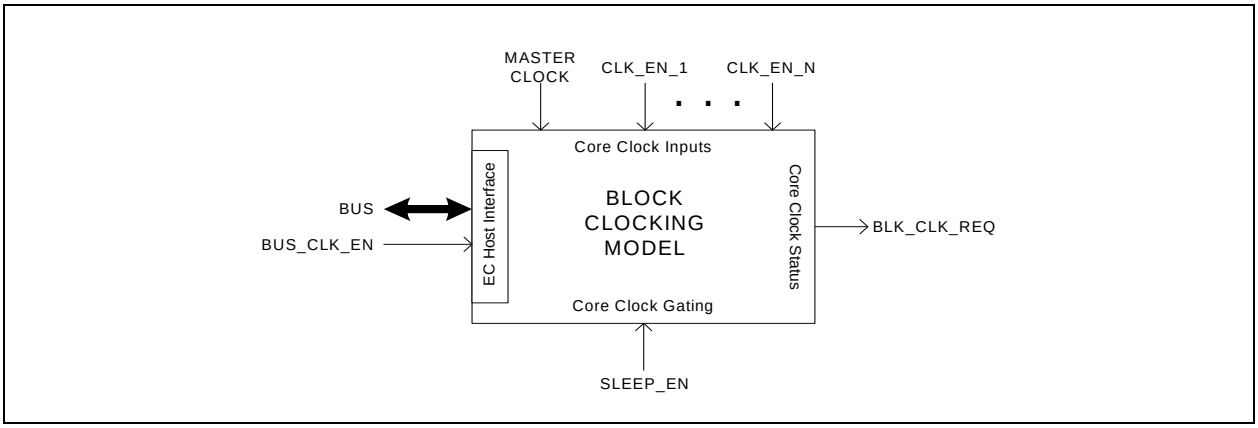


Figure 5-5 illustrates the external interface for each logical device. The generic model includes bus interface, which could be either the Host AHB or the EC AHB, core clock Inputs, which may be clocks or clock enables, logical core clock gating control and a core clock status output. In addition to the signals shown in the figure, each block may also have an Enable or Activate control bit, which when de-asserted resets the block to its initial state, gates its clocks, and if necessary powers down logic associated with the block. De-asserting the Enable/Activate control puts a block into its lowest power state.

De-asserting the Enable/Activate control puts a block into its lowest power state. In addition, most blocks can use the Power Management Interface to save power by gating clocks, without resetting internal registers. Each block as an external sleep enable input (SLEEP_EN), which is asserted when both the bit in the Block Sleep Enable Registers that corresponds to the block is asserted and the global SLEEP_FLAG bit is asserted (see Section 5.9.5, "Block Sleep Enable Registers," on page 49 and Section 5.9.4, "Clock Control Register," on page 48).

When the Power Management Interface asserts the SLEEP_EN to a block, the block may not be able to gate its clocks immediately. For example, a serial output device might be in the process of shifting a byte out onto the external wire and cannot stop in the middle of the byte. For this reason, each block has an internal Idle status bit that reflects the current state of the block. If the internal state is Idle, clocks are not needed and the block will de-assert its BLK_CLK_REQ output when the SLEEP_EN input is asserted. If the internal state is not Idle, the block will continue to operate when SLEEP_EN is asserted, keep BLK_CLK_REQ asserted, until the block becomes Idle. Once Idle, if SLEEP_EN is still asserted (if a , BLK_CLK_REQ is de-asserted and clocks are gated.

The BLK_CLK_REQ output is ‘1’ whenever the block is not disabled and still requires a clock. If clocks are not required, the CLK_REQ output is ‘0’. When no block, including the EC, requires a clock, the main ring oscillator can be shut down to reduce overall power consumption to a minimum.

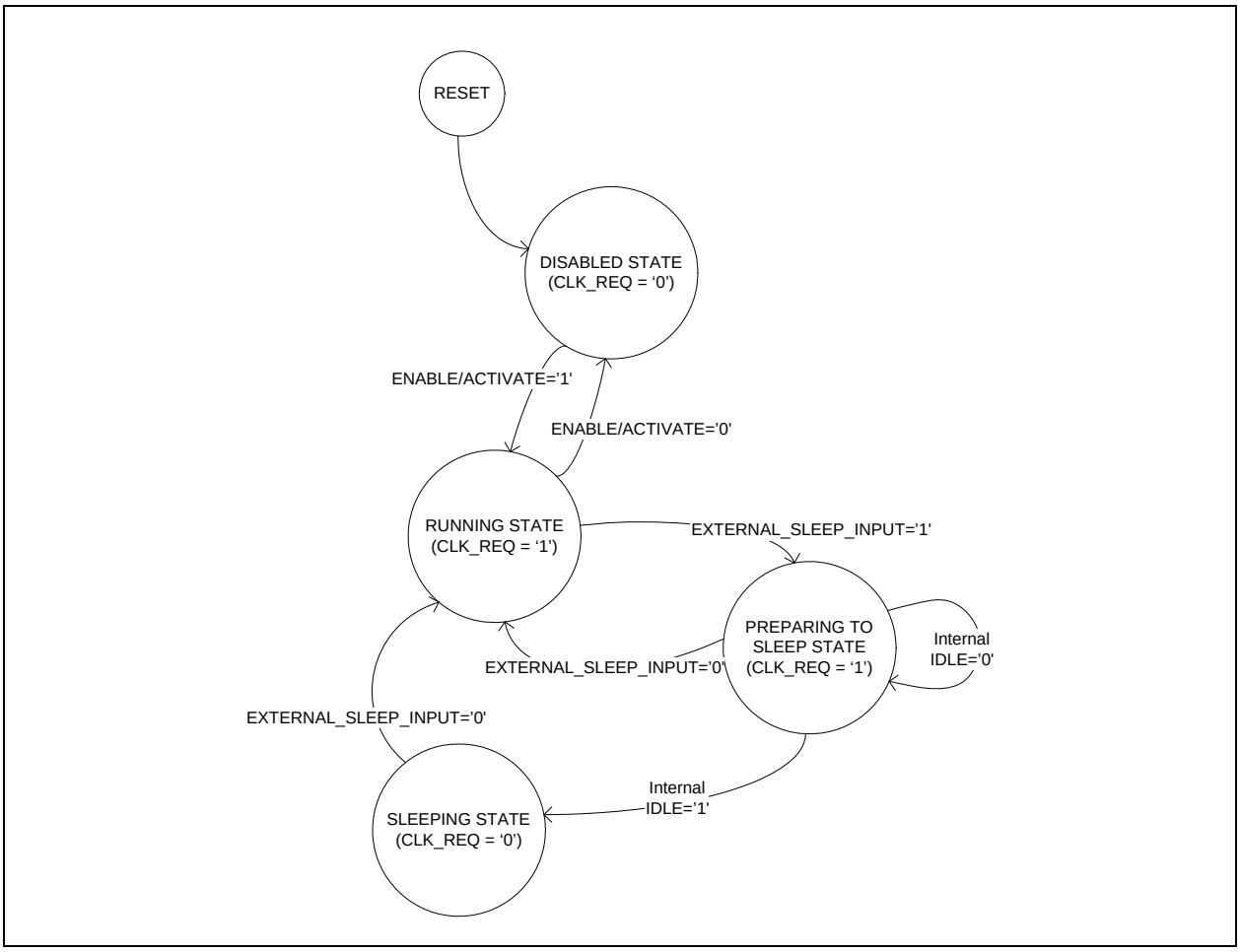
The generic block clocking model states are summarized in Table 5-9:

TABLE 5-9: GENERIC BLOCK CLOCKING MODEL

Enable/ Activate	External SLEEP_EN Input	Internal Block Idle Status	Clock Required Status Output	State	Description
0	X	X	0	DISABLED	Block is disabled by firmware and the core clock is not needed and gated 'off' internally. Internal registers may be reset.
1	0	NOT IDLE	1	FULL POWER	The full power state identifies the block normal operation mode where the block is neither disabled by firmware nor commanded to sleep by the Power Management Interface.
		IDLE			
	1	NOT IDLE		PREPARING TO SLEEP	A sleep command has been asserted but the core clock is still required because the block is not idle.
		IDLE	0	SLEEPING	A sleep command has been asserted, the block is idle and its clocks are gated

Figure 5-6, "Generic Block Clocking Mode Clock Gating State Diagram Example" shows the same conditions in a state transition diagram:

FIGURE 5-6: GENERIC BLOCK CLOCKING MODE CLOCK GATING STATE DIAGRAM EXAMPLE



Note that a block cannot transition directly from the SLEEPING state to the DISABLED state. This is because the EC cannot modify the ENABLE/ACTIVATE bit of a block while the bus clocks are gated.

5.8.3 64.52 MHz Ring Oscillator CONTROL

The 64.52 MHz Ring Oscillator can be controlled by the EC using the Generic Block Clocking Model and the ROSC_CTL control field in the Clock Control Register. As illustrated in Figure 5-7, "64.52 MHz Ring Oscillator Controls" and described in Table 5-10, "ROSC_EN Control", the 64.52 MHz Ring Oscillator remains running when one or more of the following is true:

- At least one block CLK_REQ Status output is asserted
- The ROSC_CTL forces the ring to be on
- The EC is running (ARC_CLK_DISABLE is de-asserted)
- WAKE is asserted by the Wake Interface.

The 64.52 MHz Ring Oscillator will only stop when all block CLK_REQ status outputs are not asserted, the EC is sleeping and WAKE is not asserted, or if ROSC_CTL forces the ring to shut down. The ROSC_CTL control field can force the ring oscillator to stay on or to shut down independent of the Generic Block Clocking Model.

FIGURE 5-7: 64.52 MHZ RING OSCILLATOR CONTROLS

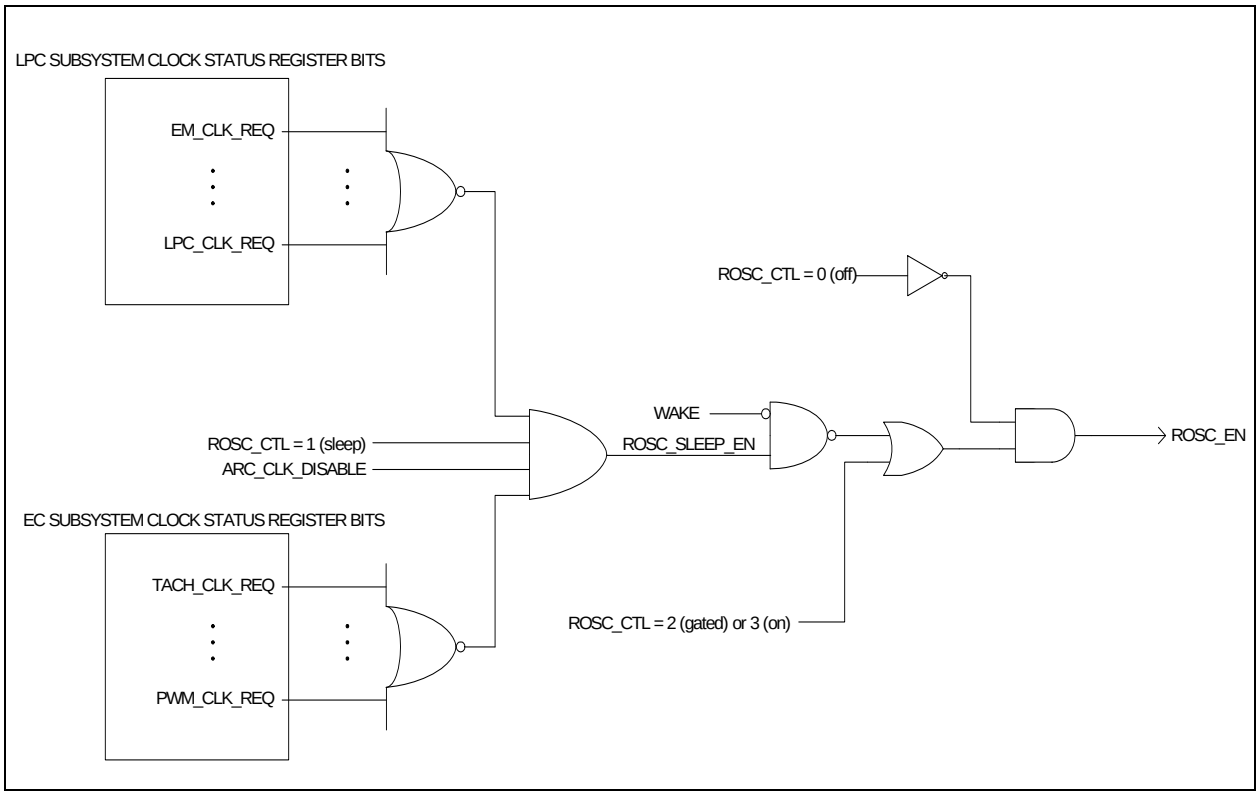


TABLE 5-10: ROSC_EN CONTROL

ROSC_SLEEP_EN	Wake	ROSC_CTL	ROSC_EN	Description
0	X	1	1	The Ring Oscillator is enabled (“on”) because at least one block requires the clock (including the EC) and the EC has not disabled the Ring Oscillator using ROSC_CTL
X	1	1	1	The Ring Oscillator is always enabled (“on”) when WAKE is asserted and the EC has not disabled the Ring Oscillator using ROSC_CTL
1	0	1	0	Ring Oscillator is disabled (“off”) when ROSC_SLEEP_EN is asserted, WAKE is not asserted and the EC has enabled Ring Oscillator sleep control using ROSC_CTL
X	X	0	0	Ring Oscillator is forced off.
1	0	2	0	The Ring Oscillator is on, but the ROSC_EN signal is used to gate the mclk clock tree
X	X	3	1	Ring Oscillator is forced on.

In Table 5-10, the signals ROSC_SLEEP_EN and ROSC_EN are as shown in Figure 5-7. The WAKE signal comes from the wake logic.

5.8.3.1 Entering Sleep States

In order to shut down the ring oscillator (or gate its output at the source), firmware must perform the following steps:

- Set ROSC_CTL in the Clock Control Register to either 1 or 2 (the two settings that enable the Power Management interface)
- Configure the Block Sleep Enable Registers such that clock gating is enabled for all blocks that should be suspended. See Section 5.8.3.2, "Wake Interface" for a description of the Block Sleep Enable mechanism.
- Assert the SLEEP_FLAG in the Clock Control Register. All blocks that are configured to sleep by the Block Sleep Enable Registers enter the PREPARING TO SLEEP state, in which the blocks will gate their clocks as soon as it is safe to do so.
- Enable Interrupts. Interrupts must be enabled or the ARC processor will not restart after an interrupt or wake event.
- Issue the `SLEEP` instruction

If after the sequence is executed all blocks de-assert their CLOCK_REQ status outputs, the Ring Oscillator with either shut down or gate its output, according to the setting in ROSC_CTL. If one or more blocks are not disabled by either their Enable/Activate bit or by their Sleep_En bit in the Block Sleep Enable Registers, the Ring Oscillator will remain operation, although all selected blocks will gate their clocks. Overall power consumption will decrease since at least some blocks are gating their clocks.

If the EC asserts the SLEEP_FLAG but does not issue the `SLEEP` instruction, the selected blocks will gate their clocks but the Ring Oscillator will continue to run, as will the EC. The EC can interrogate the Clock Required Status Registers to estimate the amount of power savings due to clock gating. If a Wake event occurs, the SLEEP_FLAG is cleared and all selected blocks will ungate their clocks.

5.8.3.2 Wake Interface

The Wake interface runs in parallel with the EC Interrupts described in Section 7.0, "EC Interrupt Aggregator," on page 63. All signals that are listed as “Wake Capable” in the tables Section 7.8, "Group Interrupt Source Registers" can generate Wake events. A Wake event restarts the Ring Oscillator if it is shut down, or ungates it if the oscillator is gated. All Wake events also generate EC interrupts, and therefore also wake the EC if it is sleeping.

Wake events only occur if the events are enabled by the matching bits in the GIRQx Enable Registers. Wake events are also only triggered on edges. A GPIO input will only generate a Wake event if its Pin Control Register is configured for edge interrupts by the Interrupt Detection field. If the Interrupt Detection field is configured for a level interrupt, an EC interrupt will be generated but a Wake event will not. Wake events are pulses: they are asserted by hardware for a limited period of time and then automatically de-asserted. The length of the period depends on the source.

5.8.4 96MHZ PLL CONTROL

The 96MHz PLL is controlled by VCCGD and the PLL_CTL control field in the Clock Control Register. See Section 5.9.4, "Clock Control Register," on page 48.

5.9 Registers

The Power, Clocks and Resets registers are instantiated on the EC AHB and are located in two address ranges with two Base Address as indicated in Table 5-11.

TABLE 5-11: POWER, CLOCKS AND RESETS INTERFACE BASE ADDRESS TABLE

Power, Clocks and Resets Blocks	LDN	AHB Base Address
Power, Clock & Reset (VTR PWR’ed)	32h	F0_C800h
Power, Clock & Reset (VBAT PWR’ed)	33h	F0_CC00h

Table 5-12 and Table 5-13 provide a summary of the registers for the Power, Clocks and Resets block.

TABLE 5-12: POWER, CLOCKS AND RESET VTR-POWERED REGISTERS SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
EC Clock Divider Register	-	0h	32	R/W	
LPC_AHB Clock Divider Register	-	4h	32	R/W	
PCR Status Register	-	8h	32	R	
Clock Control Register	-	Ch	32	R/W	
LPC Blocks Sleep Enables Register	-	10h	32	R/W	
EC Blocks Sleep Enables Register 1	-	14h	32	R/W	
LPC Blocks Clock Required Status Register	-	18h	32	R	
EC Blocks Clock Required Status Register 1	-	1Ch	32	R	
OSC_ID Register	-	20h	32	R	
PCR Test Register	-	24h	32	R/W	
EC Blocks Sleep Enables Register 2	-	30h	32	R/W	
EC Blocks Clock Required Status Register 2	-	34h	32	R	

TABLE 5-13: POWER, CLOCKS AND RESET VBAT-POWERED REGISTERS SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Power-Fail and Reset Status Register	-	0h	32	R/W	
Reserved	-	4h	32	R	
Regulator Enable Register	-	8h	32	R/W	

Note: All register addresses are naturally aligned on 32-bit boundaries. Offsets for registers that are smaller than 32 bits are reserved and must not be used for any other purpose.

5.9.1 EC CLOCK DIVIDER REGISTER

The EC Clock Divider Register contains the EC_CLK_DIV bits that are used to program the EC clock and the EC_AHB clock enable frequency as described in the "EC_CLK_DIV" section below.

In the SCH5636, the highest available frequency that can be programmed using the EC Clock Divider Register is 32.25 MHz (EC_CLK_DIV = 2), the lowest is 8 MHz (EC_CLK_DIV = 8). See Section 5.6.1, "EC Bus Clock," on page 33 for a description of EC clocking.

TABLE 5-14: EC CLOCK DIVIDER REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	00h			32-bit			**EC SIZE**	
POWER	VTR			08h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				EC_CLK_DIV			

EC_CLK_DIV

The EC_CLK_DIV bits contain the binary encoded divider that determines the EC clock and EC_AHB clock enable frequency. Valid EC_CLK_DIV values are 2h - Fh. The EC_CLK_DIV default is 08h (8 MHz). Writing a '0h' or a '1h' to the EC Clock Divider Register has no affect.

When the EC_CLK_DIV is greater than '01h,' the EC clock and EC_AHB clock enable frequency (F) is calculated using the equation in Figure 5-8, where MCLK is the undivided output of the 64.52 MHz Ring Oscillator and DIV is the EC_CLK_DIV value programmed into the EC Clock Divider Register.

FIGURE 5-8: EC_CLK_DIV EQUATION

$$F = \frac{MCLK}{DIV}$$

APPLICATION NOTE: To support EC traffic to the LPC Subsystem, the EC_AHB clock frequency must be equal to or less than the LPC_AHB clock frequency.

APPLICATION NOTE: The JTAG clock cannot be higher than 1/2 the EC clock as defined by the EC Clock Divider Register.

MCHP Reserved

These bits must not be modified.

5.9.2 LPC_AHB CLOCK DIVIDER REGISTER

The LPC_AHB Clock Divider Register contains the LPC_AHB_CLK_DIV bits that are used to program the LPC_AHB clock enable frequency as described in the "LPC_AHB_CLK_DIV" section below.

In the SCH5636, the highest available frequency that can be programmed using the LPC_AHB Clock Divider Register is 64.52 MHz (LPC_AHB_CLK_DIV = 1), the lowest is 4.3 MHz (LPC_AHB_CLK_DIV = Fh). As shown in Table 5-15, "LPC AHB Clock Divider Behavior", normal operation is only supported for LPC_AHB_CLK_DIV values 01h - 0Fh.

TABLE 5-15: LPC AHB CLOCK DIVIDER BEHAVIOR

LPC_AHB_CLK_DIV	Frequency (MHZ)	Description
00h	NO CHANGE	Reserved. Writes to the LPC_AHB Clock Divider Register are ignored.
01h - 0Fh	64.52 – 4.30	Normal Operation

TABLE 5-16: LPC AHB CLOCK DIVIDER REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	04h			32-bit			**EC SIZE**	
POWER	VTR			01h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				LPC_AHB_CLK_DIV			

LPC_AHB_CLK_DIV

The LPC_AHB_CLK_DIV is the binary encoded divider that determines the LPC_AHB clock enable frequency. Valid LPC_AHB_CLK_DIV values are 01h - 0Fh. The LPC_AHB_CLK_DIV default is 01h. Writing a '00h' to the LPC_AHB Clock Divider Register has no affect.

When the LPC_AHB_CLK_DIV is greater than '00h,' the LPC_AHB clock enable frequency (F) is calculated using the same equation as for the EC_CLK_DIV (Figure 5-8), where MCLK is the undivided output of the 64.52 MHz Ring Oscillator and DIV is the LPC_AHB_CLK_DIV value. When the LPC_AHB_CLK_DIV is '01h,' the LPC_AHB clock frequency (F) is the undivided output of the 64.52 MHz Ring Oscillator.

APPLICATION NOTE: to support EC traffic to the LPC Subsystem, the EC_AHB clock frequency must be equal to or less than the LPC_AHB clock frequency.

5.9.3 PCR STATUS REGISTER

TABLE 5-17: PCR STATUS REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	08h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE 1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R/	R	R	R	R
BIT NAME	Reserved							14M_ Active
BYTE 0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	LRESET#	MCHP Reserved	FREQ_ LOCK	PWRGD_P S	VCC_ Low	VBAT_ Low	PCICLK_ Active	32K_ Active

32K_Active

This bit monitors the state of the 32K clock input. This status bit detects edges on the clock input but does not validate the frequency.

0: The 32K clock input is not present. The internal 32K clock is derived from the ring oscillator

1: The 32K clock input is present. The internal 32K clock is derived from the pin and the ring oscillator is synchronized to the external 32K clock.

This bit is inverted and sent to bit 6 of GIRQ23, so that an interrupt status bit will be asserted if the 32K clock is not active.

PCICLK_Active

This bit monitors the state of the PCI clock input. This status bit detects edges on the clock input but does not validate the frequency.

0: The 33MHz PCI clock input is not present.

1: The 33MHz PCI clock is present.

VBAT_Low

The VBAT voltage level is sampled when VCC rises above the VCC threshold level. If at that time VBAT is below the low battery threshold (about 2.2V), this bit is set. This bit is a component of the Low_Bat bits in the SMI_STS1 and PME_STS2 runtime registers.

0: The VBAT input is above the voltage trip point

1: The VBAT input is below the voltage trip point

VCC_Low

This bit monitors the VCC threshold detector

0: The VCC input is above the voltage trip point

1: The VCC input is below the voltage trip point

VCC RESET is asserted when this bit is 0. This bit is distinguished from the PWRGD_PS bit in that this bit is derived from a threshold detector on the VCC power input, and PWRGD_PS is derived from the PWRGD_PS pin.

PWRGD_PS

This bit monitors the PWRGD_PS pin. It is synchronized to the 64.52MHz ring oscillator.

FREQ_LOCK

This bit is asserted (‘1’) when the 64.52 MHz Ring Oscillator is within the tightest tolerance to the 32K reference input clock signal, as described in Table 5-3, “64.52 MHz Ring Oscillator Timing Parameters,” on page 33.

LRESET#

This bit reflects the state of the LRESET# pin.

14M_Active

This bit monitors the state of the 14.318MHz clock input. This status bit detects edges on the clock input but does not validate the frequency.

0: The 14.318MHz clock input is not present.

1: The 14.318MHz clock is present.

5.9.4 CLOCK CONTROL REGISTER

TABLE 5-18: CLOCK CONTROL REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	0Ch			32-bit			**EC SIZE**	
POWER	VTR			0500h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				PLL_CTL		ROSC_CTL	
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R/W	R	R	R/W	R
BIT NAME	Reserved			SAA	Reserved		SLEEP_FLAG	Reserved

SLEEP_FLAG

The SLEEP_FLAG affects the system power state as described in Section 5.8.2, "Generic Block Clocking Model," on page 39. The SLEEP_FLAG is R/W. EC firmware asserts SLEEP_FLAG (‘1’), which is then typically de-asserted (‘0’) by hardware as described in Section 5.8.3.2, "Wake Interface," on page 43.

SAA

When asserted (‘1’), the Stop Auto-Adjust bit (SAA) disables automatic frequency correction of the.64.52 MHz Ring Oscillator. When SAA is not asserted (‘0’) (default), the 64.52 MHz Ring Oscillator operates normally as defined in Section 5.5.1, "64.52 MHz Ring Oscillator," on page 32.

Note that when SAA is asserted voltage and temperature variations can adversely affect the frequency of the 64.52 MHz Ring Oscillator. To provide the accuracy of the 64.52 MHz Ring Oscillator as defined in Section 5.5.1, the SAA bit must not be asserted.

APPLICATION NOTE: note that the FREQ_LOCK function as described in Section 5.9.3, "PCR Status Register," on page 46 is undefined when the SAA bit is asserted.

ROSC_CTL

Ring Oscillator control field. The EC can control the 64.52MHz ring oscillator startup latency and clock tree power consumption using the ROSC_CTL field.

0: The ring oscillator is forced to be off.

1:The ring oscillator is on when the power management interface does not assert a sleep state or asserts a wake.

2:The ring oscillator is always on. The mclk clock tree is gated low if the ring oscillator sleep enable power management interface asserts a sleep state.

3:The ring oscillator is always on. The Sleep Enable clock control functions are ignored.

PLL_CTL

96Mhz PLL control field. The EC can control the PLL startup latency and clock tree power consumption using the PLL_CTL field.

0: The PLL is forced to be off.

1:The PLL is on when VCCGD is asserted. The PLL is shut down when VCCGD is de-asserted.

2:The PLL is always on. The PLL clock tree is gated low if VCCGD is de-asserted and not gated when VCCGD is asserted.

3:The PLL is always on. VCCGD is ignored.

5.9.5 BLOCK SLEEP ENABLE REGISTERS

TABLE 5-19: LPC BLOCKS SLEEP ENABLES REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	10h			32-bit			**EC SIZE**	
POWER	VTR			00h			nSYS_RST **DEFAULT**	
BUS	Host SPB							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Runtime
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	MCHP Reserved			Flash_ SPI	MCHP Reserved			EMI

EM

Embedded Memory Interface. See Section 10.0, "Embedded Memory Interface," on page 99.

Flash_SPI

SPI Flash Controller. See Section 15.0, "Serial Peripheral Interface," on page 192.

Runtime

Runtime registers and glue logic. See Section 19.0, "Runtime Registers," on page 198.

MCHP Reserved

These fields must be written with 0.

TABLE 5-20: EC BLOCKS SLEEP ENABLES REGISTER 1

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	14h				32-bit		**EC SIZE**	
POWER	VTR				00_0000h		nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Res	Res	SMB2	SMB1	PWM4	PWM3	PWM2	PWM1
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R/W	R/W
BIT NAME	Res	Res	Res	Res	Res	Res	MBCL	TACH3
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R/W	R/W	R/W	R	R	R/W	R/W	R/W
BIT NAME	TACH2	TACH1	MSDP	Res	Res	C/T3	C/T2	C/T1

TABLE 5-21: EC BLOCKS SLEEP ENABLES REGISTER 2

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	30h				32-bit		**EC SIZE**	
POWER	VTR				0400h		nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R/W	R/W	R	R	R/W	R/W
BIT NAME	–	–	OTP	PHOT	Res	Res	PECI	EMC3
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R/W	R	R	R/W	R	R	R/W	R/W
BIT NAME	DMA	Res	Res	TACH4	Res	Res	PWM6	PWM5

C/T1

16-bit Counter/Timer 1. See Section 24.0, "16-Bit Timer," on page 255.

C/T3

16-bit Counter/Timer 2. See Section 24.0, "16-Bit Timer," on page 255.

C/T3

16-bit Counter/Timer 3. See Section 24.0, "16-Bit Timer," on page 255.

MSDP

MCU Serial Debug Port. See Section 34.0, "Serial Debug Port," on page 329.

TACH1

Tachometer 1. See Section 27.0, "TACH Monitor," on page 278.

TACH2

Tachometer 2. See Section 27.0, "TACH Monitor," on page 278.

TACH3

Tachometer 3. See Section 27.0, "TACH Monitor," on page 278.

TACH4

Tachometer 4. See Section 27.0, "TACH Monitor," on page 278.

MBCL

Master BC-Link™ controller. See Section 32.0, "Microchip BC-Link™ Master," on page 306.

PWM1

PWM controller 1. See Section 28.0, "PWM Controller," on page 286.

PWM2

PWM controller 2. See Section 28.0, "PWM Controller," on page 286.

PWM3

PWM controller 3. See Section 28.0, "PWM Controller," on page 286.

PWM4

PWM controller 4. See Section 28.0, "PWM Controller," on page 286.

SMB1

SMBus Controller 1. Section 26.0, "SMB Device Interface," on page 276.

SMB2

SMBus Controller 2. Section 26.0, "SMB Device Interface," on page 276.

DMA

DMA Controller. See Section 25.0, "DMA Controller," on page 266.

EMC3

EMC3 Analog block. See Chapter 29, "EMC3 Analog Interface," on page 453.

PECI

PECI Temperature monitor. See Section 29.0, "PECI Interface," on page 292.

PHOT

PROCHOT Monitor. See Section 30.0, "PROCHOT# Monitor," on page 294.

OTP

One-time Programmable Memory. See Chapter 36, "OTP," on page 559.

5.9.6 CLOCK REQUIRED STATUS REGISTERS

The Clock Required Status Registers indicates the core clock status per block as defined in Section 5.8.2, "Generic Block Clocking Model," on page 39. Like the Block Sleep Enable Registers, there are two Clock Required Status Registers: the LPC Blocks Clock Required Status Register and the EC Blocks Clock Required Status Register 1.

When a bit in the Clock Required Status Registers is asserted ('1'), the block is enabled and requires that the 64.52 MHz Ring Oscillator remain running as defined in Section 5.8.2, "Generic Block Clocking Model," on page 39.

When a bit in the Clock Required Status Registers is not asserted ('0'), the block is either not enabled as defined in the Generic Block Clocking Model or has been commanded to sleep and no longer requires the 64.52 MHz Ring Oscillator.

TABLE 5-22: LPC BLOCKS CLOCK REQUIRED STATUS REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	18h			32-bit			**EC SIZE**	
POWER	VTR			0Xh			nSYS_RST **DEFAULT**	
BUS	Host SPB							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							Runtime
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved	UART2	Reserved	FLASH SPI	Reserved	UART 1	LPC	EMI

TABLE 5-23: EC BLOCKS CLOCK REQUIRED STATUS REGISTER 1

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	1Ch			32-bit			**EC SIZE**	
POWER	VTR			XX_XXXXh			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Res	Res	SMB2	SMB1	PWM4	PWM3	PWM2	PWM1
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Res	Res	Res	Res	Res	Res	MBCL1	TACH3
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R/W	R	R	R	R	R
BIT NAME	TACH2	TACH1	MSDP	Res	Res	C/T2	C/T1	C/T0

TABLE 5-24: EC BLOCKS CLOCK REQUIRED STATUS REGISTER 2

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	34h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	–	–	OTP	PHOT	Res	Res	PECI	EMC3
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	DMA	Res	Res	TACH4	Res	Res	PWM6	PWM5

5.9.7 OSC_ID REGISTER

The OSC_ID Register contains the SHRINK, FOUNDRY and BLOCK_REVISION identification codes for the 64.52 MHz Ring Oscillator.

TABLE 5-25: OSC_ID REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	20h			32-bit			**EC SIZE**	
POWER	VTR			XXh			**HARDWIRED DEFAULT**	
BUS	EC SPB							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	BLOCK_REVISION				FOUNDRY		SHRINK	

SHRINK

The 2-bit SHRINK register represents the hard-coded 64.52 MHz Ring Oscillator shrink factor.

FOUNDRY

The 2-bit FOUNDRY register represents the hard-coded 64.52 MHz Ring Oscillator foundry code.

BLOCK_REVISION

The 4-bit BLOCK_REVISION register represents the hard-coded 64.52 MHz Ring Oscillator block revision number.

5.9.8 PCR TEST REGISTER

TABLE 5-26: PCR TEST REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	24h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE 0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R	R	R	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved			Sleep_Delay				RO_Wide_ Tol_En

RO_Wide_Tol_En

Enable Ring Oscillator wide tolerance mode (a test mode):

0: enabled

1: disabled

SleeP_Delay

Delay count (in ARC processor clock periods) that a “ring disable” condition has to meet before the ring can be put to sleep.

0: no delay

1: 1 ARC clock delay

15: 15 ARC clock delay

5.10 VBAT-Powered Registers

5.10.1 POWER-FAIL AND RESET STATUS REGISTER

5.10.1.1 Overview

The Power-Fail and Reset Status Register collects and retains event status when VTR is unpowered. Asserted events can cause interrupts as described in Section 5.3, "Interrupts," on page 30.

TABLE 5-27: POWER-FAIL AND RESET STATUS REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	00h			32-bit			**EC SIZE**	
POWER	VBAT			1XXX0X11b			VBAT_POR **DEFAULT**	
BUS	EC SPB							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	-
EC TYPE	R/WC	R	R/WC	R	R	R	R/WC	R/W
BIT NAME	VBAT_ POR	Reserved	WDT	Reserved	PWR_ State	INTRD_ STS	INTRUSION	INTRD_ EN

INTRD_EN

This bit enables operation of the Intruder Detection logic.

0: Intruder detection circuit disabled

1: Intruder detection circuit enabled (VBAT_POR default)

INTRUSION

When the INTRUDER# input goes high-to-low or low-to-high and INTRD_EN is 1, this bit will be set. Software must write a 1’ to clear this bit. Writes of ‘0’ to this bit are ignored. This bit is also set to ‘1’ on VBAT_POR. Writing INTRD_EN, either from 1 to 0 or from 0 to 1, may cause this bit to be set to 1, so if software wishes to disable intrusion detection, it must first set INTRD_EN to 0, then write this register again to set INTRUSION to 0.

See Chapter 10, “Intruder Detection Support,” on page 138 for details on the Intrusion bit.

INTRD_STS

This bit indicates the current state of the INTRUDER# pin.

0: INTRUDER# pin is 0

1: INTRUDER# pin is 1

PWR_State

This bit saves the last state of the SLP_S3# input when the power supply goes off (that is, this bit is set to the state of SLP_S3# on the falling edge of the PWRGD_PS input).

0: Power supply was off (SLP_S3# was asserted) (VBAT POR default)

1: Power supply was on (SLP_S3# was de-asserted)

WDT

The WDT bit is asserted (‘1’) following a Watch-Dog Timer Forced Reset. To clear the WDT bit EC firmware must write a ‘1’ to this bit; writing a ‘0’ to the WDT bit has no affect.

The WDT bit is set only when VTR power is applied, but can be set when the 64.52 MHz Ring Oscillator is off. The VBAT POR bit is set when hardware.

VBAT_POR

The VBAT_POR bit is set to ‘1’ by hardware when a VBAT POR is detected while VTR is off. If a VBAT POR occurred while VTR was off, this bit will default to ‘1’ on a VTR POR. If VBAT is below the battery voltage threshold when VTR rises, this bit will default to ‘1’. If neither condition applies when VTR rises, this bit will remain unchanged. To clear VBAT_POR, EC firmware must write a ‘1’ to this bit; writing a ‘0’ to VBAT POR has no affect.

5.10.2 REGULATOR ENABLE REGISTER

TABLE 5-28: REGULATOR ENABLE REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	08h			32-bit			**EC SIZE**	
POWER	VBAT			0001h			VBAT POR **DEFAULT**	
BUS	EC SPB							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	-
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							VRE

VRE

VTR Regulator Enable. Setting this bit to 0 disables the VTR voltage regulator. Once cleared, it should not be written to a 1. This bit will be set to 1 on the next VBAT POR.

6.0 ARC 625D EMBEDDED CONTROLLER

6.1 General Description

This chapter contains a description of the Embedded Controller used in the SCH5636.

The Embedded Controller on the SCH5636 is an ARC 625D Processor by ARC International. The ARC625D is a full-featured 32-bit embedded processor. Its features include:

- 5-stage instruction pipeline with single-cycle instruction execution
- Static branch prediction
- 32-bit data, instruction and address buses
- 16- and 32-bit instructions, with no overhead for switching between 16- and 32-bits
- 32 32-bit general purpose registers
- Scoreboarded data memory pipeline to reduce data stalls
- Debug features:
  - Debug host can access all registers and CPU memory, with a JTAG interface to host tools
  - Four action points for real-time instruction and data breakpoints
- Industry standard AHB system interface
- Power saving features:
  - Sleep mode via software instruction
  - Clock gating

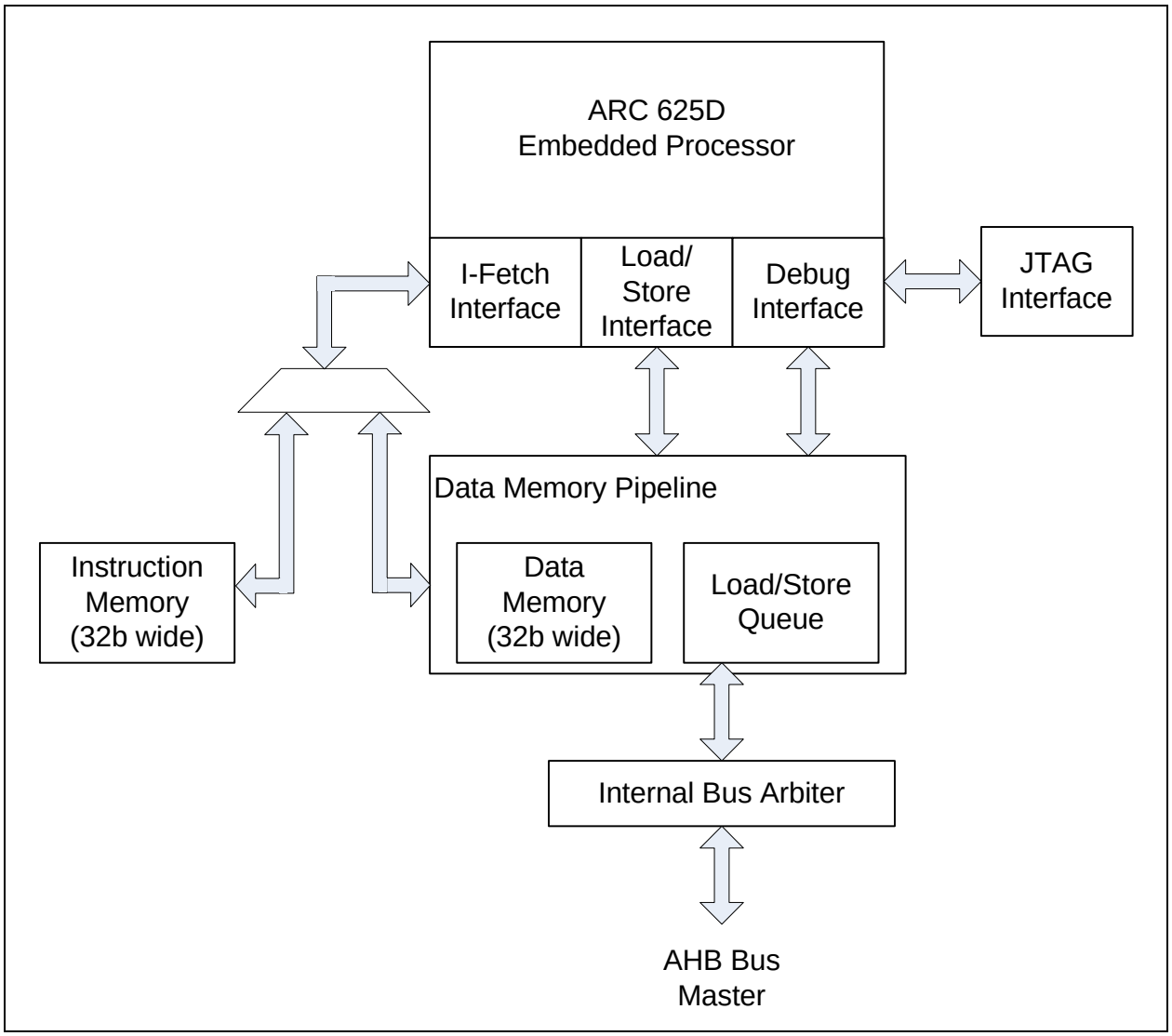
The ARC625D is highly configurable. The configuration used in the SCH5636 incorporates:

- 96K Read-only Closely Coupled instruction memory
- 16KB Single Cycle 32-bit wide dual-ported SRAM, accessible as both Closely Coupled Data Memory and Instruction Memory
- Interrupt controller with 32 interrupts
- Normalize instruction, which can find leading ones and zeros in a word
- Multiply instruction, which completes a 32x32 multiply in 10 cycles
- Four basic Actionpoints (Dedicated Breakpoint Blocks), which can trigger breakpoints on both instruction accesses and data accesses

For details on the architecture of the ARC625D processor, see ARC International's *ARCompact™ Instruction Set Architecture Programmer's Reference*, March 2005.

6.2 Block Diagram

FIGURE 6-1: ARC BLOCK DIAGRAM



6.3 ARC Pipelining

The ARC625D processor is pipelined, with five pipe stages. Loads and stores are further pipelined, through the Load/Store Queue as shown in Figure 6-1, "ARC Block Diagram", so loads and stores will take additional cycles to complete. The AHB bus is also pipelined. Because of the different pipelines, it is difficult to determine exactly how long a load or store to a register will take if the register is located on either the LPC SPB bus or the EC SPB bus.

Because the ARC processor issues all instructions in order and resolves data hazards within the pipeline, software will typically not have to consider pipeline effects. Stores will complete in the order issued, and no load instruction will return data until all stores issued previously have completed. However, there may be some situations in which it is necessary to ensure that pipelines have flushed and all stores have completed before further code execution. The following 3 instruction sequence provides this:

1. STORE to a memory location
2. LOAD from the same memory location to a processor register
3. Any instruction that uses the register in step 2) as one of its sources

The following assembly code is an example of the sequence:

```
; R0 = a value to be written to an AHB memory location
; R1 = the AHB address of the location to be written
;
ST   R0, [R1,0];; store
LD   R0, [R1,0] ; load from same location
ADD  R0, R0, 0 ; dummy instruction dependent on R0
```

6.4 EC Clocking

The EC can be configured to clock at rates ranging from 8 MHz to 32 MHz.

The processor Sleep mode, entered via a software instruction, initiates power saving via Clock gating and ultimately by stopping the 64.52 MHz Ring Oscillator. See Section 5.8, "Power Management Interface," on page 38 for detailed description of Block Clocking Model and power management.

6.5 EC Memory Map

The ARC processor executes code out of the EC Instruction Memory. This instruction memory is an ARC ICCM (Instruction Closely-Coupled Memory) and each 32-bit word can be accessed in one processor cycle. Data references can come from either the EC Data Memory or from addresses located in the AHB Address Space. The EC Data Memory is an ARC DCCM (Data Closely-Coupled Memory), so each 32-bit word can read or written in one cycle.

See ARC Address Space on page 25 for further details on the ARC address space.

6.5.1 EC DATA MEMORY

The EC has a 16KB Closely Coupled Data memory, implemented with static RAM and organized 4K x 32 bits. Loads and stores to this memory are completed in one cycle. The base address of the memory is 80_0000h in the EC address space and extends to location 80_3FFFh. The EC cannot execute instructions from this address range.

The 16KB Data memory also appears in the instruction space in the address range 6_0000h through 6_3FFF as described in Section 6.5.2, "EC Instruction Memory".

6.5.2 EC INSTRUCTION MEMORY

The primary instruction memory for the EC is a 24K x 32 bit read-only memory (ROM), located at locations 00_0000h through 01_FFFFh in the EC address space. Instruction fetches complete in one cycle. The ARC can access locations in the ROM through load and store instructions, with one additional processor cycle penalty.

The 16KB Data memory also appears in the instruction space in the address range 6_0000h through 6_3FFF. The memory is dual-ported, so instruction fetches from this space can occur in parallel with data loads and stores, without wait states for either instruction fetch or data reference.

The RAM_Select bit in the AHB SRAM Configuration Register can be used to disable instruction access to the SRAM. If instruction access to the SRAM is not needed, disabling it saves power.

Instruction fetches in the range of 00_0000h through 7F_FFFFh do not incur bus errors. Any instruction fetch to non-existent memory will return FFFF_FFFFh.

6.6 EC AHB Bus Interface

The ARC Embedded Controller has a single AHB Bus Master interface; see Section 4.2.2, "AHB Address Space," on page 26. The ARC can have at most one access pending on the AHB at one time. The ARC can perform 8-bit, 16-bit and 32-bit loads and stores on the AHB. Instruction fetches over the AHB can take the form of a 32-bit word load.

Possible AHB bus errors are described in Section 4.3.3, "AHB Bus Errors," on page 28. The ARC processor responds to a bus error with Memory Error exception. The first address that caused a memory error is recorded in the AHB Error Address Register. Because ARC exceptions are imprecise, and since several bus errors can occur between the time a bus error address is recorded and the time the ARC processes the exception, it is not always possible to determine which instruction caused a bus error.

6.7 Actionpoints (Dedicated Breakpoint Blocks)

Actionpoints are defined in the ARC 600 Ancillary Components Manual, Chapter 4. They are dedicated hardware blocks that provide an alternative source of breakpoints when the debugger cannot write to memory (e.g., the code being debugged is in ROM). They also provide the ability to break on memory or Aux register accesses.

The primary justification for including Actionpoints in the design is to provide breakpointing for code in ROM, while code is running at full speed (as opposed to being single-stepped). The debugger by default prefers to write Breakpoint instructions (BRK_S: 7FFFh) into memory in order to perform breakpoints at specified PC values. It will instead use actionpoints if:

- The memory area is declared as ROM,

  - or -

- The flag “-off=prefer_soft_bp“is given to the debugger.

Actionpoints are controlled by a dedicated set of Aux Registers, in the range 220h - 237h, organized as 3 registers per actionpoint. These are:

- AMV: A 32-bit value (Address or sometimes Data). This register supplies the initial trigger value, as masked by the AMM register. Upon triggering, it is over-written by hardware with the exact value seen.
- AMM: A 32-bit mask applied to the AMV register, making any desired bits don’t-cares.
- AC: Control register, selecting modes

The status of all actionpoints is visible in the Aux Register DEBUG, at 5h.

The SCH5636 incorporates four basic ARC Actionpoints, Actionpoint 0 through Actionpoint 3.

6.7.1 ACTIONPOINT CONFIGURATION

Actionpoints are configured to do the following:

- Trigger on an access by address and access type (Instruction, Bus access, or Aux Register)
- Instruction breakpoints trigger on execution at the address, not at the fetch itself
- Act by either Halt (debugger acts) or SW Interrupt (target SW acts)
- Qualify between Reads and Writes (or both)
- Qualify by masking bits of the address
- Invert Condition (Trigger if No Match)
- Gang actionpoints in pairs or quads: both/all must match

6.7.2 SIGNIFICANT LIMITATIONS

Address ranges may degrade performance. Because address matching is bit-masked, it may take multiple actionpoints to refine an address range. Even then, the final range is liable to be too big. The debugger allows a range to be too big, and continues from the breakpoint if the resulting trigger was not in the desired range. Note that this means that the program was being halted at undesired / unexpected times, and so is not running at full speed. A reliable way to avoid this is to specify a range only as a power of 2 in size, aligned on a boundary that is also a power of 2, the same or larger than the range.

There is no way to trigger on both the value and the address of a bus read (Memory, I/O), because the data and the address are not present simultaneously. An Aux register access, however, can trigger on data when either read or written. Do not try to enable Read and Write in the same actionpoint, because that will select only the Write data bus to monitor.

6.7.3 DEBUGGER SUPPORT

The debugger supports:

1. Break on Instruction fetch by address
   Actionpoint is used if ROM detected or “-off=prefer_soft_bp” argument is specified
2. Break on Memory Space data accesses
   Read/Write or Both
   Address and Mask Range, if size is power of two and target aligned to a power of two (requires 2 Minimal action-points, paired)
   Value and Mask (requires 2 Minimal actionpoints, paired)
   Value and Range (requires 4 Full actionpoints, quadded)
3. Aux Register accesses
   Read/Write or Both

6.8 EC Registers

The ARC Embedded Controller is instantiated on the EC AHB and has its own Logical Device Number and Base Address, as shown in Table 6-1.

TABLE 6-1: ARC EMBEDDED CONTROLLER BASE ADDRESS TABLE

ARC 625D Embedded Controller **Instance**	**LDN**	**AHB Base Address**
EC Test and Debug	3Fh	F0_FC00h

Table 6-2 is a register summary for the ARC 625D Embedded Controller block.

TABLE 6-2: ARC EMBEDDED CONTROLLER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
AHB SRAM Configuration Register	-	00h	8	R/W	
AHB Error Address Register	-	04h	8	R/WC	

6.8.1 AHB SRAM CONFIGURATION REGISTER

TABLE 6-3: AHB SRAM CONFIGURATION REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	00h					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							RAM_ Select

RAM_Select

When this bit is clear (the default case), the 16KB on-chip SRAM that is part of the EC can only be accessed by loads and stores starting at address 80_0000h. The EC can read and write data in the SRAM at addresses starting at 80_0000h but cannot directly execute instructions.

When this bit is set, the 16KB SRAM is configured to be simultaneously accessible in the address range address 6_0000h through 6_3FFFh. The EC can execute directly out of the SRAM. The EC can still read and write data in the SRAM, with no time penalty per load or store.

6.8.2 AHB ERROR ADDRESS REGISTER

TABLE 6-4: AHB ERROR ADDRESS REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	04h					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	EC_Addr[23:16]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	EC_Addr[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	EC_Addr[7:0]							

EC_Addr[23:0]

If an AHB bus error occurs as the result of an EC AHB bus access, the address that caused the error is held. Once an address is held, additional bus errors are ignored, so this register records the first AHB address that caused an AHB bus error. Any write to this register re-enables capturing AHB bus addresses.

7.0 EC INTERRUPT AGGREGATOR

7.1 General Description

The EC Interrupt Aggregator works in conjunction with the ARC625D processor's interrupt interface to handle hardware interrupts and exceptions.

Exceptions are synchronous to instructions, are not maskable, and have higher priority than interrupts. All three exceptions - reset, memory error, and instruction error - are hardwired directly to the processor. Interrupts are typically asynchronous and are maskable. The EC Interrupt Aggregator forwards 16 output interrupts to the processor's IRQ[8:23].

Interrupts classified as wake events can be recognized without a running clock, e.g., while the SCH5636 is in sleep state.

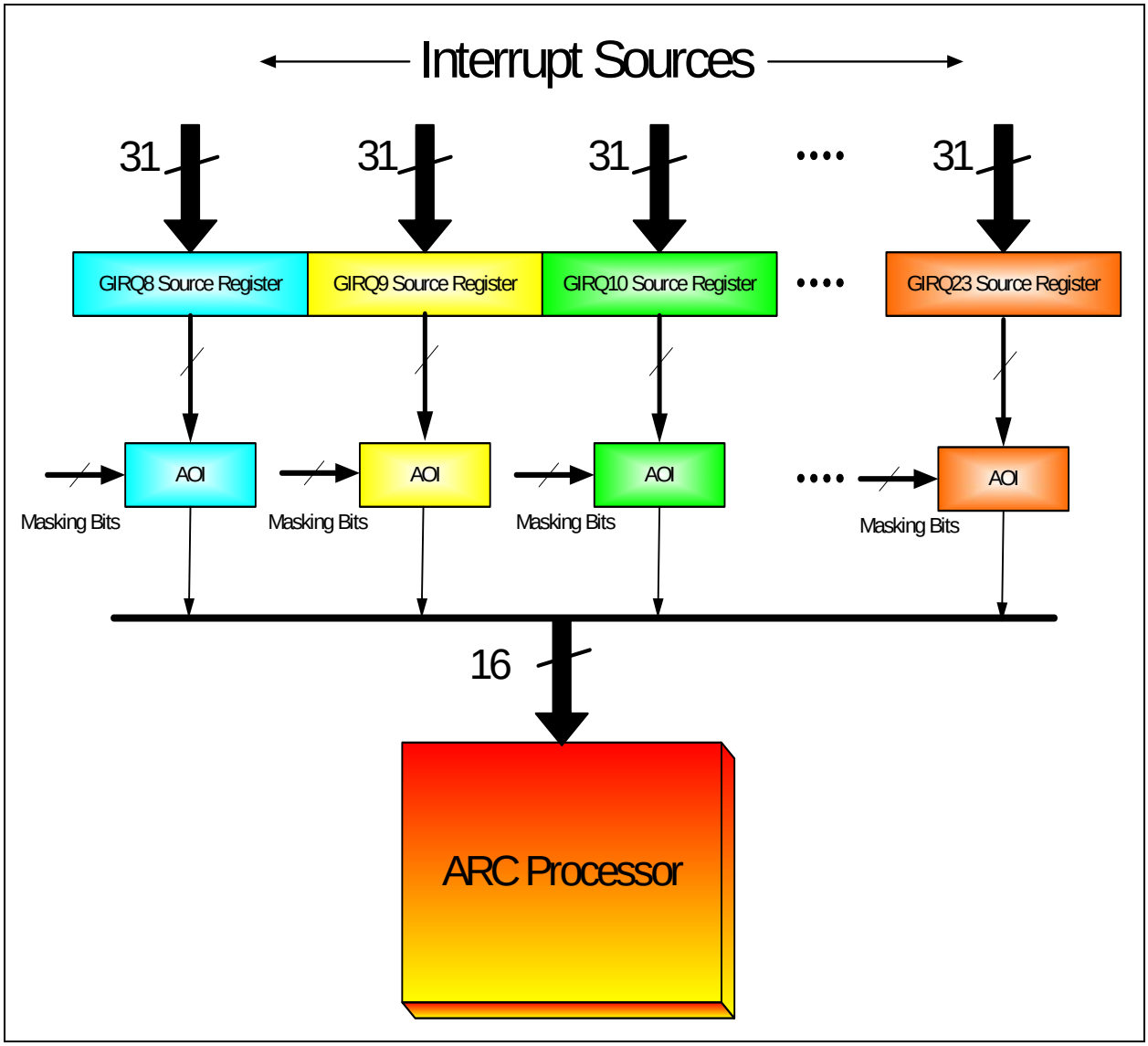
This chapter focuses on the EC Interrupt Aggregator. Please refer to ARC International's *ARCompact™ Instruction Set Architecture Programmer's Reference*, March 2005 for more information on interrupt and exception handling by the ARC625D core.

Features of the EC Interrupt Aggregator:

- Level-triggered inputs
- 16 x 31 input interrupts (31 interrupts per group; 16 groups)
- Wake interrupts recognized while clock is stopped
- Programmable base address of the Vector Table
- Assist fast interrupt handling by software
  - Support for the use of the NORM instruction to quickly locate the highest-level active interrupt.

7.2 Block Diagram

FIGURE 7-1: ARC INTERRUPT GENERATION



7.3 Power, Clocks and Reset

7.3.1 POWER DOMAIN

This block is powered by VTR for wake up capability.

See Section 5.9, "Registers," on page 43 for details on power domains.

7.3.2 CLOCKS

No clock is required for this block.

7.3.3 RESET

This block is reset by nWDT_RST. Following a reset, Interrupt Source, Enable and Result registers default to ‘0’ and all interrupts are Enabled.

See Section 5.9, "Registers," on page 43 for details on reset.

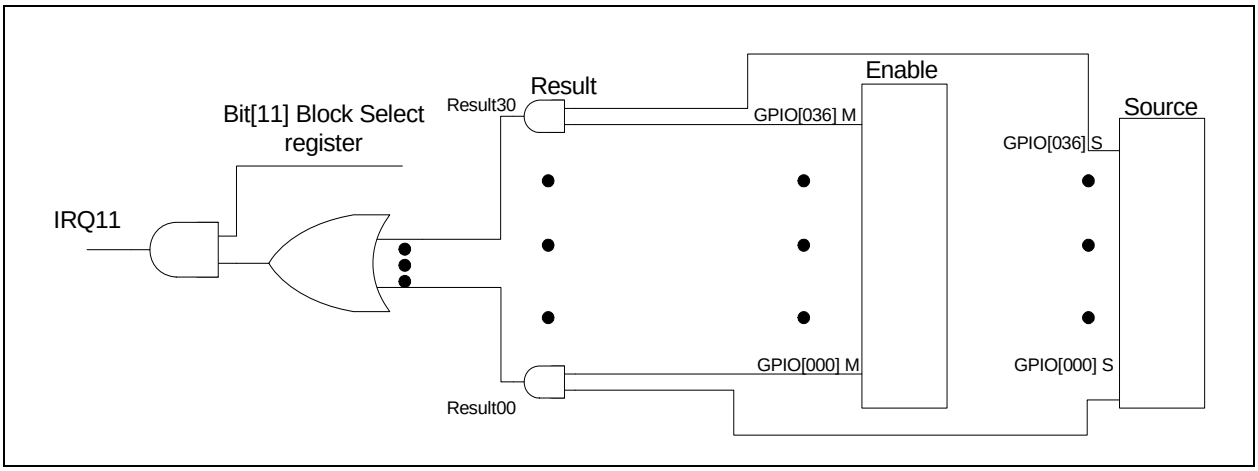
7.4 Interrupt Routing Example

The interrupt generation logic is made of 16 groups of signals, each of which consist of a Status register, a Enable register and a Result register.

The Status and Enable are latched registers. The Result register is a bit by bit AND function of the Source and Enable registers. All the bits of the Result register are OR'ed together and AND'ed with the corresponding bit in the Block Select register to form the interrupt signal that is routed to the ARC interrupt controller.

Figure 7-2 shows one example (IRQ11) of the interrupt signals:

FIGURE 7-2: INTERRUPT STRUCTURE



7.4.1 WAKE GENERATION

The EC Interrupt Aggregator routes logic from WAKE Event Sources to the WAKE input of the Power, Clocks and Resets-Power Management Interface to wake the system. This logic requires no clocks.

The interrupt sources AND'ed with the corresponding Enable bit will be OR'ed to produce a wake event

The wake up sources are identified with a "Y" in the "WAKE" column of the Bit definitions table for each IRQ's Source Register.

7.5 ARC Interrupt Summary

ARC International's *ARCompact™ Instruction Set Architecture Programmer's Reference*, March 2005 describes the ARC processor's response to interrupts. Table 7-1, "EC Interrupt Structure" summarizes the ARC interrupts, priorities and vector locations:

TABLE 7-1: EC INTERRUPT STRUCTURE

Vector	Name	Link Register	Priority (Default)	Relative Priority	Byte Offset
0	Reset	-	High	H1	00h
1	Memory Error	ILINK2	High	H2	08h
2	Instruction Error	ILINK2	High	H3	10h
3	IRQ3-Reserved	ILINK1	level 1 (low)	L27	18h
4	IRQ4-Reserved	ILINK1	level 1 (low)	L26	20h
5	IRQ5-Reserved	ILINK1	level 1 (low)	L25	28h
6	IRQ6-Reserved	ILINK2	level 2 (mid)	M2	30h
7	IRQ7-Reserved	ILINK2	level 2 (mid)	M1	38h
8	IRQ8	ILINK1	level 1 (low)	L24	40h

TABLE 7-1: EC INTERRUPT STRUCTURE (CONTINUED)

Vector	Name	Link Register	Priority (Default)	Relative Priority	Byte Offset
9	IRQ9	ILINK1	level 1 (low)	L23	48h
10	IRQ10	ILINK1	level 1 (low)	L22	50h
11	IRQ11	ILINK1	level 1 (low)	L21	58h
12	IRQ12	ILINK1	level 1 (low)	L20	60h
13	IRQ13	ILINK1	level 1 (low)	L19	68h
14	IRQ14	ILINK1	level 1 (low)	L18	70h
15	IRQ15	ILINK1	level 1 (low)	L17	78h
16	IRQ16	ILINK1	level 1 (low)	L16	80h
17	IRQ17	ILINK1	level 1 (low)	L15	88h
18	IRQ18	ILINK1	level 1 (low)	L14	90h
19	IRQ19	ILINK1	level 1 (low)	L13	98h
20	IRQ20	ILINK1	level 1 (low)	L12	A0h
21	IRQ21	ILINK1	level 1 (low)	L11	A8h
22	IRQ22	ILINK1	level 1 (low)	L10	B0h
23	IRQ23	ILINK1	level 1 (low)	L9	B8h
24	IRQ24	ILINK1	level 1 (low)	L8	C0h
25	IRQ25	ILINK1	level 1 (low)	L7	C8h
26	IRQ26	ILINK1	level 1 (low)	L6	D0h
27	IRQ27	ILINK1	level 1 (low)	L5	D8h
28	IRQ28	ILINK1	level 1 (low)	L4	E0h
29	IRQ29	ILINK1	level 1 (low)	L3	E8h
30	IRQ30	ILINK1	level 1 (low)	L2	F0h
31	IRQ31	ILINK1	level 1 (low)	L1	F8h

Note: IRQ Vector 31 is the highest L1 Priority.

7.5.1 DISABLING INTERRUPTS

Because of pipeline latency, delay through the Load/Store queue and traffic on the AHB bus, writes to SPB registers can potentially take many processor cycles to complete. Because of this latency, the Section 7.8.17 register should not be used for disabling interrupts for software operations like critical sections. Several interrupts could potentially fire between the `STORE` instruction to the interrupt registers and the instruction after the `STORE`.

The ARC `FLAG` instruction is used to modify the E1 and E2 interrupt enable bits in the `STATUS32` register. If the `FLAG` instruction is used, software can insure that no unexpected interrupts will be processed in the middle of a critical section. The following example illustrates how the `FLAG` instruction might be used to implement a critical section:

```
FLAG; 0    ; disable all interrupts
NOP        ; Pipeline Flush
NOP        ; Pipeline Flush
NOP        ; Pipeline Flush
;
; <critical section code here>
;
FLAG; 6;   ; enable all interrupts
```

7.6 Registers

The EC Interrupt Aggregator is instantiated on the EC AHB and has its own Logical Device Number, and Base Address as indicated in Table 7-2. Table 7-3 is a register summary for the EC Interrupt Aggregator block.

TABLE 7-2: INTERRUPT AGGREGATOR BASE ADDRESS TABLE

EC Interrupt Aggregator Blocks	LDN	AHB Base Address
EC Interrupt Aggregator	30h	F0_C000h

TABLE 7-3: INTERRUPT AGGREGATOR REGISTERS SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	TYPE	Bit Definitions
GIRQ8 Source Register		00h	32	R/WC	Table 7-7
GIRQ8 Enable Register		04h	32	R/W	
GRIQ8 Result Register		08h	32	R	
GIRQ9 Source Register		14h	32	R/WC	Table 7-8
GIRQ9 Enable Register		18h	32	R/W	
GIRQ9 Result Register		1Ch	32	R	
GIRQ10 Source Register		28h	32	R/WC	Table 7-9
GIRQ10 Enable Register		2Ch	32	R/W	
GIRQ10 Result Register		30h	32	R	
GIRQ11 Source Register		3Ch	32	R/WC	Table 7-10
GIRQ11 Enable Register		40h	32	R/W	
GIRQ11 Result Register		44h	32	R	
GIRQ12 Source Register		50h	32	R/WC	Table 7-11
GIRQ12 Enable Register		54h	32	R/W	
GIRQ12 Result Register		58h	32	R	
GIRQ13 Source Register		64h	32	R/WC	Table 7-12
GIRQ3 Enable Register		68h	32	R/W	
GIRQ13 Result Register		6Ch	32	R	
GIRQ14 Source Register		78h	32	R/WC	Table 7-13
GIRQ14 Enable Register		7Ch	32	R/W	
GIRQ14 Result Register		80h	32	R	
GIRQ15 Source Register		8Ch	32	R/WC	Table 7-14
GIRQ15 Enable Register		90h	32	R/W	
GIRQ15 Result Register		94h	32	R	
GIRQ16 Source Register		A0h	32	R/WC	Table 7-15
GIRQ16 Enable Register		A4h	32	R/W	
GIRQ16 Result Register		A8h	32	R	
GIRQ17 Source Register		B4h	32	R/WC	Table 7-16
GIRQ17 Enable Register		B8h	32	R/W	
GIRQ17 Result Register		BCh	32	R	
GIRQ18 Source Register		C8h	32	R/WC	Table 7-17
GIRQ18 Enable Register		CCh	32	R/W	
GIRQ18 Result Register		D0h	32	R	
GIRQ19 Source Register		DCh	32	R/WC	Table 7-18
GIRQ19 Enable Register		E0h	32	R/W	

TABLE 7-3: INTERRUPT AGGREGATOR REGISTERS SUMMARY (CONTINUED)

Register Name	Host I/O Offset	SPB Offset	Size	TYPE	Bit Definitions
GIRQ19 Result Register		E4h	32	R	
GIRQ20 Source Register		F0h	32	R/WC	Table 7-19
GIRQ20 Enable Register		F4h	32	R/W	
GIRQ20 Result Register		F8h	32	R	
GIRQ21 Source Register		104h	32	R/WC	Table 7-20
GIRQ21 Enable Register		108h	32	R/W	
GIRQ21 Result Register		10Ch	32	R	
GIRQ22 Source Register		118h	32	R/WC	Table 7-21
GIRQ22 Enable Register		11Ch	32	R/W	
GIRQ22 Result Register		120h	32	R	
GIRQ23 Source Register		12Ch	32	R/WC	Table 7-22
GIRQ23 Enable Register		130h	32	R/W	
GIRQ23 Result Register		134h	32	R	
Block Select Register		200h	32	R/W	Table 7-23

7.7 Interrupt Aggregator Registers

7.7.1 GIRQX SOURCE REGISTERS

TABLE 7-4: GIRQX SOURCE REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	-					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D31**	**D30**	**D29**	• • •		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	RWC except for reserved bits						
BIT NAME	Reserved	See Tables in Section 7.8, "Group Interrupt Source Registers"						

7.7.2 GIRQX ENABLE REGISTERS

TABLE 7-5: GIRQX ENABLE REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	-					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D31**	**D30**	**D29**	• • •		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	RW except for reserved bits						
BIT NAME	Reserved	See Tables in Section 7.8, "Group Interrupt Source Registers"						

7.7.3 GIRQX RESULT REGISTERS

TABLE 7-6: GIRQX RESULT REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	-					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D31**	**D30**	**D29**	• • •		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R						
BIT NAME	NORM	See Tables in Section 7.8, "Group Interrupt Source Registers"						

NORM

This bit is a read write bit used by the NORM instruction

7.8 Group Interrupt Source Registers

7.8.1 GIRQ8

TABLE 7-7: BIT DEFINITIONS GIRQ8 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
[30:0]	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.2 GIRQ9

TABLE 7-8: BIT DEFINITIONS GIRQ9 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
[30:0]	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.3 GIRQ10

TABLE 7-9: BIT DEFINITIONS GIRQ10 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
[7:0]	R/WC	GPIO[047:040]	Y	GPIO Interrupt Signal. This bit is set when a interrupt source is triggered. This bit is sticky; once set, it remains set until cleared by a write with the value ‘1’ to this bit. Write to this bit with a value of ‘0’ have no effect.
[15:8]	R/WC	GPIO[057:050]	Y	
[23:16]	R/WC	GPIO[067:060]	Y	
[28:24]	R/WC	GPIO[074:070]	Y	
[30:29	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.4 GIRQ11

TABLE 7-10: BIT DEFINITIONS GIRQ11 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
[7:0]	R/WC	GPIO[007:000]	Y	GPIO Interrupt Signal. This bit is set when a interrupt source is triggered. This bit is sticky; once set, it remains set until cleared by a write with the value ‘1’ to this bit. Write to this bit with a value of ‘0’ have no effect.
[15:8]	R/WC	GPIO[017:010]	Y	
[23:16]	R/WC	GPIO[027:020]	Y	
[30:24]	R/WC	GPIO[036:030]	Y	
31	R	Reserved	N	This bit is always reserved

7.8.5 GIRQ12

TABLE 7-11: BIT DEFINITIONS GIRQ12 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R/WC	SMB1	N	I^2C/SMBus controller 1 interrupt. This interrupt is signaled when the I^2C/SMBus controller 1 asserts its interrupt request
1	R/WC	SMB2	N	I^2C/SMBus controller 2 interrupt. This interrupt is signaled when the I^2C/SMBus controller 2 asserts its interrupt request
2	R	Reserved	N	Reserved
3	R/WC	SMB1 WK	Y	I^2C/SMBus controller 1 (SMBCLK1/SMBDAT1) Wake interrupt. This interrupt is signaled when there is a transition on SMBDAT1.
4	R/WC	LVSMB1 WK	Y	I^2C/SMBus controller 1 (LVSMBCLK1/LVSMBDAT1) Wake interrupt. This interrupt is signaled when there is a transition on LVSMBDAT1
5	R/WC	SMB2 WK	Y	I^2C/SMBus controller 2 (SMBCLK2/SMBDAT2) Wake interrupt. This interrupt is signaled when there is a transition on SMBDAT2
30-6	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.6 GIRQ13

TABLE 7-12: BIT DEFINITIONS GIRQ13 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
15-0	R	Reserved	N	Reserved
16	R/WC	DMA0	N	DMA Channel 0 interrupt
17	R/WC	DMA1	N	DMA Channel 1 interrupt
18	R/WC	DMA2	N	DMA Channel 2 interrupt
19	R/WC	DMA3	N	DMA Channel 3 interrupt
20	R/WC	DMA4	N	DMA Channel 4 interrupt
30-21	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.7 GIRQ14

TABLE 7-13: BIT DEFINITION GIRQ14 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R	Reserved	N	Reserved
1	R/WC	LRESET#	Y	LRESET Interrupt. This interrupt is signaled when LRESET is asserted.
2	R/WC	LPC_AHB_ERR	N	Either an LPC BAR conflict or an AHB bus error occurred as a result of an LPC access.
15:3	R	Reserved	N	Reserved
16	R/WC	SPI_TXBE_GP	N	General Purpose SPI controller Transmit Buffer Empty Interrupt
17	R/WC	SPI_RXBF_GP	N	General Purpose SPI controller Receive Buffer Full Interrupt
30-18	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.8 GIRQ15

TABLE 7-14: BIT DEFINITION GIRQ15 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R/WC	UART1	N	UART1 Interrupt
1	R/WC	UART2	N	UART2 Interrupt
2	R/WC	EMI	N	EM Interface EC Interrupt
3	R/WC	KBD_KIRQ	N	Keyboard controller keyboard interrupt
4	R/WC	KBD_MIRQ	N	Keyboard controller mouse interrupt
30-5	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.9 GIRQ16

TABLE 7-15: BIT DEFINITION GIRQ16 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R	Reserved	N	Reserved
1	R/WC	EMC	N	EMC interface interrupt
2	R	Reserved	N	Reserved
3	R/WC	PECI	N	PECI Interrupt
7-4	R	Reserved	N	Reserved
8	R/WC	PHOT	N	PROCHOT# Interrupt
30-9	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.10 GIRQ17

TABLE 7-16: BIT DEFINITION GIRQ17 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R/WC	TACH1	N	Fan TACH1 Interrupt. This interrupt is signaled when the Fan TACH1 signals an interrupt.
1	R/WC	TACH2	N	Fan TACH2 Interrupt. This interrupt is signaled when the Fan TACH2 signals an interrupt.
2	R/WC	TACH3	N	Fan TACH3 Interrupt. This interrupt is signaled when the Fan TACH3 signals an interrupt.
3	R/WC	TACH4	N	Fan TACH4 Interrupt. This interrupt is signaled when the Fan TACH4 signals an interrupt.
30-4	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.11 GIRQ18

TABLE 7-17: BIT DEFINITION GIRQ18 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
8:0	R	Reserved	N	Reserved
9	R/WC	BCM_BUSY_CLR	N	BC-Link Busy Clear Flag Interrupt
10	R/WC	BCM_ERR	N	BC_LINK Error Flag Interrupt
11	R/WC	BCM_INT#	Y	Interrupt from the BC_LINK Companion
30-12	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.12 GIRQ19

TABLE 7-18: BIT DEFINITION GIRQ19 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
30-0	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.13 GIRQ20

TABLE 7-19: BIT DEFINITION GIRQ20 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
30-0	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.14 GIRQ21

TABLE 7-20: BIT DEFINITION GIRQ21 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
30-0	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.15 GIRQ22

TABLE 7-21: BIT DEFINITIONS GIRQ22 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R/WC	PWRBTN	N	Change in state of the de-bounced power button input
1	R/WC	Code_Fetch	N	This bit is set to 1 when the Code_Fetch bit in the LED Register in the Runtime Registers is 1
2	R/WC	GPIO_Write	N	This bit is set to 1 when the GPIO_Write field in the GPIO Write Register in the Runtime Registers is written.
30:3	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.16 GIRQ23

TABLE 7-22: BIT DEFINITIONS GIRQ23 SOURCE REGISTER

Bit	EC Type	Signal Name	Wake	Description
0	R/WC	TIMER1	N	16-bit Timer Interrupt
1	R/WC	TIMER2	N	16-bit Timer Interrupt
2	R/WC	TIMER3	N	16-bit Timer Interrupt
4:3	R	Reserved	N	Reserved
5	R/WC	PFR	N	Power Fail Register Interrupt
[30:6]	R	Reserved	N	Reserved
31	R	Reserved	N	This bit is always reserved

7.8.17 BLOCK SELECT REGISTER

TABLE 7-23: BLOCK SELECT REGISTER

HOST ADDRESS							**HOST SIZE**	
EC OFFSET	200h					32-bit	**EC SIZE**	
POWER	VTR					00FF_FF00h	nWDT_RST **DEFAULT**	
BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	IRQ Enable[23:16]							
BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	IRQ Enable[15:8]							
BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R	R	R	R	R	R
BIT NAME	Reserved							

IRQ Enable[31:0]

IRQ Enable. Enable or disable IRQs going to the ARC.

0= Respective IRQ*i* will be disabled.

1= Respective IRQ*i* will be enabled.

8.0 HOST INTERFACE

8.1 General Description

8.1.1 OVERVIEW

The host processor communicates with the SCH5636 via the LPC Bus Interface. The host processor communicates through a series of read/write registers in the SCH5636. Register access is accomplished through programmed I/O or DMA LPC transfer cycles. All I/O transfer cycles are 8 bits wide. DMA transfer cycles can be 16-bits or 8-bits wide.

The Logical Devices located in the SCH5636 are identified in Table 4-2, “Host Logical Devices on SCH5636,” on page 26. The base addresses of logical devices with registers located in LPC I/O space can be moved via the configuration registers located in the LPC Interface Configuration Register Space.

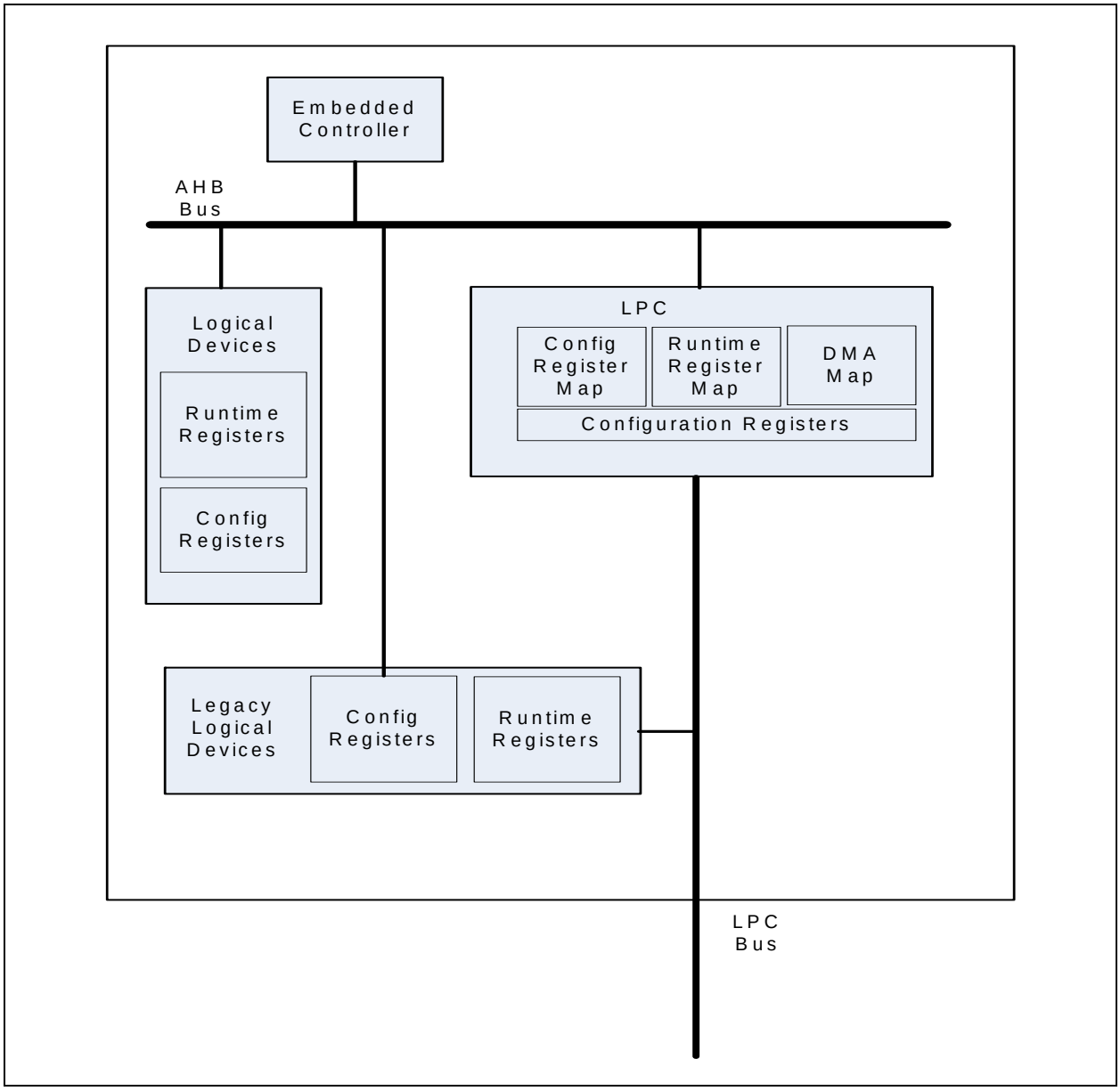
All configuration registers for the SCH5636 are accessed indirectly through the LPC I/O Configuration Register Port (IOCR-Port.) The default I/O address is 2Eh and 2Fh, but the IOCR-Port can be relocated by either the host or the EC. Detailed description of the SCH5636 Configuration Space is in Section 9.0, "Logical Device Configuration," on page 83.

TABLE 8-1: TARGETS OF LPC CYCLES CLAIMED BY THE SCH5636

Target	Acronym	Descriptions	LPC Types
LPC IO Configuration Register Port	**IOCR-Port**	Standard LPC 2Eh/2Fh Port which permits BIOS access. This port can be relocated by the EC or by the Host.	I/O
Logical Devices	**LD**	Targets physically located in the SCH5636.	DMA & I/O
Configuration Register	**CR**	256 byte space per Logical Device accessed by BIOS through the IOCR-Port.	I/O through CR-Port

8.1.2 BLOCK DIAGRAM

FIGURE 8-1: LPC INTERFACE IN SCH5636



8.2 Power

8.2.1 POWER DOMAIN

This block is powered by VTR. Although the block is not powered by VCC, the block is also controlled by PWRGD_PS. When PWRGD_PS is de-asserted, the LPC bus pins are placed in the same state they assume when VTR is off. LAD[3:0] and SERIRQ are tri-stated, LDRQ# is pulled high and LFRAME#, and LRESET# are gated high; see Table 8-2, “LPC Bus Pin Behavior on Reset,” on page 77.The LPC block is also placed in a minimal power state.

See Section 5.9, "Registers," on page 43 for details on power domains.

8.3 LPC Logical Device

Host accesses to Configuration Registers for each Logical Device on the SCH5636are managed by a Configuration block described in Section 9.0, "Logical Device Configuration," on page 83. Configuration registers are accessed through the LPC IO Configuration Register Port.

8.3.1 LPC BUS INTERFACE

The SCH5636 communicates with the host over a Low Pin Count (LPC) interface. The LPC interface uses 3.3V signaling. For detailed specifications, see the *Intel Low Pin Count Specification* and the P*CI Local Bus Specification*, Section 4.2.2. The LPC Bus Interface is listed in Table 3-1, “Signal Descriptions,” on page 11

The following cycle types are supported by the LPC Bus protocol.

- 8-bit I/O Read
- 8-bit I/O Write
- 8-bit DMA Read (for Logical Devices which support 8-bit DMA)
- 8-bit DMA Write (for Logical Devices which support 8-bit DMA)
- 16-bit DMA Read (for Logical Devices which support 16-bit DMA)
- 16-bit DMA Write (for Logical Devices which support 16-bit DMA)

LPC transactions that access registers located on the SCH5636 will require a minimum of two wait SYNCs on the LPC bus. The number of SYNCs may be larger if the internal bus is in use by the embedded controller, of if the data referenced by the host is not present in a SCH5636 register. The SCH5636 always uses Long Wait SYNCs, rather than Short Wait SYNCs, when responding to an LPC bus request.

Table 8-2, "LPC Bus Pin Behavior on Reset", shows the behavior of LPC outputs and input/outputs under reset conditions in accordance with the *Intel Low Pin Count Specification* and the P*CI Local Bus*.

TABLE 8-2: LPC BUS PIN BEHAVIOR ON RESET

Pins	VTR POR (nSYS_RST)	VCC POR	LRESET# Asserted
LAD[3:0]	Tri-state	Tri-state	Tri-State
LDRQ#	Tri-state	De-asserted (high)	De-asserted (high)
SERIRQ	Tri-state	Tri-state	Tri-State

8.3.2 LPC I/O CYCLES

LPC 8-bit I/O Read cycles and 8-bit I/O Write cycles are mapped directly to registers in the SCH5636.

8.3.3 LPC FIRMWARE HUB AND MEMORY CYCLES

The SCH5636 does not support LPC Firmware Hub cycles and LPC Memory cycles on the LPC Bus.

8.3.4 DMA READ AND WRITE CYCLES

DMA read cycles involve the transfer of data from the host (main memory) to the SCH5636. DMA write cycles involve the transfer of data from the SCH5636 to the host (main memory). Data will be coming from or going to a FIFO and will have minimal Sync times. Data transfers to/from the SCH5636 are 1 or 2 bytes.

The mechanism to configure DMA Channels on the SCH5636 is described in Section 9.5, "DMA," on page 87 in the Configuration chapter.

See the *“Low Pin Count (LPC) Interface Specification” Reference*, Section 6.4, for the field definitions and the sequence of the DMA Read and Write cycles.

8.3.4.1 DMA Protocol

DMA on the LPC bus is handled through the use of the LDRQ# lines from the SCH5636 and special encodings on LAD[3:0] from the host.

The DMA mechanism for the LPC bus is described in the “Low Pin Count (LPC) Interface Specification,” Revision 1.0.

8.3.4.2 LDRQ# and SYNC Protocol

DMA transfers are requested through an LDRQ# assertion and ended through a SYNC field.

- Anytime a peripheral has a DMA or bus master channel that needs service, it encodes the channel number on the LDRQ# signal. There is no restriction of having to wait until the CHANNEL field is observed before encoding the next request. The only restrictions on LDRQ encoding are:
- The LDRQ# signal must be inactive for at least 1 clock before starting the next encoding.
- An LDRQ# encoding to request a transfer for a particular channel should not be attempted if one is still pending for that channel.

For single mode DMA transfers:

1. The SCH5636 will use a sync encoding of 0000 to indicate that the data is valid. No data is permitted after the first byte (for channels 0-3) or word (for channels 5-7) since it is a single transfer.
2. After the SCH5636 has observed the CHANNEL field for a particular DMA channel, it can start encoding the next request for that same channel, Requests for other channels can start at any time.

For demand mode DMA transfers:

1. The SCH5636 will use a SYNC encoding of 1001 to indicate additional transfers required. This is functionally equivalent to (and replaces) sending another LDRQ encoding for that channel.
2. The 0000 encoding is used to indicate that the data is valid but it is the last data transfer associated with that demand mode transfer. For example, on the 8^{th} byte in a transfer (which clears a FIFO), the SCH5636 uses the 0000 encoding for SYNC. On the 1^{st} through the 7^{th} bytes, it uses 1001 for SYNC.
3. Once the SCH5636 has used LDRQ# to encode a request for a particular DMA channel to be active, it may not encode another active request for that channel until it has sent the 0000 encoding for SYNC to indicate no more data transfers are needed for that particular demand mode transfer for that channel.

Note:	In 8-bit demand mode, even though the SYNC encoding used is 1001, the next cycle that comes down to the SCH5636 may not be a DMA cycle, it may be an I/O cycle.

See the *"Low Pin Count (LPC) Interface Specification" Reference*, Section 6.4.3 for a description of DMA request deassertion.

For back-to-back transfers from a DMA channel, the following rule applies: The SCH5636 must not assert another message for 8 LCLKs after a de-assertion is indicated through the SYNC field. This applies to transfers on the same DMA channel.

8.3.4.3 Flushing The FIFO

Floppy Disk Controller: Flush at the end of a sector.

Parallel Port: Flush if no data for 2µs.

8.3.4.4 DMA Arbitration

The SCH5636 does not have to arbitrate internally, even though it supports more than one DMA channel. When more than one device requests service, it sends one request out, then the other.

Arbitration for DMA channels is performed through the 8237 within the host. Once the host has won arbitration on behalf of a DMA channel. It asserts LFRAME# on the LPC bus and begins the DMA transfer.

DMA Transfer Types

The DMA protocol is used for all transfer types, including single transfer mode, demand mode and verify mode. For demand mode, the serialized requests will be back-to-back. For verify mode transfers, the SCH5636 should drive data during the appropriate clocks; however, the host may ignore the values. A verify transfer 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. The LPC interface also supports increment mode.

The LPC interface does not support DMA channels being used on cascade mode (for emulating ISA masters). The LPC interface does not support clock or decrement mode.

Channels 0-3 are 8 bit channels. Channels 5-7 are 16 bit channels.

8.3.5 WAIT SYNC'S ON LPC

LPC cycles, with targets physically located in the SCH5636, are completed with no more than **two** LPC Long WAIT SYNC's, provided the internal bus clock is configured to run at the default clock rate of 64.52MHz.

8.3.6 ERROR SYNC'S ON LPC

The SCH5636 does not issue ERROR SYNC cycles.

8.4 LPC Bus Configuration

The mapping from LPC Bus cycles to internal read/write cycles is managed by the LPC Logical Device. The mapping is defined by a series of configuration registers which are defined in Section 9.0, "Logical Device Configuration," on page 83, in Section 8.4, "LPC Bus Configuration," on page 79.

8.5 Serial Interrupts

The SCH5636 supports the serial interrupt scheme, which is adopted by several companies, to transmit interrupt information to the system. The serial interrupt scheme adheres to the *Serial IRQ Specification for PCI Systems Version 6.0*.

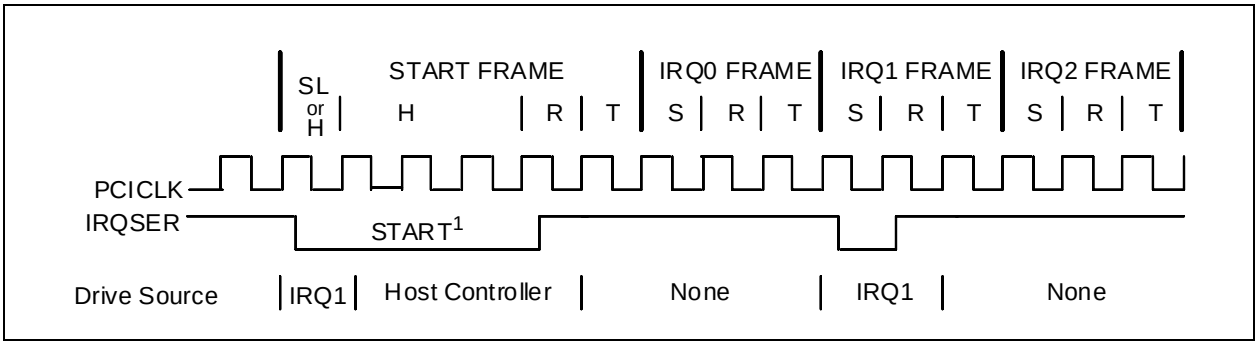
TIMING DIAGRAMS for IRQSER CYCLE

PCICLK = 33 MHz_IN pin

IRQSER = SIRQ pin

Start Frame timing with source sampled a low pulse on IRQ1

FIGURE 8-2: SERIAL INTERRUPTS WAVEFORM "START FRAME"

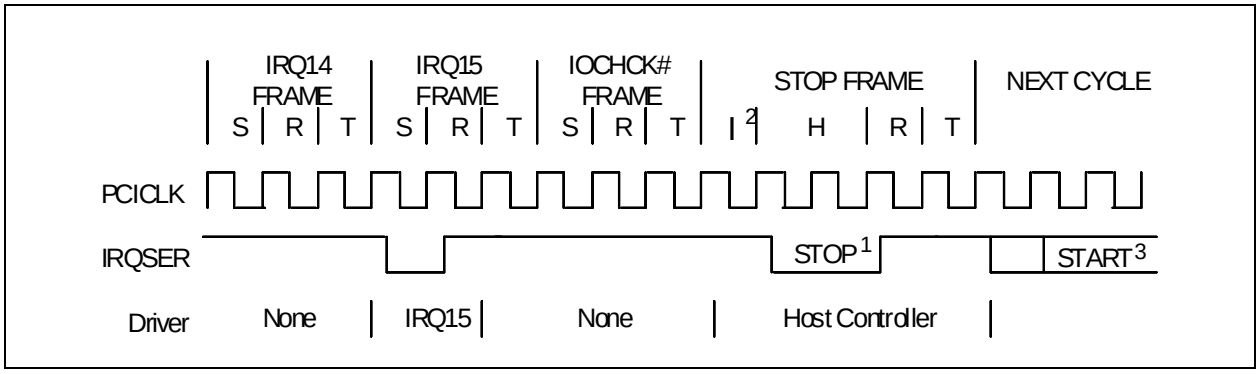


H=Host Control SL=Slave Control R=Recovery T=Turn-around S=Sample

Start Frame pulse can be 4-8 clocks wide.

Stop Frame Timing with Host using 17 IRQSER sampling period.

FIGURE 8-3: SERIAL INTERRUPT WAVEFORM "STOP FRAME"



H=Host Control R=Recovery T=Turn-around S=Sample I= Idle

Stop pulse is two clocks wide for Quiet mode, three clocks wide for Continuous mode.

There may be none, one, or more Idle states during the Stop Frame.

The next IRQSER cycle's Start Frame pulse may or may not start immediately after the turn-around clock of the Stop Frame.

8.5.1 SERIRQ MODE BIT FUNCTION

TABLE 8-3: SERIRQ_EN CONFIGURATION CONTROL

CR25 Bit[2]	Name	Description
0	SERIRQ_EN	Serial IRQ Disabled
1		Serial IRQ Enabled (Default)

8.5.1.1 IRQSER Cycle Control

There are two modes of operation for the IRQSER Start Frame.

Quiet (Active) Mode

Any device may initiate a Start Frame by driving the IRQSER low for one clock, while the IRQSER is Idle. After driving low for one clock, the IRQSER must immediately be tri-stated without at any time driving high. A Start Frame may not be initiated while the IRQSER is active. The IRQSER is Idle between Stop and Start Frames. The IRQSER is active between Start and Stop Frames. This mode of operation allows the IRQSER 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 IRQSER low in the next clock and will continue driving the IRQSER 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 IRQSER back high for one clock then tri-state.

Any IRQSER Device (i.e., The SCH5636) 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 IRQSER is already in an IRQSER Cycle and the IRQ/Data transition can be delivered in that IRQSER Cycle.

Continuous (Idle) Mode

Only the Host controller can initiate a Start Frame to update IRQ/Data line information. All other IRQSER agents become passive and may not initiate a Start Frame. IRQSER will be driven low for four to eight clocks by host controller. This mode has two functions. It can be used to stop or idle the IRQSER or the host controller can operate IRQSER in a continuous mode by initiating a Start Frame at the end of every Stop Frame.

An IRQSER mode transition can only occur during the Stop Frame. Upon reset, IRQSER bus is defaulted to continuous mode, therefore only the host controller can initiate the first Start Frame. Slaves must continuously sample the Stop Frames pulse width to determine the next IRQSER Cycle's mode.

IRQSER Data Frame

Once a Start Frame has been initiated, the SCH5636 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 SCH5636 must drive the IRQSER (SIRQ pin) low, if and only if, its last detected IRQ/Data value was low. If its detected IRQ/Data value is high, IRQSER must be left tri-stated. During the recovery phase, the SCH5636 must drive the SERIRQ high, if and only if, it had driven the IRQSER low during the previous sample phase. During the turn-around phase, the SCH5636 must tri-state the SERIRQ. The SCH5636 drives the IRQSER 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, then the sample phase is {(6 x 3) - 1 = 17} the seventeenth clock after the rising edge of the Start Pulse.

TABLE 8-4: IRQSER SAMPLING PERIODS

IRQSER Period	Signal Sampled	# Of Clocks Past Start
1	Not Used	2
2	IRQ1	5
3	nSMI/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

The SIRQ data frame will now support IRQ2 from a logical device; previously IRQSER Period 3 was reserved for use by the System Management Interrupt (nSMI). When using Period 3 for IRQ2, the user should mask off the SCH5636's SMI via the ESMI Mask Register. Likewise, when using Period 3 for nSMI, the user should not configure any logical devices as using IRQ2.

IRQSER Period 14 is used to transfer IRQ13. Each Logical devices will have IRQ13 as a choice for their primary interrupt.

Stop Cycle Control

Once all IRQ/Data Frames have completed, the host controller will terminate IRQSER activity by initiating a Stop Frame. Only the host controller can initiate the Stop Frame. A Stop Frame is indicated when the IRQSER is low for two or three clocks. If the Stop Frame's low time is two clocks, then the next IRQSER cycle's sampled mode is the Quiet mode; and any IRQSER 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, then the next IRQSER 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.

Latency

Latency for IRQ/Data updates over the IRQSER bus in bridge-less systems with the minimum IRQ/Data Frames of 17 will range up to 96 clocks (3.84µS with a 25 MHz PCI Bus or 2.88µs with a 33 MHz 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.

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 IRQSER Cycle latency in order to ensure that these events do not occur out of order.

AC/DC Specification Issue

All IRQSER agents must drive/sample IRQSER synchronously related to the rising edge of the PCI bus clock. The IRQSER (SIRQ) pin uses the electrical specification of the PCI bus. Electrical parameters will follow the PCI Specification Section 4, sustained tri-state.

Reset and Initialization

The IRQSER bus uses LRESET as its reset signal and follows the PCI bus reset mechanism. The IRQSER pin is tri-stated by all agents while LRESET is active. With reset, IRQSER slaves and bridges are put into the (continuous) Idle mode. The host controller is responsible for starting the initial IRQSER 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 IRQSER cycles. It is the host controller's responsibility to provide the default values to the 8259's and other system logic before the first IRQSER cycle is performed. For IRQSER system suspend, insertion, or removal application, the host controller should be programmed into Continuous (IDLE) mode first. This is to make sure the IRQSER bus is in Idle state before the system configuration changes.

9.0 LOGICAL DEVICE CONFIGURATION

9.1 Description

The Configuration of the SCH5636 is very flexible and is based on the configuration architecture implemented in typical Plug-and-Play components.

The SCH5636 is designed for motherboard designs in which the resources required by their components are known. With its flexible resource allocation architecture, the SCH5636 allows the BIOS to assign resources at POST.

9.2 Logical Devices

Logical devices described in this section are peripherals that are located on the SCH5636 and are accessible to the Host over the LPC bus.

Each logical device on the SCH5636 can have a set of Runtime Register and a set of Configuration Registers. The distinction between Runtime and Configuration registers is that the Host can access Runtime Registers by a direct I/O address, while it can only access Configuration Registers through a configuration port. The Logical Device Numbers for the Logical Devices resident in the SCH5636 are listed in Table 4-2, “Host Logical Devices on SCH5636,” on page 26.

9.3 Configuration Registers

9.3.1 HOST ACCESS PORT

The Host can access Configuration Registers through a port described in Section 9.3.2, on page 83. Host accesses are limited to 8 bits. There are 48 8-bit Global Configuration Registers (at offsets 00h through 2Fh), plus up to 208 8-bit registers associated with each Logical Device. The Logical Device is selected with the Logical Device Number Register (Global Configuration Register 07h). 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.

Only two states are defined (Run and Configuration). In the Run State, the chip will always be ready to enter the Configuration State.

The desired configuration registers are accessed in two steps:

a) Write the index of the Logical Device Number Configuration Register (i.e., 07h) to the INDEX PORT and then write the number of the desired logical device to the DATA PORT

b) 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 1: If accessing the Global Configuration Registers, step (a) is not required.

2: Any write to an undefined or reserved Configuration register is terminated normally on the LPC bus without any modification of state in the SCH5636. Any read to an undefined or reserved Configuration register returns FFh.

9.3.2 PRIMARY CONFIGURATION ADDRESS DECODER

The logical devices are configured through three Configuration Access Ports (CONFIG, INDEX and DATA). The BIOS uses these ports to initialize the logical devices at POST (Table 9-1).

The Base Address of the Configuration Access Ports is determined by the BAR that corresponds to Logical Device Ch, the LPC Interface. The Configuration Access Port BAR is unique in that an LPC I/O access that matches this BAR does not directly generate an internal read or write. Instead, the Device and Frame values in the BAR indicates that the LPC I/O should be handled locally in the LPC Logical Device. The Configuration map will issue an internal read or write, the results of which will be used to complete the LPC access.

TABLE 9-1: SCH5636 CONFIGURATION ACCESS PORTS

Port Name	Relative Address	Type	Port Name
CONFIG PORT	Configuration Access Ports Base Address + 0	Write	CONFIG PORT
INDEX PORT	Configuration Access Ports Base Address + 0	Read/Write	INDEX PORT
DATA PORT	Configuration Access Ports Base Address + 1		DATA PORT

9.3.2.1 Entering the Configuration State

The INDEX and DATA ports are effective only when the chip is in the Configuration State. The device enters the Configuration State when the Config Entry Key is successfully written to the CONFIG PORT.

Config Entry Key = < 55h>

9.3.2.2 Exiting the Configuration State

The device exits the Configuration State when the following Config Exit Key is successfully written to the CONFIG PORT address.

Config Exit Key = < AAh>

9.3.2.3 Read Accessing Configuration Port

The data read from the Configuration Port is undefined when not in the Configuration State. Writing the Config Entry Key puts the chip in the Configuration State. Once in the Configuration State, reading the Configuration Port will return the last value written to the Configuration Index. If no value was written the Configuration Port reads 00h.

9.3.3 CONFIGURATION SEQUENCE EXAMPLE

To program the configuration registers, the following sequence must be followed:

1. Enter Configuration State
2. Program the Configuration Registers
3. Exit Configuration State.
4. The following is an example of a configuration program in Intel 8086 assembly language.

```
;----------------------------.
; ENTER CONFIGURATION STATE
;----------------------------'
MOV    DX,CONFIG_PORT_BASE_ADDRESS
MOV    AX,055H; Config Entry Key
OUT    DX,AL
;----------------------------.
; CONFIGURE BASE ADDRESS,    |
; LOGICAL DEVICE 8           |
;----------------------------'
MOV    DX,CONFIG_PORT_BASE_ADDRESS
MOV    AL,07H
OUT    DX,AL; Point to LD# Config Reg
MOV    DX,CONFIG_PORT_BASE_ADDRESS+1
MOV    AL, 08H
OUT DX,AL; Point to Logical Device 8
;
MOV    DX,CONFIG_PORT_BASE_ADDRESS
MOV    AL,60H
OUT    DX,AL  ; Point to BASE ADDRESS REGISTER
MOV    DX,CONFIG_PORT_BASE_ADDRESS+1
MOV    AL,02H
OUT    DX,AL  ; Update BASE ADDRESS REGISTER
;----------------------------.
; EXIT CONFIGURATION STATE
;----------------------------'
MOV    DX,CONFIG_PORT_BASE_ADDRESS
MOV    AX,0AAH; Config Exit Key
OUT    DX,AL.
```

9.3.4 CONFIGURATION REGISTER ADDRESS MAPPING

The INDEX PORT defines 256 bytes for configuration. The first 48 of these bytes are Global Configuration registers, which reside in the first 48 bytes of the Configuration part of the address frame for Logical Device 3Fh. Values of INDEX greater than 48 map into registers that are specific to the Logical Device specified in the Global Configuration Logical Device Number Register 7h. These registers reside in upper 20 bytes of the Logical Device address frame. See Section 9.8.2, on page 91 for details.

9.4 Configuring Runtime Register Addresses

9.4.1 RUNTIME REGISTERS

Runtime Registers are registers that are accessible to the Host within the Host I/O address space. These Host I/O accesses are all mapped into the SCH5636 internal address space onto devices located on the Host SPB. Runtime registers all reside within the first 256 bytes of a 1KB Logical Device address frame. The Host accesses these registers with 8-bit LPC I/O accesses. Each 8-bit I/O address is mapped into an 8-bit address in the internal address space, so the first 256 bytes of the Logical Device frame can accommodate 256 LPC Runtime Registers per Logical Device. The Host I/O addresses are determined by a block of Base Address Registers located in the LPC Logical Device. The Embedded Controller can access all the Runtime Registers as well.

9.4.2 BASE ADDRESS REGISTERS

Each Logical Device has a Base Address Register (BAR). On every LPC bus I/O access all Base Address Registers are checked in parallel and if any matches the LPC I/O address the SCH5636 claims the bus cycle.

Note:	Software should that insure that no two BARs map the same LPC I/O address.

Each BAR is 32 bits wide. The format of each BAR is summarized in Table 9-2, "Base Address Register Format".

TABLE 9-2: BASE ADDRESS REGISTER FORMAT

BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
BIT NAME	LPC Host Address, most significant bits							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
BIT NAME	LPC Host Address, least significant bits							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
BIT NAME	Valid	Device	Frame					
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
BIT NAME	Reserved	Mask						

MASK

These 7 bits are used to mask off address bits in the address match between an LPC I/O address and the Host Address field of the BARs, as described in Section 9.4.3, "Mapping LPC I/O Addresses". A block of up to 128 8-bit registers can be assigned to one base address.

FRAME

These 6 bits are used to specify a logical device frame number within a bus. This field is multiplied by 400h to provide the frame address within the peripheral bus address. Frame values for frames corresponding to logical devices that are not present on the SCH5636 are invalid.

DEVICE

This bit combined with FRAME constitute the Logical Device Number. DEVICE identifies the physical location of the logical device. This bit should always be set to 0.

VALID

If this bit is 1, the BAR is valid and will participate in LPC matches. If it is 0 this BAR is ignored

HOST_ADDRESS

These 16 bits are used to match LPC I/O addresses

9.4.3 MAPPING LPC I/O ADDRESSES

bit A Base Address Register will match an LPC I/O address, and thus the SCH5636 will claim the LPC bus cycle, if the following relation holds:

(LPC Address & ~BAR.MASK) == (BAR.LPC_Address & ~BAR.MASK) && (BAR.Valid == 1)

If one of the BARs match, the LPC cycle will be claimed by the SCH5636. The Logical Device number for the matching device is located in the Frame field of the BAR.When matching LPC I/O addresses, the SCH5636 ignores address bits that correspond to '1b' bits in the MASK field. For example, the Keyboard Controller (9042 Interface) Base Address Register has 60h in the LPC Address field, the Frame field is 01h, and the MASK field is 04h. Because of the single '1b' bit in MASK, the BAR will match LPC I/O patterns in the form '00000000011000hb', so both 60h and 64h will be matched and claimed by the SCH5636.

As another example, if a standard 16550 UART was located at LPC I/O address 238h, then the UART Receive buffer would appear at address 238h and the Line Status register at 23Dh. If the BAR for the UART was set to 0238_8047h, then the UART will be matched at I/O address 238h. UART1 is located in Logical Device 7h and the UART device includes 8 registers.

9.4.4 BASE ADDRESS REGISTER TABLE

Table 9-3, "Base Address Registers Default Values", lists the Base Address Registers for all logical devices on the SCH5636 base chip. The columns to the right of the heavy line show the field definitions for the default values listed in the column labeled "Reset Default".Shaded fields in Table 9-3 are read-only. The OFFSET column shows the index within the LPC Logical Device's Configuration register space for each BAR.

The shaded LPC I/O Address, VALID, DEVICE, FRAME, MASK fields are read-only Table 9-3. The unshaded fields has read/write access.

TABLE 9-3: BASE ADDRESS REGISTERS DEFAULT VALUES

Offset	Reset Default (see Note 9-1)	LPC I/O Address	Valid	Device	Frame	Masks	Description
60h	002E_0C01h	002Eh	0	0	C	1	Logical Device 0Ch: LPC Interface (Configuration Port)
64h	0000_000Fh	0000h	0	0	0	F	Logical Device 00h: EM Interface
68h	0000_0707h	0000h	0	0	7	7	Logical Device 07h: UART 1
6Ch	0000_0807h	0000h	0	0	8	7	Logical Device 08h: UART 2
70h	0000_0A3Fh	0000h	0	0	A	3F	Logical Device 0Ah Runtime Registers
74h	0000_0F07h	0000h	0	0	F	7	Logical Device Fh Flash SPI
78h	0060_0000h	0060h See Note 9-3	0	0	0 See Note 9-2	0 See Note 9-2	Logical Device 01h: Keyboard Controller (8042 Interface)

TABLE 9-3: BASE ADDRESS REGISTERS DEFAULT VALUES (CONTINUED)

Offset	Reset Default (see Note 9-1)	LPC I/O Address	Valid	Device	Frame	Masks	Description
7Ch	0000_0000h	0000h	0	0	0 See Note 9-2	0 See Note 9-2	Logical Device 0Bh: Floppy Disk Controller
80h	0878_8000h	0000h	1	0	0 See Note 9-2	0 See Note 9-2	Logical Device 11h: Parallel Port

Note 9-1 All BAR registers except the BAR at 60h (LPC Logical Device 0Ch, the LPC Interface) are reset on VTR POR, VCC RESET or LRESET. The BAR at 60h, LPC Logical Device 0Ch, is reset on VTR POR only.

Note 9-2 The FRAME and MASK fields for these Legacy devices are not used to determine which LPC I/O addresses to claim. The address range match is maintained within the blocks themselves.

Note 9-3 The Keyboard Controller contains registers are offset +0h, +4h (for the keyboard interface) and +32h (for legacy GateA20/KBDRST). The default BAR for the keyboard controller, set for 60h, therefore provides for the standard 60h/64h and 92h ports for keyboard control. Any address assigned to this BAR must have bit[2] and bit[5] equal to 0.

9.5 DMA

9.5.1 DMA CONFIGURATION REGISTERS

The SCH5636 will claim an LPC DMA request if the requested channel is listed as valid in the Table 9-4, "DMA Configuration Register Map". A channel is claimed if the DMA Configuration Register Format that corresponds to the channel is maps to a Logical Device. In order to execute the DMA operation, the SCH5636 translates the DMA access into a bus read or write of the FIFO that corresponds to channel in question. The address of a DMA FIFO will always be one of the first 16 32-aligned addresses within the DMA quadrant of a Logical Device frame.

The mapping in the DMA Configuration Register Map is used both for mapping LPC DMA I/O requests from the Host to Logical Devices, as well as for mapping DMA requests from Logical Devices to the LPC Bus LDRQ# DMA request signal.

The Host can access the DMA Configuration registers with 8-bit accesses.

TABLE 9-4: DMA CONFIGURATION REGISTER MAP

Offset	Type	Reset	Configuration Register Name
50h	R/W	0000h	DMA Channel 0
52h	R/W	0000h	DMA Channel 1
54h	R/W	0000h	DMA Channel 2
56h	R/W	0000h	DMA Channel 3
58h	R	0000h	DMA Channel 4 (Reserved)
5Ah	R/W	0000h	DMA Channel 5
5Ch	R/W	0000h	DMA Channel 6
5Eh	R/W	0000h	DMA Channel 7

Note 9-4 DMA Channel 4 is reserved in the SCH5636. LPC Host cycles with DMA channel 4 asserted will be unclaimed by the SCH5636.

TABLE 9-5: DMA CONFIGURATION REGISTER FORMAT

BYTE1 BIT	D15	D14	D13	D12	D11	D10	D9	D8
BIT NAME	Valid	Device	Frame					
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
BIT NAME	Reserved				Offset			

OFFSET

This field should always be set to 0.

FRAME

These 6 bits Logical Device number for the DMA target.

DEVICE

This field should always be set to 0.

VALID

If this bit is 1, the DMA Channel is active on the SCH5636. If it is 0 this DMA Channel is ignored.

9.6 SERIRQ Interrupts

The SCH5636 can routes Logical Device interrupts onto SIRQ stream frames IRQ[0:15]. Routing is controlled by the SIRQ Interrupt Configuration Registers. There is one SIRQ Interrupt Configuration Register for each accessible SIRQ Frame (IRQ); all 16 registers are listed in Table 9-6, "SIRQ Interrupt Configuration Register Map". Each SIRQ Interrupt Configuration Register controls a series of multiplexors which route to a single Logical Device interrupt as illustrated in FIGURE 9-1: SIRQ Routing Internal Logical Devices on page 90. The format for each SIRQ Interrupt Configuration Register is described in Table 9-7. Each Logical Device can have up to two LPC SERIRQ interrupts. When the SCH5636 is polled by the host, each SIRQ frame routes the level of the Logical Device interrupt (selected by the corresponding SIRQ Interrupt Configuration Register) to the SIRQ stream.

Note: Two Logical Devices cannot share a Serial IRQ.

The SIRQ Interrupt Configuration Register The Host can access the Interrupt Configuration registers with 8-bit accesses.

Note: A SERIRQ interrupt is deactivated by setting an entry in the SIRQ Interrupt Configuration Register Map to FFh, which is the default reset value.

9.6.1 SERIRQ CONFIGURATION REGISTERS

TABLE 9-6: SIRQ INTERRUPT CONFIGURATION REGISTER MAP

Offset	Type	Reset	Configuration Register Name
40h	R/W	FFh	IRQ0
41h	R/W	FFh	IRQ1
42h	R/W	FFh	IRQ2 (nSMI)
43h	R/W	FFh	IRQ3
44h	R/W	FFh	IRQ4
45h	R/W	FFh	IRQ5
46h	R/W	FFh	IRQ6
47h	R/W	FFh	IRQ7
48h	R/W	FFh	IRQ8

TABLE 9-6: SIRQ INTERRUPT CONFIGURATION REGISTER MAP (CONTINUED)

Offset	Type	Reset	Configuration Register Name
49h	R/W	FFh	IRQ9
4Ah	R/W	FFh	IRQ10
4Bh	R/W	FFh	IRQ11
4Ch	R/W	FFh	IRQ12
4Dh	R/W	FFh	IRQ13
4Eh	R/W	FFh	IRQ14
4Fh	R/W	FFh	IRQ15

Note 9-5 The SIRQ Interrupt Configuration Registers are though the Host Access Port as 8-bit accesses. The EC can access the SIRQ Interrupt Configuration Registers as 32-bit, 16-bit across 8-bit boundary or as individual 8-bit accesses.

TABLE 9-7: SIRQ INTERRUPT CONFIGURATION REGISTER FORMAT

BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
BIT NAME	Select	Device	Frame					

FRAME

These six bits select the Logical Device as the source for the interrupt.

This field defaults to 3Fh

DEVICE

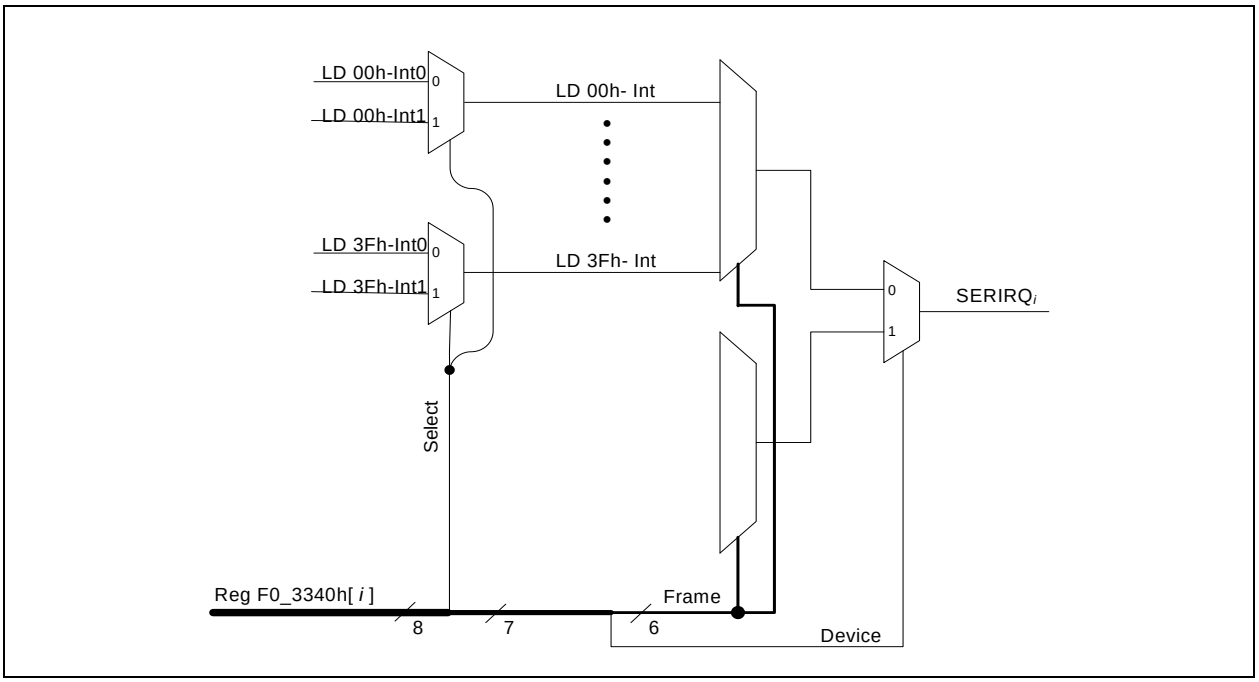
This field should always be set to 0 in order to enable a SERIRQ.

SELECT

If this bit is 0, the first interrupt signal from the Logical Device is selected for the SERIRQ vector. If this bit is 1, the second interrupt signal from the Logical Device is selected.This field is ignored if the Logical Device has only one interrupt signal.

This field defaults to 1.

FIGURE 9-1: SIRQ ROUTING INTERNAL LOGICAL DEVICES



9.6.1.1 SIRQ Routing

TABLE 9-8: LOGICAL DEVICE SIRQ ROUTING

SIRQ Interrupt Configuration Register			
SELECT	DEVICE	FRAME	**Logical Device Interrupt Source**
0	0	0h	EMI SIRQ, Mailbox register - See Section 10.3, on page 99
1	0	0h	EMI SIRQ, Interrupt source register - See Section 10.3, on page 99
0	0	1h	Keyboard SIRQ - Section 11.5.3, on page 112
1	0	1h	Mouse SIRQ - Section 11.5.3, on page 112
0	0	7h	UART1 SIRQ - Section 12.3.1, on page 124
0	0	8h	UART2 SIRQ - Section 12.3.1, on page 124
0	0	Ah	PME from Runtime Registers - Section 19.3.15, on page 209
1	0	Ah	SMI from Runtime Registers - Section 19.3.15, on page 209
0	0	Bh	Floppy SIRQ - Section 14.10, on page 186
0	0	11h	Parallel Port SIRQ - Section 13.2.13, on page 153

9.7 Configuration Register Reset Conditions

There are two reset conditions that will cause Configuration Registers on the SCH5636 to reset to default values. A reset can be caused by a VTR Power On Reset condition (signaled by nSYS_RST) or by an nSIO_RESET condition. The conditions that cause nSIO_RESET to be asserted are defined in Section 5.7.9, "nSIO_RESET," on page 38. In addition, firmware running on the Embedded Controller can set all Configuration Registers to a default condition.

9.8 Logical Device Configuration/Control Registers

A separate set of control and configuration registers exist for each Logical Device and is selected with the Logical Device # Register (07h). The Logical Devices are listed in Table 4-2, “Host Logical Devices on SCH5636,” on page 26, and the registers within each Logical Device are listed in Section 9.8.2, on page 91.

9.8.1 LOGICAL DEVICE ACTIVATION

Many Logical Devices have a register, called Activate, that is used to activate the Logical Device. When a Logical Device is inactive, it is powered down. The format for the Activate Register is shown in Table 9-9, "Activate Register".

Activating a Logical Device does not cause the SCH5636 to claim LPC addresses associated with the device. Address matching for all Logical Devices is enabled or disabled in the LPC Logical Device, as described in Section 9.4, "Configuring Runtime Register Addresses," on page 85.

TABLE 9-9: ACTIVATE REGISTER

HOST OFFSET	BYTE0: 30h					8-bit	**HOST SIZE**	
POWER	VTR					00b	nSYS_RST **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Activate

Activate

When this bit is 1, the logical device is powered and functional. When this bit is 0, the logical device is powered down and inactive.

9.8.2 CONFIGURATION REGISTER MAP

The SCH5636 Configuration register map is shown in Table 9-10, "SCH5636 Configuration Register Map". Logical Device numbers are in hexadecimal. All Logical Devices are accessible by both the Host and the EC. Logical Devices may be numbered between 00h and 3Fh.

TABLE 9-10: SCH5636 CONFIGURATION REGISTER MAP

LPC CR Index	EC Offset	Type Note 9-6	Reset Note 9-7	Configuration Register Name
Configuration Registers for LDN 0h (EM Interface)				
-	-	-	-	None
Configuration Registers for LDN 1h (Keyboard Controller)				
30h	330h	R/W	00h on nSIO_RESET	Activate
F0h	3F0h	R/W	00h on nSIO_RESET	KRST_GA20
F1	3F1h	R/W	00h on nSIO_RESET	Keyboard Select
Configuration Registers for LDN 7h (UART1)				
30h	330h	R/W	00h on nSYS_RST/ nSIO_RESET (see Note 9-8)	Activate Register
F0h	3F0h	R/W	00h on nSYS_RST	Configuration Select Register
Configuration Registers for LDN 8h (UART2)				
30h	330h	R/W	00h on nSYS_RST/ nSIO_RESET (see Note 9-8)	Activate Register

TABLE 9-10: SCH5636 CONFIGURATION REGISTER MAP (CONTINUED)

LPC CR Index	EC Offset	Type Note 9-6	Reset Note 9-7	Configuration Register Name
F0h	3F0h	R/W	00h on nSYS_RST	Configuration Select Register
Configuration Registers for LDN Ah **(**Runtime Registers**)** **(See** Table 9-15, "Runtime Registers, Logical Device Ah"**)**				
F0h	3F0h	R/W	00h on nSYS_RST	SPEKEY
Configuration Registers for LDN Bh **(**Floppy Disk Controller**)** **(See** Table 9-16, "Floppy Disk Controller, Logical Device Bh"**)**				
30h	330h	R/W	00h	Activate Register
F0h	3F0h	R/W	0Eh	FDD Mode Register
F1h	3F1h	R/W	00h	FDD Option Register
F2h	3F2h	R/W	FFh	FDD Type Register
F3h	3F3h	R	00h	Reserved
F4h	3F4h	R/W	00h	FDD0
F4h	3F5h	R/W	00h	FDD1
Configuration Registers for LDN Ch **(**LPC Interface**)**				
30h	330h	R/W	00h on nSIO_RESET	Activate Register
40h	340h	R/W	FFh on nSIO_RESET	SIRQ IRQ0 Configuration Register
41h	341h	R/W	FFh on nSIO_RESET	SIRQ IRQ1 Configuration Register
42h	342h	R/W	FFh on nSIO_RESET	SIRQ IRQ2 (nSMI) Configuration Register
43h	343h	R/W	FFh on nSIO_RESET	SIRQ IRQ3 Configuration Register
44h	344h	R/W	FFh on nSIO_RESET	SIRQ IRQ4 Configuration Register
45h	345h	R/W	FFh on nSIO_RESET	SIRQ IRQ5 Configuration Register
46h	346h	R/W	FFh on nSIO_RESET	SIRQ IRQ6 Configuration Register
47h	347h	R/W	FFh on nSIO_RESET	SIRQ IRQ7 Configuration Register
48h	348h	R/W	FFh on nSIO_RESET	SIRQ IRQ8 Configuration Register
49h	349h	R/W	FFh on nSIO_RESET	SIRQ IRQ9 Configuration Register
4Ah	34Ah	R/W	FFh on nSIO_RESET	SIRQ IRQ10 Configuration Register
4Bh	34Bh	R/W	FFh on nSIO_RESET	SIRQ IRQ11 Configuration Register
4Ch	34Ch	R/W	FFh on nSIO_RESET	SIRQ IRQ12 Configuration Register
4Dh	34Dh	R/W	FFh on nSIO_RESET	SIRQ IRQ13 Configuration Register
4Eh	34Eh	R/W	FFh on nSIO_RESET	SIRQ IRQ14 Configuration Register
4Fh	34Fh	R/W	FFh on nSIO_RESET	SIRQ IRQ15 Configuration Register
50h	350h	R/W	00h on nSIO_RESET	DMA Channel 0, LSB Configuration Register
51h	351h	R/W	00h on nSIO_RESET	DMA Channel 0, MSB Configuration Register
52h	352h	R/W	00h on nSIO_RESET	DMA Channel 1, LSB Configuration Register
53h	353h	R/W	00h on nSIO_RESET	DMA Channel 1, MSB Configuration Register
54h	354h	R/W	00h on nSIO_RESET	DMA Channel 2, LSB Configuration Register
55h	355h	R/W	00h on nSIO_RESET	DMA Channel 2 MSB Configuration Register
56h	356h	R/W	00h on nSIO_RESET	DMA Channel 3, LSB Configuration Register
57h	357h	R/W	00h on nSIO_RESET	DMA Channel 3, MSB Configuration Register
58h	358h	R/W	00h on nSIO_RESET	DMA Channel 4, LSB Configuration Register
59h	359h	R/W	00h on nSIO_RESET	DMA Channel 4 MSB Configuration Register
5Ah	35Ah	R/W	00h on nSIO_RESET	DMA Channel 5, LSB Configuration Register
5Bh	35Bh	R/W	00h on nSIO_RESET	DMA Channel 5, MSB Configuration Register

TABLE 9-10: SCH5636 CONFIGURATION REGISTER MAP (CONTINUED)

LPC CR Index	EC Offset	Type Note 9-6	Reset Note 9-7	Configuration Register Name
5Ch	35Ch	R/W	00h on nSIO_RESET	DMA Channel 6, LSB Configuration Register
5Dh	35Dh	R/W	00h on nSIO_RESET	DMA Channel 6, MSB Configuration Register
5Eh	35Eh	R/W	00h on nSIO_RESET	DMA Channel 7, LSB Configuration Register
5Fh	35Fh	R/W	00h on nSIO_RESET	DMA Channel 7, MSB Configuration Register
60h - 63h	360h	R/W / R	002E_0C01h on nSYS_RST n nSIO_RESET	BAR for Configuration Port
64h - 67h	364h	R/W / R	0000_000Fh on nSIO_RESET	BAR for EMI
68h - 6Bh	368h	R/W / R	0000_0707h on nSIO_RESET	BAR for UART1
6C - 6F	36Ch	R/W / R	0000_0807h on nSIO_RESET	BAR for UART2
70h - 73h	370h	R/W / R	0000_0A3Fh on nSIO_RESET	BAR for Runtime Registers
74h - 7Fh	3374h	R/W / R	0000_0F07h on nSIO_RESET	BAR for Flash SPI
78h - 7Bh	378h	R/W / R	0060_0000h on nSIO_RESET	BAR for 8042
7Ch - 7Fh	37Ch	R/W / R	0000_0000h on nSIO_RESET	BAR for Floppy Disk Controller
80h - 83h	380h	R/W / R	0878_8000h on nSIO_RESET	BAR for Parallel Port
Configuration Registers for LDN 11h **(**Parallel Port**)** **(See** Table 9-17, "Parallel Port, Logical Device 11h"**)**				
30h	330h	R/W	00h	Activate Register
F0h	3F0h	R/W	3Ch	PP Mode Register
F1h	3F1h	R/W	00h	PP Mode Register 2
Configuration Registers for LDN 3Fh **(**Global Configuration**)**				
00h - 02h	300h - 302h	-		Reserved
03H	303	-	-	MCHP Reserved
04h - 06h	304h - 306h	-	00h on nSIO_RESET	Reserved
07h	307h	R/W	-	Logical Device Number
08h - 1Fh	308h- 31Fh	-		Reserved
20h	320h	R	C7h	Device ID
21h	321h	R	Current Revision hardwired	Device Revision A read-only register which provides device revision information
22h– 23h	322h- 323h	-	04h on nSIO_RESET	MCHP Reserved
24h	324h	R/W	00h	Device Mode
25h – 2Fh	325h- 32Fh	-		MCHP Reserved

Note 9-6 R/W / R means that some parts of a register are read/write and some parts are read-only.

Note 9-7 Resets are defined in Section 5.0, "Power, Clocks and Resets": nSYS_RST on page 37 and nSIO_RESET on page 38.

Note 9-8 The Activate register is reset on nSYS_RST if the Power bit in the Configuration Select Register is 0. It is reset on nSIO_RESET if the Power bit in the Configuration Select Register is 1.

9.9 Chip-Level (Global) Control/Configuration Registers

As with all Configuration Registers, the INDEX PORT is used to select a Global Configuration Register in the chip. The DATA PORT is then used to access the selected register.

The Host can access all the Global Configuration registers at the offsets listed in Table 9-11, "Chip-Level (Global) Control/Configuration Registers" through the INDEX PORT and the DATA PORT.

TABLE 9-11: CHIP-LEVEL (GLOBAL) CONTROL/CONFIGURATION REGISTERS

Register	Offset	Description
CHIP (GLOBAL) CONTROL REGISTERS		
Reserved	00h - 06h	Reserved - Writes are ignored, reads return 0.
Logical Device Number	07h	A write to this register selects the current logical device. This allows access to the control and configuration registers for each logical device. **Note:** The Activate command operates only on the selected logical device.
Reserved	08h - 1Fh	Reserved - Writes are ignored, reads return 0.
Device ID	20h	A read-only register which provides device identification: Bits[7:0] = C7h
Device Revision Hard Wired	21h	A read-only register which provides device revision information. Bits[7:0] = current revision when read
Reserved	22h - 23h	Reserved - Writes are ignored, reads return 0.
Device Mode	24h	Bit [1:0] Reserved – writes ignored, reads return “0”. Bit[2] SerIRQ Mode) = 0: Serial IRQ Disabled. = 1: Serial IRQ Enabled (Default). Bit [7:3] Reserved – writes ignored, reads return “0”.
Reserved	25h - 27h	Reserved - Writes are ignored, reads return 0.
Test Register	28h	MCHP Test Mode Register, Reserved for Microchip
Test Register	29h	MCHP Test Mode Register, Reserved for Microchip
Reserved	2Ah - 2Bh	Reserved - Writes are ignored, reads return 0.
Test Register	2Ch	MCHP Test Mode Register, Reserved for Microchip
Test Register	2Dh	MCHP Test Mode Register, Reserved for Microchip
Test Register	2Eh	MCHP Test Mode Register, Reserved for Microchip
Test Register	2Fh	MCHP Test Mode Register, Reserved for Microchip

9.10 Microchip-Defined Logical Device Configuration Registers

Host logical devices not listed have no Microchip-defined configuration registers.

TABLE 9-12: KEYBOARD CONTROLLER, LOGICAL DEVICE 1H

Name	REG Index	Definition
KRST_GA20 Default = 00h on nSIO_RESET Bits[6:5] reset on nSYS_RST only	F0h R/W	KRESET and GateA20 Select Bit[7] Reserved Bit[6] M_ISO. Enables/disables isolation of mouse signals into 8042. Does not affect MDAT signal to mouse wakeup (PME) logic. 1 = block mouse clock and data signals into 8042 0 = do not block mouse clock and data signals into 8042 Bit[5] K_ISO. Enables/disables isolation of keyboard signals into 8042. Does not affect KDAT signal to keyboard wakeup (PME) logic. 1 = block keyboard clock and data signals into 8042 0 = do not block keyboard clock and data signals into 8042 Bit[4] MLATCH = 0 MINT is the 8042 MINT ANDed with Latched MINT (default) = 1 MINT is the latched 8042 MINT Bit[3] KLATCH = 0 KINT is the 8042 KINT ANDed with Latched KINT (default) = 1 KINT is the latched 8042 KINT Bit[2] Port 92 Select = 0 Port 92 Disabled (default) = 1 Port 92 Enabled Bit[1] MCHP Reserved. Must be written with a 0. Bit[0] MCHP Reserved. Must be written with a 0. See Note 9-9
Keyboard Select Default = 00h on on nSIO_RESET	F1h R/W	Bit[0] Kbd/mouse Swap. This bit is used to swap the keyboard and mouse clock and data pins into/out of the 8042 as follows: 1 = internally swap the KCLK pin and the MCLK pin, and the KDAT pin and the MDAT pin into/out of the 8042. 0 = do not swap the keyboard and mouse clock and data pins Bit[1] MCHP Reserved. Must be written with a 0. Bit[7:2] reserved See Note 9-9
8042 Reset Default = 00h on nSIO_RESET	F2h R/W	Bit[0] 8042 Reset. 1 = Put the 8042 into reset 0 = Take the 8042 out of reset Bit[7:1] Reserved

Note 9-9 Wake on Specific Key and general Keyboard and Mouse PME events are unaffected by the M_ISO, K_ISO, KBD/MOUSE SWAP functions of the keyboard logical device configuration registers.

TABLE 9-13: SERIAL PORT 1, LOGICAL DEVICE 7

Name	REG Index	Definition
Serial Port 1 Configuration Select Register Default = 00h on nSYS_RST	F0 (R/W)	Bit[0] CLK SRC = 0 1.8432MHz clock source from 64.52MHz ring oscillator (default) = 1 1.8432MHz clock sourced from 96MHz PLL Bit[1] Power = 0 UART runtime registers controlled by VTR, reset on nSYS_RST (default) = 1 UART runtime registers controlled by VCC, reset on nSIO_RESET Bit[2] Polarity = 0 TX and RX pins are not inverted (default) = 1 TX and RX pins are inverted Bit[7:3] Reserved, set to zero

TABLE 9-14: SERIAL PORT 2, LOGICAL DEVICE 8

Name	REG Index	Definition
Serial Port 2 Configuration Select Register Default = 00h on nSYS_RST	F0 (R/W)	Bit[0] CLK SRC = 0 1.8432MHz clock source from 64.52MHz ring oscillator (default) = 1 1.8432MHz clock sourced from 96MHz PLL Bit[1] Power = 0 UART runtime registers controlled by VTR, reset on nSYS_RST (default) = 1 UART runtime registers controlled by VCC, reset on nSIO_RESET Bit[2] Polarity = 0 TX and RX pins are not inverted (default) = 1 TX and RX pins are inverted Bit[7:3] Reserved, set to zero

TABLE 9-15: RUNTIME REGISTERS, LOGICAL DEVICE AH

Name	REG Index	Definition
SPEKEY Default = 00h on nSIO_RESET	F0h (R/W)	Bit[0] Reserved – returns a '0' when read. Bit[1] SPEKEY_EN. This bit is used to turn the logic for the "wake on specific key" feature on and off. It will disable the clock input to the logic when turned off. The logic will draw no power when disabled. 0 = "Wake on specific key" logic is on (default) 1 = "Wake on specific key" logic is off Bits[7:2] are reserved

TABLE 9-16: FLOPPY DISK CONTROLLER, LOGICAL DEVICE BH

Name	REG Index	Definition
FDD Mode Register Default = 0Eh on nSIO_RESET	F0h (R/W)	Bit[0] Floppy Mode = 0 Normal Floppy Mode (default) = 1 Enhanced Floppy Mode 2 (OS2) Bit[1] FDC DMA Mode = 0 Burst Mode is enabled = 1 Non-Burst Mode (default) Bit[3:2] Interface Mode = 11 AT Mode (default) = 10 (Reserved) = 01 PS/2 = 00 Model 30 Bit[4] Reserved Bit[5] Reserved, set to zero Bit[6] FDC Output Type Control = 0 FDC outputs are OD12 open drain (default) = 1 FDC outputs are O12 push-pull Bit[7] FDC Output Control = 0 FDC outputs active (default) = 1 FDC outputs tri-stated

TABLE 9-16: FLOPPY DISK CONTROLLER, LOGICAL DEVICE BH (CONTINUED)

Name	REG Index	Definition
FDD Option Register Default = 00h on nSIO_RESET	F1h (R/W)	Bit[0] Forced Write Protect = 0 Inactive (default) = 1 FDD WRTPRT# input is forced active when either of the drives has been selected. WRTPRT# (to the FDC Core) = WP (FDC SRA register, bit 1) = (DS0# AND Forced Write Protect) OR WRTPRT# (from the FDD Interface) OR Floppy Write Protect **Notes:** • The Floppy Write Protect bit is in the Device Disable register. • Boot floppy is always drive 0. Bit[1] Reserved Bits[3:2] Density Select = 00 Normal (default) = 01 Normal (reserved for users) = 10 1 (forced to logic “1”) = 11 0 (forced to logic “0”) Bit[7:4] Reserved.
FDD Type Register Default = 0xFF on nSIO_RESET	F2h (R/W)	Bits[1:0] Floppy Drive A Type Bits[3:2] Reserved (could be used to store Floppy Drive B type) Bits[5:4] Reserved (could be used to store Floppy Drive C type) Bits[7:6] Reserved (could be used to store Floppy Drive D type)
	F3h (R)	Reserved, Read as 0 (read only)
FDD0 Default = 00h on nSIO_RESET	F4h (R/W)	Bits[1:0] Drive Type Select: DT1, DT0 Bit[2] Read as 0 (read only) Bit[3] Data Rate Table Select: DRT0 Bit[4] MCHP Reserved. Must be written as 0 Bits[5] Read as 0 (read only) Bits[6] Precompensation Disable PTS = 0 Use Precompensation = 1 No Precompensation Bits[7] Read as 0 (read only)
	F5h (R/W)	MCHP Reserved

TABLE 9-17: PARALLEL PORT, LOGICAL DEVICE 11H

Name	REG Index	Definition
Interrupt Select Default = 0x00 on nSIO_RESET	70h	Bits[7:4] Reserved, set to zero. Bit[3:0] SERIRQ Channel The contents of this register is **only** used as a source for the Parallel Port IRQ field in the cnfgB Extended Parallel Port register. It does not affect the SERIRQ channel on which Parallel Port interrupts appear. If use of the cnfgB register is required, this register should be programmed with the channel number assigned to the Parallel Port in the LPC Logical Device, as shown in Table 9-6, "SIRQ Interrupt Configuration Register Map". If cnfgB is not required, then this register may be left in its default state.
DMA Channel Select Default = 0x00 on nSIO_RESET	74h	Bits[7:3] Reserved, set to zero. Bit[2:0] DMA Channel The contents of this register is **only** used as a source for the Parallel Port DMA field in the cnfgB Extended Parallel Port register. It does not affect the DMA channel on which Parallel Port transfers appear. If use of the cnfgB register is required, this register should be programmed with the channel number assigned to the Parallel Port in the LPC Logical Device, as shown in Table 9-4, "DMA Configuration Register Map". If cnfgB is not required, then this register may be left in its default state.

TABLE 9-17: PARALLEL PORT, LOGICAL DEVICE 11H (CONTINUED)

Name	REG Index	Definition
PP Mode Register Default = 3Ch on nSIO_RESET	F0h (R/W)	Bits[2:0] Parallel Port Mode = 100 Printer Mode (default) = 000 Standard and Bi-directional (SPP) Mode = 001 EPP-1.9 and SPP Mode = 101 EPP-1.7 and SPP Mode = 010 ECP Mode = 011 ECP and EPP-1.9 Mode = 111 ECP and EPP-1.7 Mode Bit[6:3] ECP FIFO Threshold 0111b (default) Bit[7] PP Interrupt Type Not valid when the parallel port is in the Printer Mode (100) or the Standard & Bi-directional Mode (000). = 1 Pulsed Low, released to high-Z. = 0 IRQ follows ACK# when parallel port in EPP Mode or [Printer, SPP, EPP] under ECP. IRQ level type when the parallel port is in ECP, TEST, or Centronics FIFO Mode.
PP Mode Register 2 Default = 00h on nSIO_RESET	F1h (R/W)	Bits[3:0] Reserved. Set to zero Bit [4] TIMEOUT_SELECT = 0 TMOUT (EPP Status Reg.) cleared on write of ‘1’ to TMOUT. = 1 TMOUT cleared on trailing edge of read of EPP Status Reg. Bits[7:5] Reserved. Set to zero.

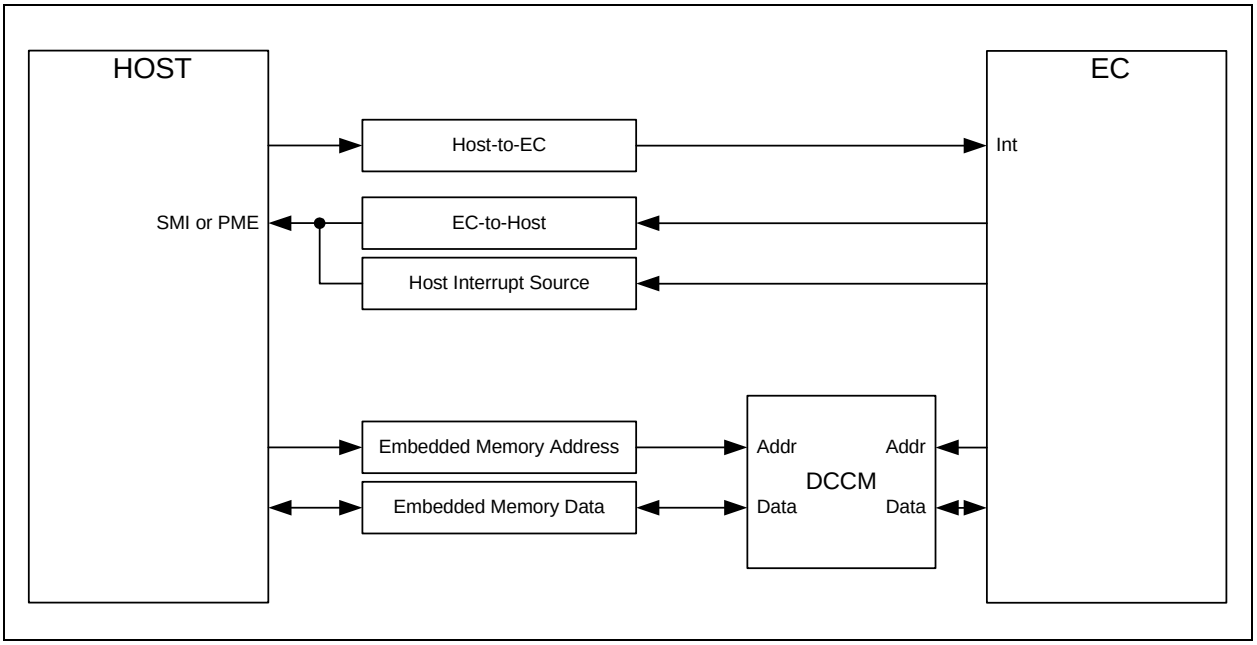
10.0 EMBEDDED MEMORY INTERFACE

10.1 General Description

The Embedded Memory Interface provides a standard run-time mechanism for the host to communicate with the Embedded Controller (EC) and other logical components in the SCH5636 Block Diagram. The Embedded Memory Interface includes 12 byte-addressable registers in the Host's I/O address space, as well as 20 bytes of registers that are accessible only by the EC. The Embedded Memory Interface can be used by the Host to read any byte in a region of EC closely-coupled memory, designated by the EC, without requiring any assistance from the EC. In addition, a portion of the memory can be can be configured so that it can be written by the Host without any EC assistance.

10.1.1 BLOCK DIAGRAM

FIGURE 10-1: EMBEDDED MEMORY INTERFACE BLOCK DIAGRAM



10.2 Reset

This block is reset when nSYS_RST is asserted.

10.3 Interrupts

The Host can generate interrupt events to the EC, as described in Section 10.3.1, "EC Interrupts". In addition, the EC can generate events to the Host, either directly through a SERIRQ, or in conjunction with the PME/SMI logic. The EC to Host interrupts are described in Section 10.3.2, "Embedded Memory Interface SERIRQ Routing" and in Section 10.3.3, "Embedded Memory Interface PME/SMI Routing" and illustrated in FIGURE 19-1: SMI/PME Interrupt Routing on page 199.

10.3.1 EC INTERRUPTS

The Embedded Memory Interface can generate an interrupt event for HOST-to-EC events. See the HOST-to-EC Mailbox Register on page 104. The interrupt for the EMI is routed onto the EM_Int bit in the GIRQ15 Source register.

10.3.2 EMBEDDED MEMORY INTERFACE SERIRQ ROUTING

The EC can use the Embedded Memory Interface to generate SERIRQ events for EC-to-HOST EC events. There are two methods by which the EC generates SERIRQ events to the Host: writes to the EC-to-Host Mailbox Register, and writes to the EC_SWI bits in the Interrupt Source Register.

Writes to the EC-to-Host Mailbox Register generate an interrupt that is routed to the SERIRQ block (see Section 9.6.1, "SERIRQ Configuration Registers," on page 88). For this interrupt, the SELECT bit is set to ‘0’ in the Interrupt Configuration Register. This interrupt signal is an active high level interrupt, so the selected SERIRQ channel will be high when the EM interrupt is asserted.

The EC can also generate an event by setting to 1 any of the EC_SWI bits in the Interrupt Source Register that are enabled by a 1 in the corresponding EC_SWI_EN bits in the Interrupt Mask Register. This event is also asserted by a write to the EC-to-Host Mailbox Register if the EC_WR_EN is 1. The event can be routed to any frame in the SERIRQ stream. For this interrupt, the SELECT bit is set to ‘1’ in the Interrupt Configuration Register. This interrupt signal is active low, so the selected SERIRQ channel will be low when the EM interrupt is asserted. The assertion level matches the level of the IO_SMI# pin, when an SMI is asserted through the Runtime Registers block because of the EMI software interrupt bits.

10.3.3 EMBEDDED MEMORY INTERFACE PME/SMI ROUTING

The two signals from the EMI that are used to generate SERIRQ events can also be used to generate either PME or SMI events.

EM_EVT1 is asserted when the EC writes the EC-to-Host Mailbox Register. EM_EVT1 is an active high level, and is equivalent to the SELECT=0 SERIRQ interrupt. EM_EVT2 is asserted when any of the EC_SWI bits in the Interrupt Source Register is 1 when there is a corresponding 1 bit in the EC_SWI_EN field in the Interrupt Mask Register. EM_EVT2 is also asserted by a write to the EC-to-Host Mailbox Register if the EC_WR_EN is 1. EM_EVT2 is active high level and is therefore the inverse of the SELECT=1 SERIRQ interrupt. EM_EVT1 and EM_EVT2 are routed to the PME and SMI runtime registers, where they can be enabled for event generation. See Section 19.0, "Runtime Registers," on page 198 for details.

10.4 Description

The Embedded Memory Interface contains a Mailbox that enables the Host to send an 8-bit message to the EC and the EC to send an 8-bit message to the Host. When written by the sender, the messages can generate an interrupt at the receiver.

In addition to the messages that can be exchanged, the Embedded Memory Interface permits the Host to read and write a portion of the EC’s Data Closely Coupled Memory (DCCM). Host reads and writes take place without intervention or assistance from the EC.

The Embedded Memory Interface occupies 12 bytes in the Host I/O space. Two bytes constitute the Host-to-EC and EC-to-Host message links. Six bytes are used for the interface into the EC DCCM, two for address and four for data. The four data bytes are used for reads and writes to the EC DCCM using the EC’s Direct Memory Interface (DMI).

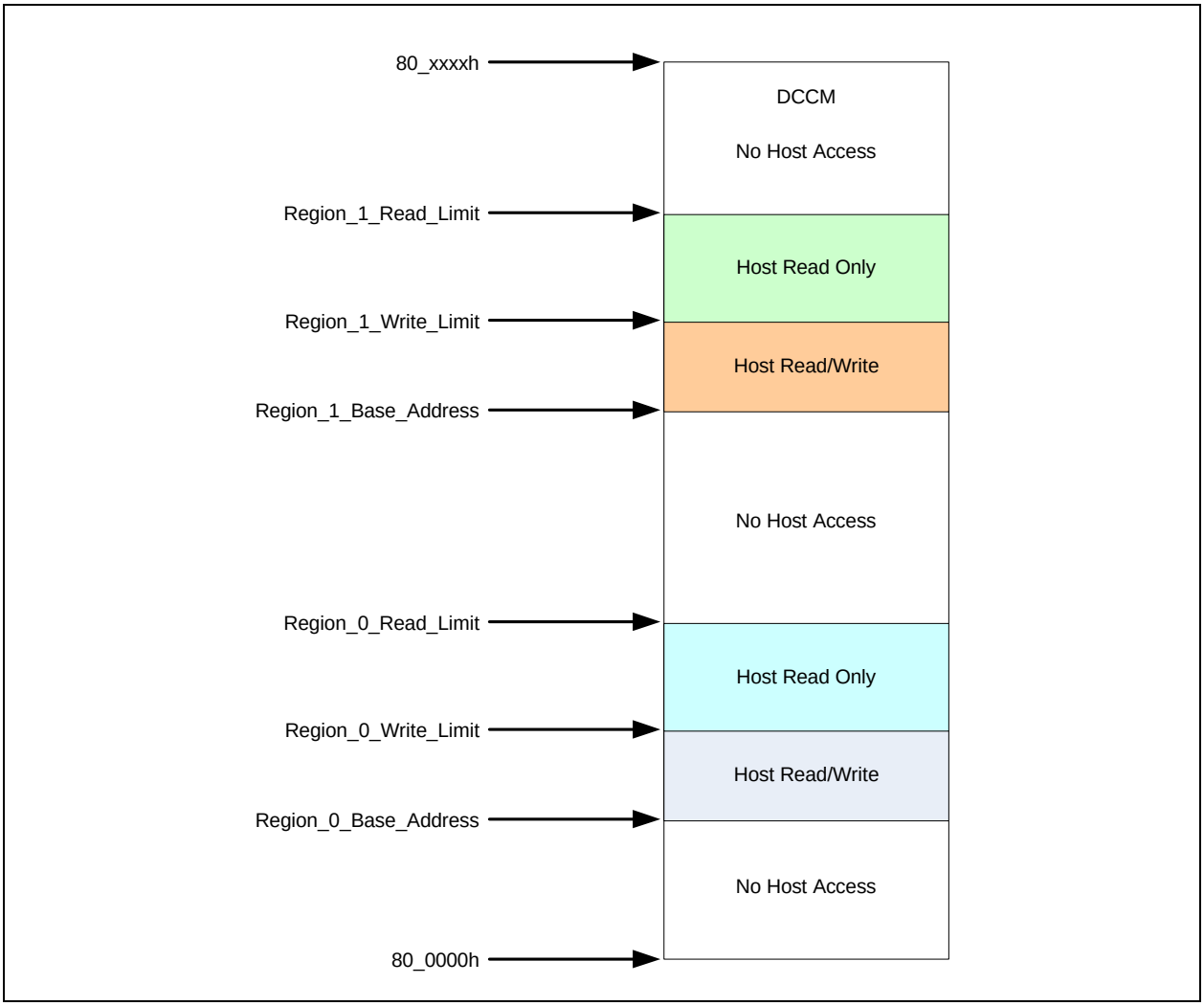
When the Host reads one of the four bytes in the Embedded Memory Interface data register, data from the DCCM at the address defined by the Embedded Memory Interface address register is returned to the Host. Writes to a byte write the corresponding byte in the DCCM. The Embedded Memory Interface can be configured so that, although Host I/O is always byte at a time, transfers between the Embedded Memory Interface data bytes and the DCCM can configured to occur as single bytes, 2-byte blocks or 4-byte blocks. This is done so that data that the EC treats as 16-bit or 32-bit will be consistent in the Host, even though one byte of the DCCM data may change between two or more 8-bit accesses by the Host.

In addition, there is an auto-increment function for the Embedded Memory Interface address register. When enabled, the Host can read or write blocks of memory in the DCCM by repeatedly accessing the Embedded Memory Interface data register, without requiring Host updates to the Embedded Memory Interface address register.

10.4.1 EMBEDDED MEMORY MAP

Each Embedded Memory interface provides direct access for the Host into two windows in the EC DCCM SRAM. This mapping is shown in Figure 10-2, "Embedded Memory Addressing":

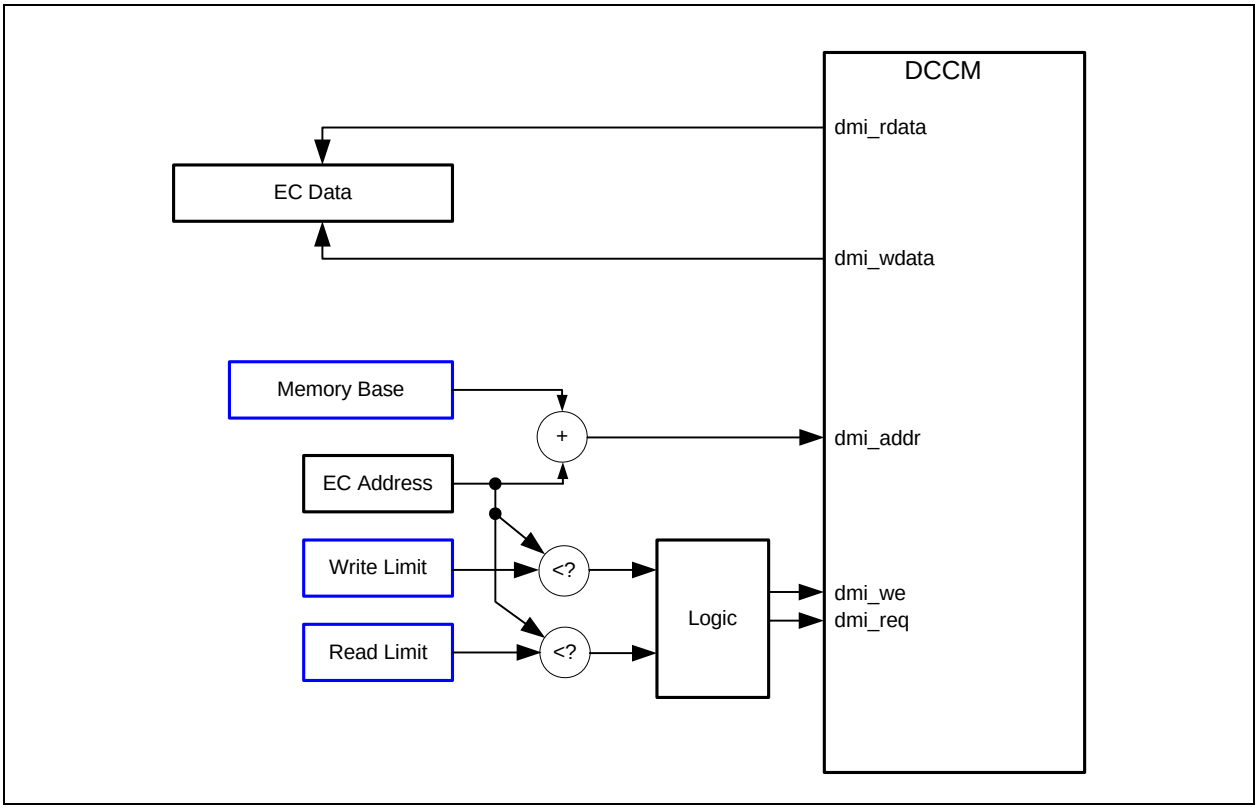
FIGURE 10-2: EMBEDDED MEMORY ADDRESSING



The Base addresses, the Read limits and the Write limits are defined by registers that are in the EC address space and cannot be accessed by the Host. In each region, the Read limit need not be greater than the Write limit. The regions can be contiguous or overlapping.For example, if the Region 0 Read limit is set to 0 and the Write limit is set to a positive number, then the Embedded Memory interface defines a region in the EC memory that the EC can read and write but is write-only for the host. This might be useful for storage of security data, which the Host might wish to send to the EC but should not be readable in the event a virus invades the Host.

Each window into the EC memory can be as large as the DCCM. The Embedded Memory Interface uses the EC's DCCM Direct Memory Interface (DMI) in order to access the memory. Figure 10-3, "Embedded Memory Region Address Control" shows the relationship between one off the regions in the Embedded Memory Interface and the DCCM DMI:

FIGURE 10-3: EMBEDDED MEMORY REGION ADDRESS CONTROL



10.4.2 EMBEDDED MEMORY INTERFACE USAGE

The Embedded Memory Interface provides a generic facility for communication between the Host and the EC and can be used for many functions. Some examples are:

- Virtual registers. A block of read-only memory locations in the DCCM can be used to implement a set of virtual registers. The EC can update these locations with that the Host can later read.
- Program downloading. Because the Instruction Closely Coupled Memory is implemented in the same SRAM as the DCCM, the Embedded Memory Interface can be used by the Host to download new program segments for the EC. The Read/Write window would be configured by the Host to point to the beginning of the loadable program region, which could then be loaded by the Host.
- Data exchange. The Read/Write portion of the memory window can be used to contain a communication packet. The Host, by default, "owns" the packet, and can write it at any time. When the Host wishes to communicate with the EC, it sends the EC a command, through the Host-to-EC message facility, to read the packet and perform some operations as a result. When it is completed processing the packet, the EC can inform the Host, either through a message in the EC-to-Host channel or by triggering an event such as an SMI directly. If return results are required, the EC can write the results into the Read/Write region, which the Host can read directly when it is informed that the EC has completed processing. Depending on the command, the operations could entail update of virtual registers in the DCCM, reads of any register in the EC address space, or writes of any register in the EC address space. Because there are two regions that are defined by the base registers, the memory used for the communication packet does not have to be contiguous with a set of virtual registers.

Because there are two Embedded Memory Interface memory regions, the Embedded Memory Interface cannot be used for more than two of these functions at a time. The Host can request that the EC switch from one function to another through the use of the Host-to-EC mailbox register.

The Application ID Register is provided to help software applications track ownership of an Embedded Memory Interface. An application can write the Application ID Register with its Application ID, then immediately read it back. If the read value is not the same as the value written, then another application has ownership of the interface.

Note: The protocol used to pass commands back and forth through the Embedded Memory Interface Registers Interface is left to the System designer. Microchip can provide an application example of working code in which the host uses the Embedded Memory Interface registers to gain access to all of the EC registers.

10.5 Registers

The Table 10-1 is a register summary for the Embedded Memory Interface. The LPC I/O address for each Run-Time Register is described below as an offset from its Base Address Register.

TABLE 10-1: EMBEDDED MEMORY INTERFACE REGISTER SUMMARY

Register Name	I/O Offset	Size	Type	Notes
RUNTIME REGISTERS				
HOST-to-EC Mailbox Register	00h	8	R/W	Note 10-1
EC-to-Host Mailbox Register	01h	8	R/WC	Note 10-2
EC Address Register	02h 03h	8 8	R/W	
EC Data Register	04h 05h 06h 07h	8 8 8 8	R/W	
Interrupt Source Register	08h 09h	8 8	R/WC	
Interrupt Mask Register	0Ah 0Bh	8 8	R/W	
Application ID Register	0Ch	8	R/W	
Interrupt Set Register	114h	19	R/W	
Host Clear Enable Register	116h	16	R/W	

Note 10-1 Interrupt is cleared when read by the EC.

Note 10-2 Interrupt is cleared when read by the host.

10.5.1 EMBEDDED MEMORY INTERFACE CONTROL REGISTERS

Mailbox Register, HOST-to-EC, and Mailbox Register, EC-to-HOST, are specifically designed to pass commands between the host and the EC. If enabled, these registers can generate interrupts.

When the host performs a write of the HOST-to-EC mailbox register, an interrupt will be generated and seen by the EC if unmasked.

When the EC writes the EC-to-HOST mailbox register, an SIRQ event or an event such as SMI or PME may be generated and seen by the host if unmasked.

10.6 Registers

10.6.1 HOST-TO-EC MAILBOX REGISTER

TABLE 10-2: HOST-TO-EC MAILBOX REGISTER

HOST OFFSET	00h					8-Bit	**HOST SIZE**	
POWER	VTR					00h	nSYS_RST **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	HOST_EC_MBOX[7:0]							

HOST_EC_MBOX

If enabled, an interrupt to the EC marked by the EM bit in the GIRQ15 Source register will be generated whenever the Host writes this register. The Host and the EC can read and write this register at offset 000h.

Writes of a 1 to any bit in this register by the EC to this register will cause the bit to be cleared. Writes of a 0 to any bit have no effect.

10.6.2 EC-TO-HOST MAILBOX REGISTER

TABLE 10-3: EC-TO-HOST MAILBOX REGISTER

HOST OFFSET	01h					8-Bit	**HOST SIZE**	
EC OFFSET	01h					8-Bit	**EC SIZE**	
POWER	VTR					00h	nSYS_RST **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	EC_HOST_MBOX[7:0]							

EC_HOST_MBOX

An EC write to this register at offset 101h will set bit EC_WR in the Interrupt Source Register to ‘1b’. The EC_WR bit is routed to the Runtime Registers as the EM_EVT1 bit. The EC can also read this register.

Writes of a 1 to any bit in this register at offset 01h, by the Host or by the EC, will cause the bit to be cleared. Writes of a 0 to any bit have no effect.

10.6.3 EC ADDRESS REGISTER

TABLE 10-4: EC ADDRESS REGISTER

HOST OFFSET	Byte 0: 02h Byte 1: 03h					8-bit	**HOST SIZE**	
EC OFFSET	02h					16-bit	**EC SIZE**	
POWER	VTR					0000h	nSYS_RST **DEFAULT**	
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Region	EC_Address[14:8]						
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	EC_Address[7:2]						Access_Type	

Access_Type

This field defines the type of access that occurs when the EC Data Register is read or written.

00: 8-bit access. Any byte read of Byte 0 through Byte 3 in the EC Data Register causes the corresponding byte within the 32-bit double word addressed by EC_Address to be loaded into the byte of EC Data Register and returned by the read. Any byte write to Byte 0 through Byte 3 in the EC Data Register writes the corresponding byte within the 32-bit double word addressed by EC_Address, as well as the byte of the EC Data Register.

01: 16-bit access. A read of Byte 0 in the EC Data Register causes the 16 bits in the DCCM at an offset of EC_Address to be loaded into Byte 0 and Byte 1 of the EC Data Register. The read then returns the contents of Byte 0. A read of Byte 2 in the EC Data Register causes the 16 bits in the DCCM at an offset of EC_Address+2 to be loaded into Byte 2 and Byte 3 of the EC Data Register. The read then returns the contents of Byte 2. A read of Byte 1 or Byte 3 in the EC Data Register return the contents of the register, without any update from the DCCM.

A write of Byte 1 in the EC Data Register causes Bytes 1 and 0 of the EC Data Register to be written into the 16 bits in the DCCM at an offset of EC_Address. A write of Byte 3 in the EC Data Register causes Bytes 3 and 2 of the EC Data Register to be written into the 16 bits in the DCCM at an offset of EC_Address+2. A write of Byte 0 or Byte 2 in the EC Data Register updates the contents of the register, without any change to the DCCM.

10: 32-bit access. A read of Byte 0 in the EC Data Register causes the 32 bits in the DCCM at an offset of EC_Address to be loaded into the entire EC Data Register. The read then returns the contents of Byte 0. A read of Byte 1, Byte 2 or Byte 3 in the EC Data Register returns the contents of the register, without any update from the DCCM.

A write of Byte 3 in the EC Data Register causes the EC Data Register to be written into the 32 bits in the DCCM at an offset of EC_Address. A write of Byte 0, Byte 1 or Byte 2 in the EC Data Register updates the contents of the register, without any change to the DCCM.

11: Auto-increment 32-bit access. This defines a 32-bit access, as in the 10 case. In addition, any read or write of Byte 3 in the EC Data Register causes the EC Address Register to be incremented by 1. That is, the EC_Address field will point to the next 32-bit double word in the DCCM.

EC_Address[14:2]

This field defines the location in memory that can be read and/or written with the EC Data Register. The address is an offset from the base of the Host-accessible region in the EC DCCM SRAM. The base of the Host-accessible region.

Region

The field specifies which of two segments in the on-chip SRAM is to be used in conjunction with EC_Address[14:2] to generate accesses to the memory.

10.6.4 EC DATA REGISTER

TABLE 10-5: EC DATA REGISTER

HOST OFFSET	Byte 0: 04h Byte 1: 05h Byte 2: 06h Byte 3: 07h					8-bit	**HOST SIZE**	
EC OFFSET	04h					32-bit	**EC SIZE**	
POWER	VTR					0000_0000h	**VTR POR DEFAULT**	
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data3[7:0]							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data2[7:0]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data1[7:0]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data0[7:0]							

DATA

This is a 32-bit register which returns data to the Host from the EC DCCM at the address specified by EC_Address[14:2]. The description of bits Access_Type in the EC Address Register defines which reads and writes from the Host trigger transfers of data between this register and the DCCM.

A write to the EC Data Register when the EC Address Register is in a read-only or a no-access region, as defined by the Memory Base and Limit registers, will update the EC Data Register but memory will not be modified. A read to the EC Data Register when the EC Address Register is in a no-access region, as defined by the Memory Base and Limit registers, will not trigger a memory read and will not modify the EC Data Register. In auto-increment mode (Access_Type=11b), reads of Byte 3 of the EC Data Register will still trigger increments of the EC Address Register when the address is out of bounds, while writes of Byte 3 will not.

10.6.5 INTERRUPT SOURCE REGISTER

TABLE 10-6: INTERRUPT SOURCE REGISTER

HOST OFFSET	Byte 0: 08h Byte 1: 09h					8-Bit	**HOST SIZE**	
EC OFFSET	08h					16-Bit	**EC SIZE**	
POWER	VTR					0000h	nSYS_RST **DEFAULT**	
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	EC_SWI[14:7]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R
BIT NAME	EC_SWI[6:0]							EC_WR

EC_WR

This bit is set autonomously when the EC-to-Host Mailbox Register has been written by the EC at offset 101h. This bit appears as the signal EM_EVT1 in the Runtime Registers and can be used to enable a PME or SMI. In addition, if this bit is 1 and bit EC_WR_EN in the Interrupt Mask Register is 1, the signal EM_EVT2 will be asserted in the Runtime Register block.

This bit is automatically cleared by a read of the EC-to-Host Mailbox Register at offset 01h.

EC_SWI

Each bit in this field is cleared when written with a ‘1b’. The ability to clear the bit can be disabled by the EC.

The signal EM_EVT2 will be asserted in the Runtime Register block if any bit in this field is 1 and the corresponding bit in the EC_SWI_EN field in the Interrupt Mask Register is also 1.

10.6.6 INTERRUPT MASK REGISTER

TABLE 10-7: INTERRUPT MASK REGISTER

HOST OFFSET	Byte 0: 0Ah Byte 1: 0Bh					8-Bit	**HOST SIZE**	
EC OFFSET	0Ah					16-Bit	**EC SIZE**	
POWER	VTR					0000h	nSYS_RST **DEFAULT**	
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	EC_SWI_EN[14:7]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	EC_SWI_EN[6:0]							EC_WR_EN

EC_WR_EN

If this bit is ‘1b’, the interrupt generated by bit EC_WR in the Interrupt Source Register is enabled.

EC_SWI_EN

Each bit that is set to ‘1b’ in this field enables the generation of and interrupt by the corresponding bit in the EC_SWI field in the Interrupt Source Register.

10.6.7 APPLICATION ID REGISTER

TABLE 10-8: APPLICATION ID REGISTER

HOST OFFSET	0Ch					8-Bit	**HOST SIZE**	
EC OFFSET	0Ch					8-Bit	**EC SIZE**	
POWER	VTR					00h	nSYS_RST **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Application_ID[7:0]							

Application_ID

When this field is 00h it can be written with any value. When set to a non-zero value, writing that value will clear this register to 00h. When set to a non-zero value, writing any value other than the current contents will have no effect.

10.6.8 INTERRUPT SET REGISTER

This register is accessible to the EC only.

TABLE 10-9: INTERRUPT SET REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	114h					16-Bit	**EC SIZE**	
POWER	VTR					0000h	nSYS_RST **DEFAULT**	
BUS	Host AHB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS
BIT NAME	EC_SWI_Set[14:7]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R
BIT NAME	EC_SWI_Set[6:0]							Reserved

EC_SWI_Set[14:0]

This register provides the EC with a means of updating the Interrupt Source Register. Writing a bit in this field with a ‘1b’ sets the corresponding bit in the Interrupt Source Register to ‘1b’. Writing a bit in this field with a ‘0b’ has no effect. Reading this field returns the current contents of the Interrupt Source Register.

10.6.9 HOST CLEAR ENABLE REGISTER

This register is accessible to the EC only.

TABLE 10-10: HOST CLEAR ENABLE REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	116h					16-Bit	**EC SIZE**	
POWER	VTR					0000h	nSYS_RST **DEFAULT**	
BUS	Host AHB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Host_Clr_Enable[14:7]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
BIT NAME	Host_Clr_Enable[6:0]							Reserved

Host_Clr_Enable[14:0]

When a bit in this field is ‘0b’, the corresponding bit in the Interrupt Source Register cannot be cleared by writes to the Interrupt Source Register. When a bit in this field is ‘1b’, the corresponding bit in the Interrupt Source Register can be cleared when that register bit is written with a ‘1b’.

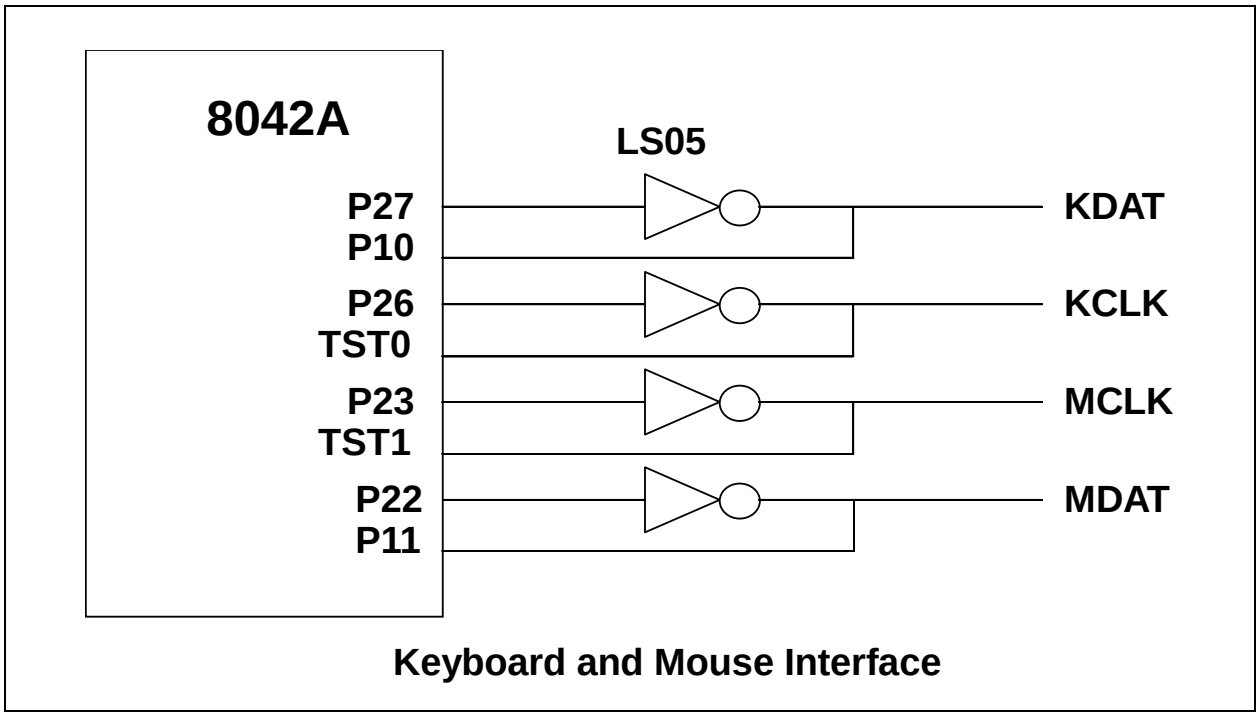
These bits allow the EC to control whether the status bits in the Interrupt Source Register are based on an edge or level event

11.0 KEYBOARD CONTROLLER

11.1 General Description

The SCH5636 is a Super I/O and Universal Keyboard Controller that is designed for intelligent keyboard management in desktop computer applications. The Universal Keyboard Controller uses an 8042 microcontroller CPU core. This section concentrates on the SCH5636 enhancements to the 8042. For general information about the 8042, refer to the "Hardware Description of the 8042" in the *8-Bit Embedded Controller Handbook*.

FIGURE 11-1: SCH5636 KEYBOARD AND MOUSE INTERFACE



KIRQ is the Keyboard IRQ.

MIRQ is the Mouse IRQ.

Port 21 is used to create a GATEA20 signal from the SCH5636.

11.2 Power, Clocks and Reset

11.2.1 POWER DOMAIN

This block is powered by the VTR power supply.

11.2.2 CLOCKS

This block uses a 12MHz clock derived from the 96MHz PLL. as well as the Host Bus Clock when the Configuration registers are accessed.

11.2.3 RESET

This block is reset when nSIO_RESET is asserted.

11.3 Interrupts

The Keyboard Interface can generate two Serial Interrupt Requests on the LPC bus. Each can be put on any of the SERIRQ channels as configured by the SERIRQ configuration registers in the LPC host interface block. A Keyboard interrupt is the Primary interrupt (chosen when the SELECT field in the LPC SERIRQ configuration register is 0) and a mouse interrupt is the secondary interrupt (chosen when the SELECT field is 1).

The Keyboard interrupt and Mouse interrupt signals are also routed to GIRQ15, as the KBC_KIRQ and KBC_MIRQ bits, so the EC may also respond to keyboard controller interrupts.

11.4 Keyboard Interface

The SCH5636 LPC interface is functionally compatible with the 8042 style host interface. It consists of the D0-7 data signals; the read and write signals and the Status register, Input Data register, and Output Data register. Table 11-1 shows how the interface decodes the control signals. In addition to the above signals, the host interface includes keyboard and mouse IRQs.

TABLE 11-1: I/O ADDRESS MAP

Address	Command	Block	Function (Note:)
60h	Write	KDATA	Keyboard Data Write (C/D=0)
	Read	KDATA	Keyboard Data Read
64h	Write	KDCTL	Keyboard Command Write (C/D=1)
	Read	KDCTL	Keyboard Status Read

Note: These registers consist of three separate 8-bit registers. Status, Data/Command Write and Data Read.

Keyboard Data Write

bit This is an 8 bit write only register. When written, the C/D status bit of the status register is cleared to zero and the IBF bit is set.

Keyboard Data Read

This is an 8 bit read only register. If enabled by “ENABLE FLAGS”, when read, the KIRQ output is cleared and the OBF flag in the status register is cleared. If not enabled, the KIRQ and/or AUXOBF1 must be cleared in software.

Keyboard Command Write

This is an 8 bit write only register. When written, the C/D status bit of the status register is set to one and the IBF bit is set.

Keyboard Status Read

This is an 8 bit read only register. Refer to the description of the Status Register for more information.

CPU-to-Host Communication

The SCH5636 CPU can write to the Output Data register via register DBB. A write to this register automatically sets Bit 0 (OBF) in the Status register. See Table 11-2.

TABLE 11-2: HOST INTERFACE FLAGS

8042 INSTRUCTION	FLAG
OUT DBB	Set OBF, and, if enabled, the KIRQ output signal goes high

Host-to-CPU Communication

The host system can send both commands and data to the Input Data register. The CPU differentiates between commands and data by reading the value of Bit 3 of the Status register. When bit 3 is “1”, the CPU interprets the register contents as a command. When bit 3 is “0”, the CPU interprets the register contents as data. During a host write operation, bit 3 is set to “1” if SA2 = 1 or reset to “0” if SA2 = 0.

KIRQ

If “EN FLAGS” has been executed and P24 is set to a one: the OBF flag is gated onto KIRQ. The KIRQ signal can be connected to system interrupt to signify that the SCH5636 CPU has written to the output data register via “OUT DBB,A”. If P24 is set to a zero, KIRQ is forced low. On power-up, after a valid RST pulse has been delivered to the device, KIRQ is reset to 0. KIRQ will normally reflects the status of writes “DBB”. (KIRQ is normally selected as IRQ1 for keyboard support.)

If “EN FLAGS” has not been executed: KIRQ can be controlled by writing to P24. Writing a zero to P24 forces KIRQ low; a high forces KIRQ high.

MIRQ

If “EN FLAGS” has been executed and P25 is set to a one:; IBF is inverted and gated onto MIRQ. The MIRQ signal can be connected to system interrupt to signify that the SCH5636 CPU has read the DBB register. If “EN FLAGS” has not been executed, MIRQ is controlled by P25, Writing a zero to P25 forces MIRQ low, a high forces MIRQ high. (MIRQ is normally selected as IRQ12 for mouse support).

Gate A20

A general purpose P21 is used as a software controlled Gate A20 or user defined output.

11.5 External Keyboard and Mouse Interface

bit Industry-standard PC-AT-compatible keyboards employ a two-wire, bidirectional TTL interface for data transmission. Several sources also supply PS/2 mouse products that employ the same type of interface. To facilitate system expansion, the SCH5636 provides four signal pins that may be used to implement this interface directly for an external keyboard and mouse.
bit The SCH5636 has four high-drive, open-drain output, bidirectional port pins that can be used for external serial interfaces, such as external keyboard and PS/2-type mouse interfaces. They are KCLK, KDAT, MCLK, and MDAT. P26 is inverted and output as KCLK. The KCLK pin is connected to TEST0. P27 is inverted and output as KDAT. The KDAT pin is connected to P10. P23 is inverted and output as MCLK. The MCLK pin is connected to TEST1. P22 is inverted and output as MDAT. The MDAT pin is connected to P11.

Note:	External pull-ups may be required.

11.5.1 KEYBOARD/MOUSE SWAP BIT

There is a Kbd/mouse Swap bit in the Keyboard Select configuration register located at F1h in Logical Device 1. This bit can be used to swap the keyboard and mouse clock and data pins into/out of the 8042. The default value of this bit is ‘0’ on VCC RESET, VTR POR and PCI Reset. This bit is defined as follows:

1 = internally swap the KCLK pin and the MCLK pin, and the KDAT pin and the MDAT pin into/out of the 8042.

0 = do not swap the keyboard and mouse clock and data pins

11.5.2 KEYBOARD POWER MANAGEMENT

The keyboard provides support for two power-saving modes: soft power-down mode and hard power-down mode. In soft power-down mode, the clock to the ALU is stopped but the timer/counter and interrupts are still active. In hard power down mode the clock to the 8042 is stopped.

Soft Power-Down Mode

This mode is entered by executing a HALT instruction. The execution of program code is halted until either RESET is driven active or a data byte is written to the DBBIN register by a master CPU. If this mode is exited using the interrupt, and the IBF interrupt is enabled, then program execution resumes with a CALL to the interrupt routine, otherwise the next instruction is executed. If it is exited using RESET then a normal reset sequence is initiated and program execution starts from program memory location 0.

Hard Power-Down Mode

This mode is entered by executing a STOP instruction. The oscillator is stopped by disabling the oscillator driver cell. When either RESET is driven active or a data byte is written to the DBBIN register by a master CPU, this mode will be exited (as above). However, as the oscillator cell will require an initialization time, either RESET must be held active for sufficient time to allow the oscillator to stabilize. Program execution will resume as above.

11.5.3 INTERRUPTS

The SCH5636 provides the two 8042 interrupts: IBF and the Timer/Counter Overflow.

11.5.4 MEMORY CONFIGURATIONS

The SCH5636 provides 2K of on-chip ROM and 256 bytes of on-chip RAM.

11.5.5 REGISTER DEFINITIONS

Host I/F Data Register

The Input Data register and Output Data register are each 8 bits wide. A write to this 8 bit register will load the Keyboard Data Read Buffer, set the OBF flag and set the KIRQ output if enabled. A read of this register will read the data from the Keyboard Data or Command Write Buffer and clear the IBF flag. Refer to the KIRQ and Status register descriptions for more information.

Host I/F Status Register

The Status register is 8 bits wide.

Table 11-3 shows the contents of the Status register.

TABLE 11-3: STATUS REGISTER

D7	D6	D5	D4	D3	D2	D1	D0
UD	UD	UD	UD	C/D	UD	IBF	OBF

Status Register

This register is cleared on a reset. This register is read-only for the Host and read/write by the SCH5636 CPU.

UD Writable by SCH5636 CPU. These bits are user-definable.

C/D (Command Data)-This bit specifies whether the input data register contains data or a command (0 = data, 1 = command). During a host data/command write operation, this bit is set to “1” if SA2 = 1 or reset to “0” if SA2 = 0.

IBF (Input Buffer Full)- This flag is set to 1 whenever the host system writes data into the input data register. Setting this flag activates the SCH5636 CPU’s nIBF (MIRQ) interrupt if enabled. When the SCH5636 CPU reads the input data register (DBB), this bit is automatically reset and the interrupt is cleared. There is no output pin associated with this internal signal.

OBF (Output Buffer Full) - This flag is set to whenever the SCH5636 CPU write to the output data register (DBB). When the host system reads the output data register, this bit is automatically reset.

11.5.6EXTERNAL CLOCK SIGNAL

The SCH5636 Keyboard Controller clock source is a 12 MHz clock generated from the internal 96MHz PLL. The reset pulse must last for at least 24 16 MHz clock periods. The pulse-width requirement applies to both internally (VCC RESET) and externally generated reset signals. In power-down mode, the external clock signal is not loaded by the chip.

11.5.7 DEFAULT RESET CONDITIONS

The SCH5636 has one source of hardware reset for the keyboard controller: an external reset via the LRESET# pin. Refer to Table 11-4 for the effect of each type of reset on the internal registers.

TABLE 11-4: RESETS

Description	Hardware Reset (LRESET#)
KCLK	Low
KDAT	Low
MCLK	Low
MDAT	Low
Host I/F Data Reg	N/A
Host I/F Status Reg	00H
Note: N/A = Not Applicable	

11.5.7.1 GATEA20 and Keyboard Reset

The SCH5636 provides two options for GateA20 and Keyboard Reset: 8042 Software Generated GateA20 and KRESET and Port 92 Fast GateA20 and KRESET.

11.5.7.2 Port 92 Fast GATEA20 and Keyboard Reset

Port 92 Register

This port can only be read or written if Port 92 has been enabled via bit 2 of the KRST_GA20 Register (Logical Device 1, F0h) set to 1.

This register is used to support the alternate reset (nALT_RST) and alternate A20 (ALT_A20) functions.

Name	Port 92
Location	92h
Default Value	24h
Attribute	Read/Write
Size	8 bits

TABLE 11-5: PORT 92 REGISTER

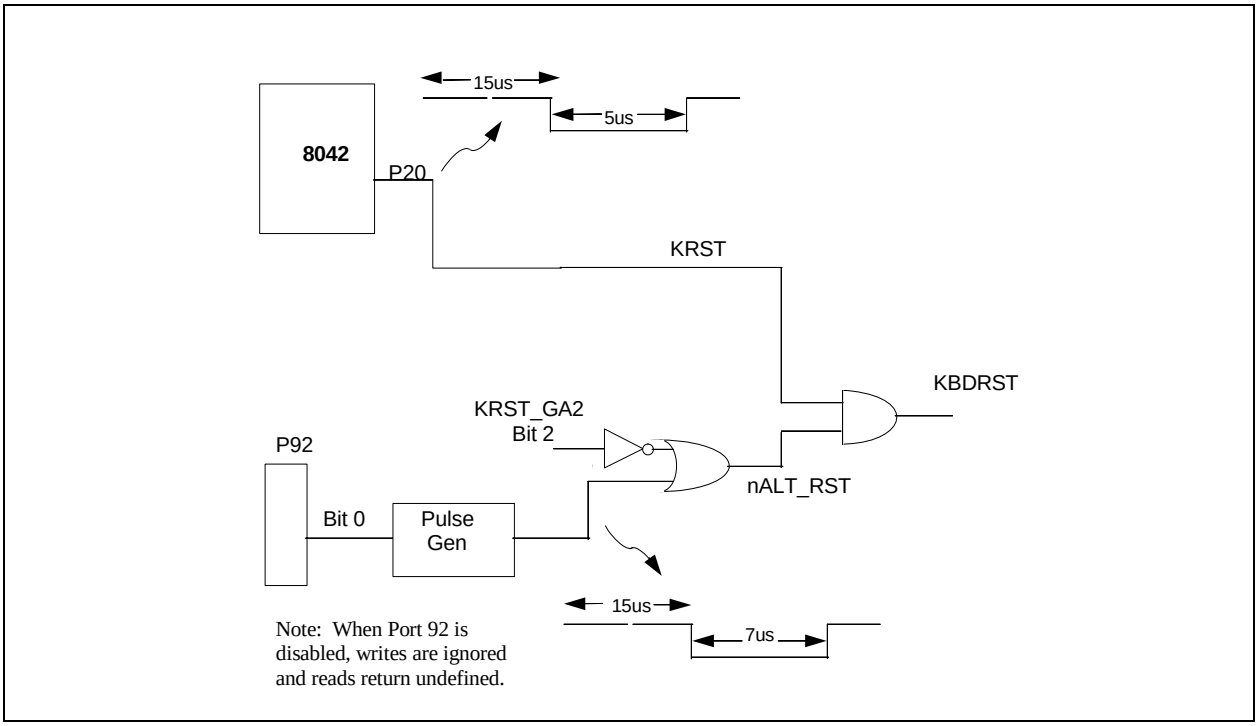
Bit	Function
7:6	Reserved. Returns 00 when read
5	Reserved. Returns a 1 when read
4	Reserved. Returns a 0 when read
3	Reserved. Returns a 0 when read
2	Reserved. Returns a 1 when read
1	ALT_A20 Signal control. Writing a 0 to this bit causes the ALT_A20 signal to be driven low. Writing a 1 to this bit causes the ALT_A20 signal to be driven high.
0	Alternate System Reset. This read/write bit provides an alternate system reset function. This function provides an alternate means to reset the system CPU to effect a mode switch from Protected Virtual Address Mode to the Real Address Mode. This provides a faster means of reset than is provided by the Keyboard controller. This bit is set to a 0 by a system reset. Writing a 1 to this bit will cause the nALT_RST signal to pulse active (low) for a minimum of 1 µs after a delay of 500 ns. Before another nALT_RST pulse can be generated, this bit must be written back to a 0.

NGATEA20		
8042 P21	**ALT_A20**	**System NA20M**
0	0	0
0	1	1
1	0	1
1	1	1

Bit 0 of Port 92, which generates the nALT_RST signal, is used to reset the CPU under program control. This signal is AND'ed together externally with the reset signal (KBDRST#) from the keyboard controller to provide a software means of resetting the CPU. This provides a faster means of reset than is provided by the keyboard controller. Writing a 1 to bit 0 in the Port 92 Register causes this signal to pulse low for a minimum of 7µs, after a delay of a minimum of 15µs. Before another nALT_RST pulse can be generated, bit 0 must be set to 0 either by a system reset of a write to Port 92. Upon reset, this signal is driven inactive high (bit 0 in the Port 92 Register is set to 0).

If Port 92 is enabled, i.e., bit 2 of KRST_GA20 is set to 1, then a pulse is generated by writing a 1 to bit 0 of the Port 92 Register and this pulse is AND'ed with the pulse generated from the 8042. This pulse is output on pin KRESET and its polarity is controlled by the GPI/O polarity configuration.

FIGURE 11-2: KBDRST IMPLEMENTATION



Bit 1 of Port 92, the ALT_A20 signal, is used to force nA20M to the CPU low for support of real mode compatible software. This signal is externally OR'ed with the A20GATE signal from the keyboard controller and CPURST to control the nA20M input of the CPU. Writing a 0 to bit 1 of the Port 92 Register forces ALT_A20 low. ALT_A20 low drives nA20M to the CPU low, if A20GATE from the keyboard controller is also low. Writing a 1 to bit 1 of the Port 92 Register forces ALT_A20 high. ALT_A20 high drives nA20M to the CPU high, regardless of the state of A20GATE from the keyboard controller. Upon reset, this signal is driven low.

Latches On Keyboard and Mouse IRQs

The implementation of the latches on the keyboard and mouse interrupts are shown in Figure 11-3 and Figure 11-4.

FIGURE 11-3: KEYBOARD LATCH

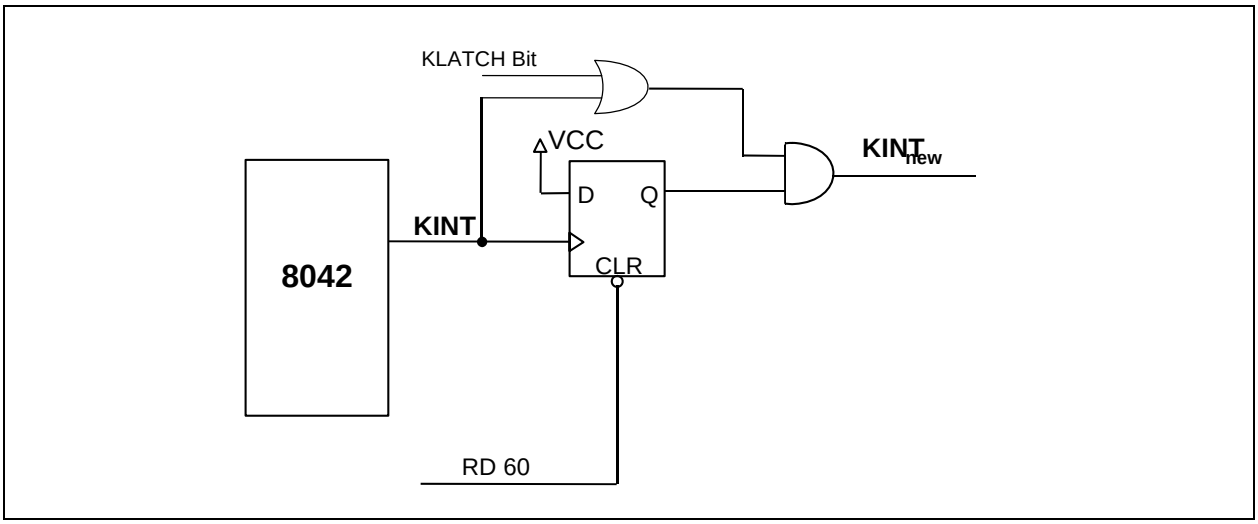
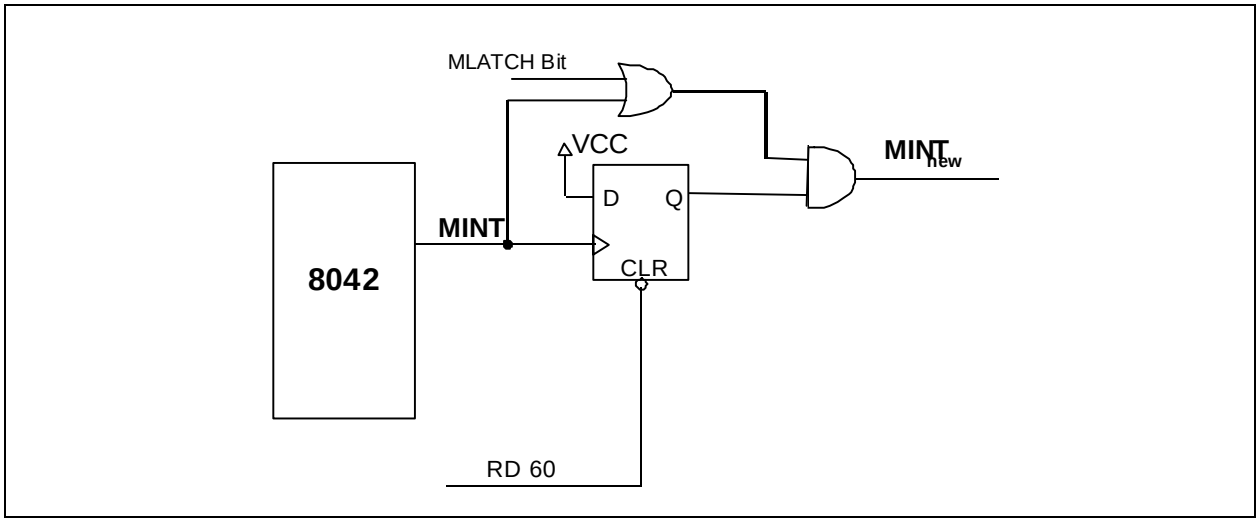


FIGURE 11-4: MOUSE LATCH



The KLATCH and MLATCH bits are located in the KRST_GA20 register, in Logical Device 1 at F0h.

These bits are defined as follows:

Bit[4]: MLATCH – Mouse Interrupt latch control bit. 0=MINT is the 8042 MINT ANDed with Latched MINT (default), 1=MINT is the latched 8042 MINT.

Bit[3]: KLATCH – Keyboard Interrupt latch control bit. 0=KINT is the 8042 KINT ANDed with Latched KINT (default), 1=KINT is the latched 8042 KINT.

See Section 11.7, "Detailed Description of Configuration Registers," on page 118 for a description of this register.

11.5.8 KEYBOARD AND MOUSE PME GENERATION

The SCH5636 sets the associated PME Status bits when the following conditions occur:

- Active (falling) Edge on Keyboard Data Signal (KDAT)
- Active (falling) Edge on Mouse Data Signal (MDAT)

These events can cause a PME to be generated if the associated PME Wake Enable register bit and the global PME_EN bit are set. Refer Section 19.0, "Runtime Registers," on page 198 for details on the PME Status and Enable registers.

Both the keyboard interrupt and mouse interrupt PMEs can be generated when the part is powered by VCC. The keyboard data and mouse data PMEs can be generated both when the part is powered by VCC, and VTR (VCC=0).

When using the keyboard and mouse data signals for wakeup, it may be necessary to isolate the keyboard signals (KCLK, KDAT, MCLK, MDAT) from the 8042 prior to entering certain system sleep states. This is due to the fact that the normal operation of the 8042 can prevent the system from entering a sleep state or trigger false PME events. The SCH5636 has “isolation” bits for the keyboard and mouse signals, which allow the keyboard and mouse data signals to go into the wakeup logic but block the clock and data signals from the 8042. These bits may be used anytime it is necessary to isolate the 8042 keyboard and mouse signals from the 8042 before entering a system sleep state.

See the Microchip Application Note titled *“Keyboard and Mouse Wakeup Functionality”* for more information.

The bits used to isolate the keyboard and mouse signals from the 8042 are located in Logical Device 1, Register F0h (KRST_GA20) and are defined below. These bits reset on VTR POR only.

Bit[6]M_ISO. Enables/disables isolation of mouse signals into 8042. Does not affect the MDAT signal to The mouse wakeup (PME) logic.

1 = block mouse clock and data signals into 8042

0 = do not block mouse clock and data signals into 8042

Bit[5] K_ISO. Enables/disables isolation of keyboard signals into 8042. Does not affect the KDAT signal to the keyboard wakeup (PME) logic.

1 = block keyboard clock and data signals into 8042

0 = do not block keyboard clock and data signals into 8042

When the keyboard and/or mouse isolation bits are used, it may be necessary to reset the 8042 upon exiting the sleep state. If either of the isolation bits are set prior to entering a sleep state where VCC goes inactive (S3-S5), then the 8042 must be reset upon exiting the sleep mode. Write 40h to global configuration register 2Ch to reset the 8042. The 8042 must then be taken out of reset by writing 00h to register 2Ch since the bit that resets the 8042 is not self-clearing. Caution: Bit 6 of configuration register 2Ch is used to put the 8042 into reset - do not set any of the other bits in register 2Ch, as this may produce undesired results.

It is not necessary to reset the 8042 if the isolation bits are used for a sleep state where VCC does not go inactive (S1, S2).

When the external keyboard and external mouse are powered up, the KDAT and MDAT lines are driven low. This sets the KBD bit (D3) and the MOUSE bit (D4) of the PME Wake Status Register since the KDAT and MDAT signals cannot be isolated internal to the part. This causes an IO_PME# to be generated if the keyboard and/or mouse PME events are enabled. Note that the keyboard and mouse isolation bits only prevent the internal 8042 in the part from setting these status bits.

Case 1: Keyboard and/or Mouse Powered by VTR

The KBD and/or MOUSE status bits will be set upon a VTR POR if the keyboard and/or mouse are powered by VTR. In this case, an IO_PME# will not be generated, since the keyboard and mouse PME enable bits are reset to zero on a VTR POR. The BIOS software needs to clear these PME status bits after power-up.

Case 2: Keyboard and/or Mouse Powered by VCC

The KBD and/or MOUSE status bits will be set upon a VCC POR if the keyboard and/or mouse are powered by VCC. In this case, an IO_PME# will be generated if the enable bits were set for wakeup, since the keyboard and mouse PME enable bits are VTR powered. If the keyboard and mouse are powered by VCC, the enable bits for keyboard and mouse events should be cleared prior to entering a sleep state where VCC is removed (i.e., S3) to prevent a false PME from being generated. In this case, the keyboard and mouse should only be used as PME and/or wake events from the S0 and/or S1 states. The BIOS software needs to clear these PME status bits after power-up.

11.6 'Wake on Specific Key' Option

bit The SCH5636 has logic to detect a single keyboard scan code for wakeup (PME generation). The scan code is programmed onto the Keyboard Scan Code Register, a runtime register at offset 26h from the base address located in the primary base I/O address in Logical Device A. This register is powered by VTR and reset on VTR POR.
bit The PME status bit for this event is located in the PME_STS1 register at bit 5 and the PME enable bit for this event is located in the PME_EN1 register at bit 5. See Section 19.3.3, "PME_STS1 Register," on page 201 and Section 19.3.6, "PME_EN1 Register," on page 204 for a definition of these registers.
bit Data transmissions from the keyboard consist of an 11-bit serial data stream. A logic 1 is sent at an active high level. The following table shows the functions of the bits.

TABLE 11-6: BIT FUNCTION

Bit	Function
1	Start bit (always 0)
2	Data bit 0 (least significant bit)
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7 (most significant bit)
10	Parity bit (odd parity)
11	Stop Bit (always 1)

The process to find a match for the scan code stored in the Keyboard Scan Code register is as follows:

Begin sampling the data at the first falling edge of the keyboard clock following a period where the clock line has been high for 115-145usec. The data at this first clock edge is the start bit. The first data bit follows the start bit (clock 2). Sample the data on each falling edge of the clock. Store the eight bits following the stop bit to compare with the scan code stored in the Keyboard Scan Code register. Sample the comparator within 100usec of the falling edge of clock 9 (for example, at clock 10).

Sample the parity bit and check that the 8 data bits plus the parity bit always have an odd number of 1's (odd parity).

Repeat until a match is found. If the 8 data bits match the scan code stored in the Keyboard Scan Code register and the parity is correct, then it is considered a match. When a match is found and if the stop bit is 1, set the event status bit (bit 5 of the PME_STS1 register) to '1' within 100usec of the falling edge of clock 10.

The state machine will reset after 11 clocks and the process will restart. The process will continue until it is shut off by setting the SPEKEY_EN bit (see description below).

The state machine will reset if there is a period where the clock remains high for more than one keyboard clock period (115-145usec) in the middle of the transmission (i.e., before clock 11). This is to prevent the generation of a false PME.

The SPEKEY_EN bit at bit 1 of the SPEKEY Configuration register at F0h in Logical Device A is used to control the "wake-on-specific feature. This bit is used to turn the logic for this feature on and off. The logic will draw no power when disabled. The bit is defined as follows:

0 = "Wake on specific key" logic is on (default)

1 = "Wake on specific key" logic is off

Note:	The generation of a PME for this event is controlled by the PME enable bit (located in the PME_EN1 register at bit 5) when the logic for feature is turned on.

11.7 Detailed Description of Configuration Registers

11.7.1 ACTIVATE

TABLE 11-7: ACTIVATE REGISTER

HOST OFFSET	30h					8-bit	**HOST SIZE**	
EC OFFSET	330h					32-bit	**EC SIZE**	
POWER	VTR					00b	nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R/W
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Activate

Activate

When this bit is 1, the 8042 logical device is powered and functional. When this bit is 0, the 8042 logical device is powered down and inactive.

11.7.2 KRST_GA20

TABLE 11-8: KRST_GA20 REGISTER

HOST OFFSET	F0h					8-bit	**HOST SIZE**	
EC OFFSET	3F0h					8-bit	**EC SIZE**	
POWER	VTR			Bits[6:5] reset on nSYS_RST only		00b	nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R	R
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R	R
BIT NAME	POL	M_ISO	K_ISO	MLATCH	KLATCH	P92	Reserved	

P92

Port 92 Select.

0= Port 92 Disabled (default)

1= Port 92 Enabled

KLATCH

0= KINT is the 8042 KINT ANDed with Latched KINT (default)

1= KINT is the latched 8042 KINT

MLATCH

0= MINT is the 8042 MINT ANDed with Latched MINT (default)

1= MINT is the latched 8042 MINT

K_ISO

Enables/disables isolation of keyboard signals into 8042. Does not affect KDAT signal to keyboard wakeup (PME) logic.

0= do not block keyboard clock and data signals into 8042 (default)

1= block keyboard clock and data signals into 8042

M_ISO

Enables/disables isolation of mouse signals into 8042. Does not affect MDAT signal to mouse wakeup (PME) logic.

1= block mouse clock and data signals into 8042 (default)

0= do not block mouse clock and data signals into 8042

POL

Polarity Select for P12

0= P12 active low (default)

1= P12 active highz

11.7.3 KEYBOARD SELECT

TABLE 11-9: KEYBOARD SELECT REGISTER

HOST OFFSET	F1h					8-bit	**HOST SIZE**	
EC OFFSET	3F1h					8-bit	**EC SIZE**	
POWER	VTR			00b Bits[6:5] reset on nSYS_RST only			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R/W
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							KMS

KMS

Keyboard//Mouse swap.This bit is used to swap the keyboard and mouse clock and data pins into/out of the 8042 as follows:

0= do not swap the keyboard and mouse clock and data pins (default)

1= internally swap the KCLK pin and the MCLK pin, and the KDAT pin and the MDAT pin into/out of the 8042.

12.0 SERIAL PORT (UART)

12.1 General Description

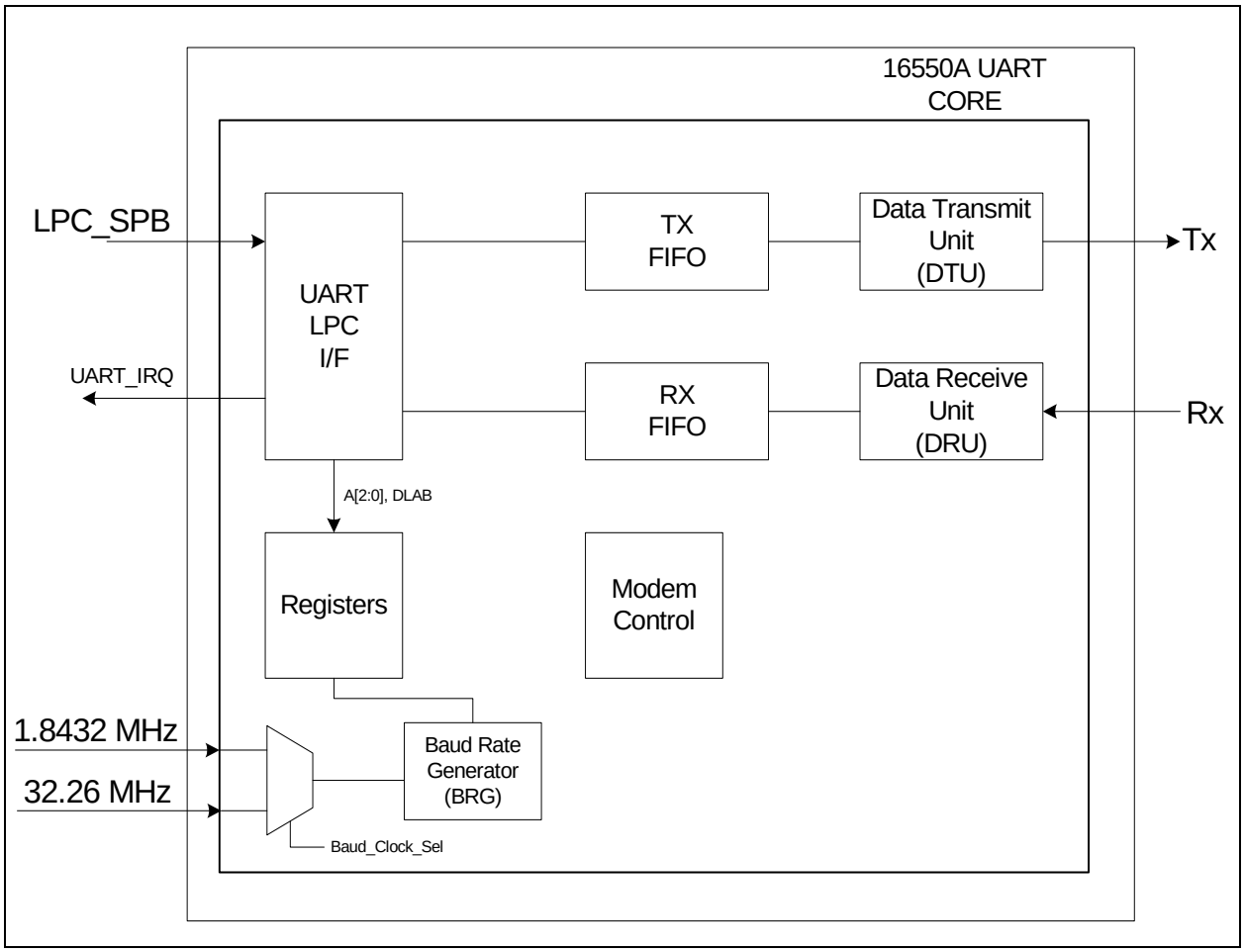
The SCH5636 incorporates two full function UARTs. Each UART is compatible with the 16450, the 16450 ACE registers and the 16C550A. The UARTs perform serial-to-parallel conversion on received characters and parallel-to-serial conversion on transmit characters. Two sets of baud rates are provided. When the 1.8432 MHz source clock is selected, standard baud rates from 50 to 115.2K are available. When the source clock is 32.26 MHz, baud rates from 126K to 2,016K are available. The character options are programmable for 1 start; 1, 1.5 or 2 stop bits; even, odd, sticky or no parity; and prioritized interrupts. The UART contains a programmable baud rate generator that is capable of dividing the input clock or crystal by a number from 1 to 65535. The UART is also capable of supporting the MIDI data rate. Refer to the Configuration Registers for information on disabling, powerdown and changing the base address of the UART. The interrupt from a UART is enabled by programming OUT2 of the UART to a logic “1”. OUT2 being a logic “0” disables the UART's interrupt. The UART is accessible by both the Host and the EC.

12.1.1 FEATURES

- Programmable word length, stop bits and parity
- Programmable baud rate generator
- Interrupt generator
- Loop-back mode
- Interface registers
- 16-byte Transmit FIFO
- 16-byte Receive FIFO
- Multiple clock sources
- VTR & VCC operation
- Pin Polarity control
- Low power sleep mode

12.1.2 BLOCK DIAGRAM

FIGURE 12-1: SERIAL PORT (UART) BLOCK DIAGRAM



12.1.3 BLOCK DIAGRAM SIGNAL LIST

TABLE 12-1: SERIAL PORT (UART) REGISTER INTERFACE PORT LIST

Signal Name	Direction	Description
UART_IRQ	Output	Host Interrupt routed to SERIRQ
LPC_SPB	I/O Bus	Bus used for register access
RX	Input	UART Receive data pin
TX	Output	UART Transmit data pin
1.8432 MHz	Input	UART clock input (1.8462MHz)
32.26 MHz	Input	UART alternate clock input (32.26 MHz)

12.2 Power, Clocks and Reset

12.2.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

12.2.2 CLOCKS

Registers in this block are clocked at the Host Bus Clock rate which is derived by the master clock MCLK. Baud rates are derived from 1.8432MHz clock input. The 1.8432MHz. is itself derived from either MCLK or sourced from the 96MHz PLL. An alternate 32.26MHz clock, derived from MCLK, is also available as a clock input.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

Note:	When the CLK_SRC bit in the Configuration Select Register is '1', the baud clock is derived from the 96MHz PLL.

12.2.3 RESET

Table 12-2 details the effect of nSYS_RST or nSIO_RESET on each of the runtime registers of the Serial Port.

TABLE 12-2: RESET FUNCTION TABLE

Register/Signal	Reset Control	Reset State
Interrupt Enable Register	RESET	All bits low
Interrupt Identification Reg.		Bit 0 is high; Bits 1 - 7 low
FIFO Control		All bits low
Line Control Reg.		
MODEM Control Reg.		
Line Status Reg.		All bits low except 5, 6 high
MODEM Status Reg.		Bits 0 - 3 low; Bits 4 - 7 input
TXD1, TXD2		High
INTRPT (RCVR errs)	RESET/Read LSR	Low
INTRPT (RCVR Data Ready)	RESET/Read RBR	
INTRPT (THRE)	RESET/Read IIR/Write THR	
OUT2B	RESET	High
RTSB		
DTRB		
OUT1B		
RCVR FIFO	RESET/ FCR1*FCR0/_FCR0	All Bits Low
XMIT FIFO	RESET/ FCR1*FCR0/_FCR0	

The Runtime registers can be configured to be reset on either nSYS_RST or nSIO_RESET. The POWER bit in the Configuration Select Register controls which reset effects the runtime registers. Refer to Table 12-2 for effected registers and Section 5.0, "Power, Clocks and Resets" for definitions of nSYS_RST on page 37 or nSIO_RESET on page 38.

See Section 5.9, "Registers," on page 43 for details on reset.

12.3 Interrupts

12.3.1 SERIAL PORT (UART) SIRQ ROUTING

The Serial Port (UART) can generate a SIRQ event to the Host. See the Interrupt Enable Register (IER) on page 127 and the Interrupt Identification Register (IIR) on page 128 for a description of interrupt generation. This interrupt is routed to the SIRQ block (see SERIRQ Configuration Registers on page 88).

12.4 Registers

Table 12-3 is a register summary for one instance of the Serial Port (UART). The LPC I/O address for each Run-Time Register is described below as an offset from its Base Address Register. Each Configuration register access through the Host Access Port is via its LDN indicated in Table 4-2, “Host Logical Devices on SCH5636,” on page 26 and its Host Access Port index which is described as “Host Config Index” in the tables below.

TABLE 12-3: SERIAL PORT REGISTER SUMMARY

Register Name	Offset	DLAB (Note 12-1)	Size	Type	Notes
RUNTIME REGISTERS					
Receive Buffer Register (RB),	00h	0	8	R	
Transmit Buffer Register (TB)	00h	0	8	W	
Programmable Baud Rate Generator	00h	1	8	R/W	
Programmable Baud Rate Generator	01h	1	8	R/W	
Interrupt Enable Register (IER)	01h	0	8	R/W	
FIFO Control Register (FCR),	02h	X	8	W	
Interrupt Identification Register (IIR)	02h	X	8	R	
Line Control Register (LCR)	03h	X	8	R/W	
Modem Control Register (MCR)	04h	X	8	R/W	
Line Status Register (LSR)	05h	X	8	R	
Modem Status Register (MSR)	06h	X	8	R	
Scratchpad Register (SCR)	07h	X	8	R/W	
CONFIGURATION REGISTERS					
Activate	30h	-	8	R/W	
Configuration Select Register	F0h	-	8	R/W	

Note 12-1 DLAB is Bit 7 of the Line Control Register.

TABLE 12-4: REGISTER SUMMARY

Address (Note 12-2)	R/W	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADDR = 0 DLAB = 0	R	Receive Buffer r	Data Bit 7	Data Bit 6	Data Bit 5	Data Bit 4	Data Bit 3	Data Bit 2	Data Bit 1	Data Bit 0 (Note 12-3)
ADDR = 0 DLAB = 0	W	Transmitter Holding r	Data Bit 7	Data Bit 6	Data Bit 5	Data Bit 4	Data Bit 3	Data Bit 2	Data Bit 1	Data Bit 0
ADDR = 1 DLAB = 0	R/W	Interrupt Enable r	Reserved				Enable Modem Status Interrupt (EMSI)	Enable Receiver Line Status Interrupt (ELSI)	Enable Trans-mitter Holding Register Empty Interrupt (ETHREI)	Enable Received Data Available Interrupt (ERDAI)
ADDR = 2	R	Interrupt Ident. r	FIFOs Enabled (Note 12-7)	FIFOs Enabled (Note 12-7)	Reserved		Interrupt ID Bit (Note 12-7)	Interrupt ID Bit	Interrupt ID Bit	"0" if Interrupt Pending
ADDR = 2	W	FIFO Control r	RCVR Trigger MSB	RCVR Trigger LSB	Reserved		DMA Mode Select (Note 12-8)	XMIT FIFO Reset	RCVR FIFO Reset	FIFO Enable
ADDR = 3	R/W	Line Control r	Divisor Latch Access Bit (DLAB)	Set Break	Stick Parity	Even Parity Select (EPS)	Parity Enable (PEN)	Number of Stop Bits (STB)	Word Length Select Bit 1 (WLS1)	Word Length Select Bit 0 (WLS0)
ADDR = 4	R/W	MODEM Control r	**Reserved**			**Loop**	**OUT2 (Note 12-5)**	**OUT1 (Note 12-5)**	**Request to Send (RTS)**	**Data Terminal Ready (DTR)**
ADDR = 5	R/W	Line Status r	**Error in RCVR FIFO (Note 12-7)**	**Transmitter Empty (TEMT) (Note 12-4)**	**Transmitter Holding Register (THRE)**	**Break Interrupt (BI)**	**Framing Error (FE)**	**Parity Error (PE)**	**Overrun Error (OE)**	**Data Ready (DR)**
ADDR = 6	R/W	MODEM Status r	Data Carrier Detect (DCD)	Ring Indicator (RI)	Data Set Ready (DSR)	Clear to Send (CTS)	Delta Data Carrier Detect (DDCD)	Trailing Edge Ring Indicator (TERI)	Delta Data Set Ready (DDSR)	Delta Clear to Send (DCTS)
ADDR = 7	R/W	Scratch r (Note 12-6)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADDR = 0 DLAB = 1	R/W	Divisor Latch (LS)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADDR = 1 DLAB = 1	R/W	Divisor Latch (MS)	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8

UART Register Summary Notes:

Note 12-2 DLAB is Bit 7 of the Line Control Register (ADDR = 3).

Note 12-3 Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

Note 12-4 When operating in the XT mode, this bit will be set any time that the transmitter shift register is empty.

Note 12-5 This bit no longer has a pin associated with it.

Note 12-6 When operating in the XT mode, this register is not available.

Note 12-7 These bits are always zero in the non-FIFO mode.

Note 12-8 Writing a one to this bit has no effect. DMA modes are not supported in this chip.

12.5 Detailed Description of Accessible Runtime Registers

12.5.1 RECEIVE BUFFER REGISTER (RB)

TABLE 12-5: RECEIVE BUFFER (RB)

HOST OFFSET	0h (DLAB=0)			8-bit			**HOST SIZE**	
EC OFFSET	0h (DLAB=0)			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Received Data byte [7:0]							

Received Data Byte

This register holds the received incoming data byte. Bit 0 is the least significant bit, which is transmitted and received first. Received data is double buffered; this uses an additional shift register to receive the serial data stream and convert it to a parallel 8 bit word which is transferred to the Receive Buffer register. The shift register is not accessible.

12.5.2 TRANSMIT BUFFER REGISTER (TB)

TABLE 12-6: TRANSMIT BUFFER (TB)

HOST OFFSET	0h (DLAB=0)			8-bit			**HOST SIZE**	
EC OFFSET	0h (DLAB=0)			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	W	W	W	W	W	/W	W	W
EC TYPE	W	W	W	W	W	W	W	W
BIT NAME	Transmit data byte [7:0]							

Transmit Data Byte

This register contains the data byte to be transmitted. The transmit buffer is double buffered, utilizing an additional shift register (not accessible) to convert the 8 bit data word to a serial format. This shift register is loaded from the Transmit Buffer when the transmission of the previous byte is complete.

12.5.3 INTERRUPT ENABLE REGISTER (IER)

TABLE 12-7: INTERRUPT ENABLE (IER)

HOST OFFSET	1h (DLAB=0)			8-bit			**HOST SIZE**	
EC OFFSET	1h (DLAB=0)			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R/W	R/W	R/W	R/W
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				EMSI	ELSI	ETHREI	ERDAI

The lower four bits of this register control the enables of the five interrupt sources of the Serial Port interrupt. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of this register. Similarly, setting the appropriate bits of this register to a high, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and disables any Serial Port interrupt out of the SCH5636. All other system functions operate in their normal manner, including the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are described below.

ERDAI

This bit enables the Received Data Available Interrupt (and timeout interrupts in the FIFO mode) when set to logic “1”.

ETHREI

This bit enables the Transmitter Holding Register Empty Interrupt when set to logic “1”.

ELSI

This bit enables the Received Line Status Interrupt when set to logic “1”. The error sources causing the interrupt are Overrun, Parity, Framing and Break. The Line Status Register must be read to determine the source.

EMSI

This bit enables the MODEM Status Interrupt when set to logic “1”. This is caused when one of the Modem Status Register bits changes state.

12.5.4 FIFO CONTROL REGISTER (FCR)

TABLE 12-8: FIFO CONTROL (FCR)

HOST OFFSET	02h			8-bit			**HOST SIZE**	
EC OFFSET	02h			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	W	W	W	W	W	W	W	W
EC TYPE	W	W	W	W	W	W	W	W
BIT NAME	RECV FIFO Trigger Level		Reserved			Clear XMIT FIFO	Clear RECV FIFO	EXRF

This is a write only register at the same location as the IIR.

Note:	DMA is not supported.

EXRF

Enable XMIT and RECV FIFO. Setting this bit to a logic “1” enables both the XMIT and RCVR FIFOs. Clearing this bit to a logic “0” disables both the XMIT and RCVR FIFOs and clears all bytes from both FIFOs. When changing from FIFO Mode to non-FIFO (16450) mode, data is automatically cleared from the FIFOs. This bit must be a 1 when other bits in this register are written to or they will not be properly programmed.

Clear RECV FIFO

Setting this bit to a logic “1” clears all bytes in the RCVR FIFO and resets its counter logic to “0”. The shift register is not cleared. This bit is self-clearing.

Clear XMIT FIFO

Setting this bit to a logic “1” clears all bytes in the XMIT FIFO and resets its counter logic to “0”. The shift register is not cleared. This bit is self-clearing.

RECV FIFO Trigger Level

These bits are used to set the trigger level for the RCVR FIFO interrupt.

TABLE 12-9: RECV FIFO TRIGGER LEVEL

Bit 7	Bit 6	RECV FIFO Trigger Level (Bytes)
0	0	1
	1	4
1	0	8
	1	14

12.5.5 INTERRUPT IDENTIFICATION REGISTER (IIR)

TABLE 12-10: INTERRUPT IDENTIFICATION (IIR)

HOST OFFSET	02h				8-bit		**HOST SIZE**	
EC OFFSET	02h				8-bit		**EC SIZE**	
POWER	VCC or VTR				01h		nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	FIFO_En		Reserved		IntID			IPEND

By accessing this register, the host CPU can determine the highest priority interrupt and its source. Four levels of priority interrupt exist. They are in descending order of priority:

1. Receiver Line Status (highest priority)
2. Received Data Ready
3. Transmitter Holding Register Empty
4. MODEM Status (lowest priority)

Information indicating that a prioritized interrupt is pending and the source of that interrupt is stored in the Interrupt Identification Register (refer to Table 12-11). When the CPU accesses the IIR, the Serial Port freezes all interrupts and indicates the highest priority pending interrupt to the CPU. During this CPU access, even if the Serial Port records new interrupts, the current indication does not change until access is completed. The contents of the IIR are described below.

IPEND

This bit can be used in either a hardwired prioritized or polled environment to indicate whether an interrupt is pending. When bit 0 is a logic “0”, an interrupt is pending and the contents of the IIR may be used as a pointer to the appropriate internal service routine. When bit 0 is a logic “1”, no interrupt is pending.

IntID

These three bits of the IIR are used to identify the highest priority interrupt pending as indicated by Table 12-11. In non-FIFO mode, Bit[3] is a logic “0”. In FIFO mode Bit[3] is set along with Bit[2] when a timeout interrupt is pending.

TABLE 12-11: INTERRUPT CONTROL TABLE

FIFO Mode Only	Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Priority Level**	**Interrupt Type**	**Interrupt Source**	**Interrupt Reset Control**
0	0	0	1	-	None	None	-
	1	1	0	Highest	Receiver Line Status	Overrun Error, Parity Error, Framing Error or Break Interrupt	Reading the Line Status Register
		0		Second	Received Data Available	Receiver Data Available	Read Receiver Buffer or the FIFO drops below the trigger level.
1					Character Timeout Indication	No Characters Have Been Removed From or Input to the RCVR FIFO during the last 4 Char times and there is at least 1 char in it during this time	Reading the Receiver Buffer Register
0	0	1		Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if Source of Interrupt) or Writing the Transmitter Holding Register
	0	0		Fourth	MODEM Status	Clear to Send or Data Set Ready or Ring Indicator or Data Carrier Detect	Reading the MODEM Status Register

FIFO_En

These two bits are set when the FIFO CONTROL Register bit 0 equals 1.

12.5.6 LINE CONTROL REGISTER (LCR)

TABLE 12-12: LINE CONTROL (LCR)

HOST OFFSET	03h			8-bit			**HOST SIZE**	
EC OFFSET	03h			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	DLAB	Break Control	Stick Parity	Parity Select	Enable Parity	Stop Bits	Word Length	

This register contains the format information of the serial line. The bit definitions are:

Word Length

These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits 0 and 1 is as follows:

Stop Bits

This bit specifies the number of stop bits in each transmitted or received serial character. Table 12-13 summarizes the information.

TABLE 12-13: STOP BITS

Bit 2	Word Length	Number of Stop Bits
0	--	1
1	5 bits	1.5
	6 bits	2
	7 bits	
	8 bits	

Note 12-9 The receiver will ignore all stop bits beyond the first, regardless of the number used in transmitting.

TABLE 12-14: SERIAL CHARACTER

Bit 1	Bit 0	Word Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

The Start, Stop and Parity bits are not included in the word length.

Enable Parity

Parity Enable bit. When bit 3 is a logic “1”, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and the first stop bit of the serial data. (The parity bit is used to generate an even or odd number of 1s when the data word bits and the parity bit are summed).

Parity Select

Even Parity Select bit. When bit 3 is a logic “1” and bit 4 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 bit 3 is a logic “1” and bit 4 is a logic “1” an even number of bits is transmitted and checked.

Stick Parity

Stick Parity bit. When parity is enabled it is used in conjunction with bit 4 to select Mark or Space Parity. When LCR bits 3, 4 and 5 are 1 the Parity bit is transmitted and checked as a 0 (Space Parity). If bits 3 and 5 are 1 and bit 4 is a 0, then the Parity bit is transmitted and checked as 1 (Mark Parity). If bit 5 is 0 Stick Parity is disabled.

Bit 3 is a logic “1” and bit 5 is a logic “1”, the parity bit is transmitted and then detected by the receiver in the opposite state indicated by bit 4.

Break Control

Set Break Control bit. When bit 6 is a logic “1”, the transmit data output (TXD) is forced to the Spacing or logic “0” state and remains there (until reset by a low level bit 6) regardless of other transmitter activity. This feature enables the Serial Port to alert a terminal in a communications system.

DLAB

Divisor Latch Access Bit (DLAB). It must be set high (logic “1”) to access the Divisor Latches of the Baud Rate Generator during read or write operations. It must be set low (logic “0”) to access the Receiver Buffer Register, the Transmitter Holding Register, or the Interrupt Enable Register.

12.5.7 MODEM CONTROL REGISTER (MCR)

TABLE 12-15: MODEM CONTROL (MCR)

HOST OFFSET	04h			8-bit			**HOST SIZE**	
EC OFFSET	04h			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved			LOOP-BACK	OUT2	OUT1	RTS	DTR

This 8-bit register controls the interface with the MODEM or data set (or device emulating a MODEM). The contents of the MODEM control register are described below.

DTR

This bit controls the Data Terminal Ready (nDTR) output. When bit 0 is set to a logic “1”, the nDTR output is forced to a logic “0”. When bit 0 is a logic “0”, the nDTR output is forced to a logic “1”.

RTS

This bit controls the Request To Send (nRTS) output. Bit 1 affects the nRTS output in a manner identical to that described above for bit 0.

OUT1

This bit controls the Output 1 (OUT1) bit. This bit does not have an output pin and can only be read or written by the CPU.

OUT2

Output 2 (OUT2). This bit is used to enable an UART interrupt. When OUT2 is a logic “0”, the serial port interrupt output is forced to a high impedance state - disabled. When OUT2 is a logic “1”, the serial port interrupt outputs are enabled.

LOOPBACK

This bit provides the loopback feature for diagnostic testing of the Serial Port. When bit 4 is set to logic “1”, the following occur:

1. The TXD is set to the Marking State (logic “1”).
2. The receiver Serial Input (RXD) is disconnected.
3. The output of the Transmitter Shift Register is “looped back” into the Receiver Shift Register input.
4. All MODEM Control inputs (nCTS, nDSR, nRI and nDCD) are disconnected.
5. The four MODEM Control outputs (nDTR, nRTS, OUT1 and OUT2) are internally connected to the four MODEM Control inputs (nDSR, nCTS, RI, DCD).
6. The Modem Control output pins are forced inactive high.
7. Data that is transmitted is immediately received.

This feature allows the processor to verify the transmit and receive data paths of the Serial Port. In the diagnostic mode, the receiver and the transmitter interrupts are fully operational. The MODEM Control Interrupts are also operational but the interrupts' sources are now the lower four bits of the MODEM Control Register instead of the MODEM Control inputs. The interrupts are still controlled by the Interrupt Enable Register.

12.5.8 LINE STATUS REGISTER (LSR)

TABLE 12-16: LINE STATUS (LSR)

HOST OFFSET	05h			8-bit			**HOST SIZE**	
EC OFFSET	05h			8-bit			**EC SIZE**	
POWER	VCC or VTR			60h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	FIFO Error	Trans-mit Error	Trans-mit Empty	Break Interrupt	Frame Error	Parity Error	Overrun Error	Data Ready

Data Ready

Data Ready (DR). It is set to a logic “1” whenever a complete incoming character has been received and transferred into the Receiver Buffer Register or the FIFO. Bit 0 is reset to a logic “0” by reading all of the data in the Receive Buffer Register or the FIFO.

Overrun Error

Overrun Error (OE). Bit 1 indicates that data in the Receiver Buffer Register was not read before the next character was transferred into the register, thereby destroying the previous character. In FIFO mode, an overrun error will occur only when the FIFO is full and the next character has been completely received in the shift register, the character in the shift register is overwritten but not transferred to the FIFO. The OE indicator is set to a logic “1” immediately upon detection of an overrun condition, and reset whenever the Line Status Register is read.

Parity Error

Parity Error (PE). Bit 2 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” whenever the Line Status Register is read. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO.

Frame Error

Framing Error (FE). Bit 3 indicates that the received character did not have a valid stop bit. Bit 3 is set to a logic “1” whenever the stop bit following the last data bit or parity bit is detected as a zero bit (Spacing level). The FE is reset to a logic “0” whenever the Line Status Register is read. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO. The Serial Port will try to 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'.

Break Interrupt

Break Interrupt (BI). Bit 4 is set to a logic “1” whenever the received data input is held in the Spacing state (logic “0”) for longer than a full word transmission time (that is, the total time of the start bit + data bits + parity bits + stop bits). The BI is reset after the CPU reads the contents of the Line Status Register. In the FIFO mode this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO. When break occurs only one zero character is loaded into the FIFO. Restarting after a break is received, requires the serial data (RXD) to be logic “1” for at least 1/2 bit time.

Bits 1 through 4 are the error conditions that produce a Receiver Line Status Interrupt BIT 3

Note 12-10 whenever any of the corresponding conditions are detected and the interrupt is enabled

Transmit Empty

Transmitter Holding Register Empty (THRE). Bit 5 indicates that the Serial Port is ready to accept a new character for transmission. In addition, this bit causes the Serial Port to issue an interrupt when the Transmitter Holding Register interrupt enable is set high. The THRE bit is set to a logic “1” when a character is transferred from the Transmitter Holding Register into the Transmitter Shift Register. The bit is reset to logic “0” whenever the CPU loads the Transmitter Holding Register. In the FIFO mode this bit is set when the XMIT FIFO is empty, it is cleared when at least 1 byte is written to the XMIT FIFO. Bit 5 is a read only bit.

Transmit Error

Transmitter Empty (TEMT). Bit 6 is set to a logic “1” whenever the Transmitter Holding Register (THR) and Transmitter Shift Register (TSR) are both empty. It is reset to logic “0” whenever either the THR or TSR contains a data character. Bit 6 is a read only bit. In the FIFO mode this bit is set whenever the THR and TSR are both empty,

FIFO Error

This bit is permanently set to logic “0” in the 450 mode. In the FIFO mode, this bit is set to a logic “1” when there is at least one parity error, framing error or break indication in the FIFO. This bit is cleared when the LSR is read if there are no subsequent errors in the FIFO.

12.5.9 MODEM STATUS REGISTER (MSR)

TABLE 12-17: MODEM STATUS (MSR)

HOST ADDRESS	06h			8-bit			**HOST SIZE**	
EC OFFSET	06h			8-bit			**EC SIZE**	
POWER	VCC or VTR			xxxx0000b			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	DCD#	RI#	DSR	CTS	DCD	RI	DSR	CTS

This 8 bit register provides the current state of the control lines from the MODEM (or peripheral device). In addition to this current state information, four bits of the MODEM Status Register (MSR) provide change information.

These bits are set to logic “1” whenever a control input from the MODEM changes state. They are reset to logic “0” whenever the MODEM Status Register is read.

CTS

Delta Clear To Send (DCTS). Bit 0 indicates that the nCTS input to the chip has changed state since the last time the MSR was read.

DSR

Delta Data Set Ready (DDSR). Bit 1 indicates that the nDSR input has changed state since the last time the MSR was read.

RI

Trailing Edge of Ring Indicator (TERI). Bit 2 indicates that the nRI input has changed from logic “0” to logic “1”.

DCD

Delta Data Carrier Detect (DDCD). Bit 3 indicates that the nDCD input to the chip has changed state.

Note 12-11 Whenever bit 0, 1, 2, or 3 is set to a logic “1”, a MODEM Status Interrupt is generated.

CTS

This bit is the complement of the Clear To Send (nCTS) input. If bit 4 of the MCR is set to logic “1”, this bit is equivalent to nRTS in the MCR.

DSR

This bit is the complement of the Data Set Ready (nDSR) input. If bit 4 of the MCR is set to logic “1”, this bit is equivalent to DTR in the MCR.

RI#

This bit is the complement of the Ring Indicator (nRI) input. If bit 4 of the MCR is set to logic “1”, this bit is equivalent to OUT1 in the MCR.

DCD

This bit is the complement of the Data Carrier Detect (nDCD) input. If bit 4 of the MCR is set to logic “1”, this bit is equivalent to OUT2 in the MCR.

APPLICATION NOTE: The Modem Status Register (MSR) only provides the current state of the UART MODEM control lines in Loopback Mode. The SCH5636 does not support external connections for the MODEM Control inputs (nCTS, nDSR, nRI and nDCD) or for the four MODEM Control outputs (nDTR, nRTS, OUT1 and OUT2).

12.5.10 SCRATCHPAD REGISTER (SCR)

TABLE 12-18: SCRATCH PAD (SCR)

HOST OFFSET	07h			8-bit			**HOST SIZE**	
EC OFFSET	07h			8-bit			**EC SIZE**	
POWER	VCC or VTR			00h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Scratch							

Scratch

This 8 bit read/write register has no effect on the operation of the Serial Port. It is intended as a scratchpad register to be used by the programmer to hold data temporarily.

12.5.11 PROGRAMMABLE BAUD RATE GENERATOR

TABLE 12-19: PROGRAMMABLE BAUD RATE GENERATOR

HOST OFFSET	BYTE1: 01h (DLAB = 1) BYTE0: 00h (DLAB = 1)					8-bit	**HOST SIZE**	
EC OFFSET	BYTE1: 01h (DLAB = 1) BYTE0: 00h (DLAB = 1)			8-bit			**EC SIZE**	
POWER	VCC or VTR			0000h			nSYS_RST or nSIO_RESET **DEFAULT**	
BYTE1 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Baud_ Clock_ Sel	Baud_Rate_Divisor[15:8]						
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Baud_Rate_Divisor[7:0]							

Baud_Rate_Divisor

The Serial Port contains a programmable Baud Rate Generator that is capable of dividing the internal clock source by any divisor from 1 to 65535. The clock source is either a 1.8432MHz clock derived from the 64.52MHz ring oscillator or a 32.26Mhz clock also derived from the ring oscillator. The output frequency of the Baud Rate Generator is 16x the Baud rate. Two eight bit latches store the divisor in 16 bit binary format. These Divisor Latches must be loaded during initialization in order to insure desired operation of the Baud Rate Generator. Upon loading either of the Divisor Latches, a 16 bit Baud counter is immediately loaded. This prevents long counts on initial load. If a 0 is loaded into the BRG registers, the output divides the clock by the number 3. If a 1 is loaded, the output is the inverse of the input oscillator. If a two is loaded, the output is a divide by 2 signal with a 50% duty cycle. If a 3 or greater is loaded, the output is low for 2 bits and high for the remainder of the count.

Table 12-20 and Table 12-21 shows the baud rates possible.

TABLE 12-20: UART BAUD RATES (1.8432MHZ SOURCE)

Desired Baud Rate	Divisor Used to Generate 16X Clock
50	2304
75	1536
110	1047
134.5	857
150	768
300	384
600	192
1200	96
1800	64
2000	58
2400	48
3600	32
4800	24
7200	16
9600	12
19200	6
38400	3
57600	2
115200	1

TABLE 12-21: UART BAUD RATES (32.26MHZ SOURCE)

Desired Baud Rate	BAUD_CLOCK_SEL	Divisor Used to Generate 16X Clock
126000	1	16
168000	1	12
183000	1	11
201600	1	10
224000	1	9
252000	1	8
288000	1	7
336000	1	6
403800	1	5
504100	1	4
672100	1	3

TABLE 12-21: UART BAUD RATES (32.26MHZ SOURCE) (CONTINUED)

Desired Baud Rate	BAUD_CLOCK_SEL	Divisor Used to Generate 16X Clock
1008000	1	2
2016000	1	1

Baud_Clock_Sel

If the CLK_SRC bit is '0' and the Baud_Clock_Sel bit is '0,' the 1.8432MHz clock is used to generate the baud clock. Table 12-20 shows some baud rates that can be generated with this clock. The CLK_SRC bit is D0 in the UART Logical Device configuration register offset 0xF0.

If the CLK_SRC bit is '0' and the Baud_Clock_Sel bit is '1,' the 32.26MHz clock is used to generate the baud clock. Table 12-21 shows some baud rates that can be generated with this clock.

If the CLK_SRC bit is '1,' the Baud_Clock_Sel bit as no effect.

12.6 Detailed Description of Configuration Registers

12.6.1 ACTIVATE

TABLE 12-22: ACTIVATE REGISTER

HOST OFFSET	30h					8-bit	**HOST SIZE**	
EC OFFSET	330h					32-bit	**EC SIZE**	
POWER	VTR					00b	nSYS_RST or nSIO_RESET **DEFAULT (SEE Note 12-12)**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R/W
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Activate

Activate

When this bit is 1, the UART logical device is powered and functional. When this bit is 0, the UART logical device is powered down and inactive.

Note 12-12 If the Power bit in the Configuration Select register is 1, then the Activate register is reset on nSIO_RESET. If the Power bit is 0, then the Activate register is reset on nSYS_RST.

12.6.2 CONFIGURATION SELECT

TABLE 12-23: CONFIGURATION SELECT REGISTER

HOST OFFSET	F0h					8-bit	**HOST SIZE**	
EC OFFSET	3F0h					8-bit	**EC SIZE**	
POWER	VTR					00b	nSYS_RST **DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	R	R	R	R	R	R	R	R/W
EC TYPE	R	R	R	R	R	R/W	R/W	R/W
BIT NAME	Reserved					Polarity	Power	CLK_SRC

CLK_SRC

When this bit is 0, the UART clock is derived from the internal ring oscillator. When this bit is 1, the UART clock is derived from an external clock source

POWER

When this bit is 1, the UART Runtime Registers (the registers at offsets 0h through 7h from the base of the UART Logical Device) are controlled by VCC. They are set to their POR defaults on a nSIO_RESET.

When this bit is 0, the UART Runtime Registers are controlled by VTR. They are set to their POR defaults on an nSYS-_RST.

POLARITY

When the Polarity bit is asserted (‘1’), the UART_TX and UART_RX pins functions are inverted. When the Polarity bit is not asserted (default), the UART_TX and UART_RX pins functions are not inverted.

12.7 Sleep Enable/Clock Request Power state controls

TABLE 12-24: GENERIC BLOCK CLOCKING MODEL BEHAVIOR

External Sleep Input	Block Idle Status	Clock Required Status Output	State	Description
X	X	0	DISABLED	Block is disabled by firmware and the core clock is not needed. Note: it is up to the host to maintain that the block is not in use before the internal ENABLE bit is asserted.
0	NOT IDLE	1	NORMAL OPERATION	The block is neither disabled by firmware nor commanded to sleep.
	IDLE	0		
1	NOT IDLE	1	PREPARING TO SLEEP	A sleep command has been asserted but the core clock is still required because the block is not idle.
	IDLE	0	SLEEPING	A sleep command has been asserted, the block is idle and the core clock can be stopped.

13.0 PARALLEL PORT

The SCH5636 incorporates an IBM XT/AT compatible parallel port. This supports the optional PS/2 type bi-directional parallel port (SPP), the Enhanced Parallel Port (EPP) and the Extended Capabilities Port (ECP) parallel port modes. Refer to the Configuration Registers for information on disabling, power- down, changing the base address of the parallel port, and selecting the mode of operation.

The parallel port also incorporates Microchip's ChiProtect circuitry, which prevents possible damage to the parallel port due to printer power-up.

The functionality of the Parallel Port is achieved through the use of eight addressable ports, with their associated registers and control gating. The control and data port are read/write by the CPU, the status port is read/write in the EPP mode. The address map of the Parallel Port is shown below:

DATA PORT	BASE ADDRESS + 00H
STATUS PORT	BASE ADDRESS + 01H
CONTROL PORT	BASE ADDRESS + 02H
EPP ADDR PORT	BASE ADDRESS + 03H
EPP DATA PORT 0	BASE ADDRESS + 04H
EPP DATA PORT 1	BASE ADDRESS + 05H
EPP DATA PORT 2	BASE ADDRESS + 06H
EPP DATA PORT 3	BASE ADDRESS + 07H

TABLE 13-1: REGISTER BIT MAP

	D0	D1	D2	D3	D4	D5	D6	D7	Note
DATA PORT	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-1
STATUS PORT	TMOUT	0	0	nERR	SLCT	PE	ACK#	nBUSY	13-1
CONTROL PORT	STROBE	AUTOFD	INIT#	SLC	IRQE	PCD	0	0	13-1
EPP ADDR PORT	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-2
EPP DATA PORT 0	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-2
EPP DATA PORT 1	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-2
EPP DATA PORT 2	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-2
EPP DATA PORT 3	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	13-2

Note 13-1 These registers are available in all modes.

Note 13-2 These registers are only available in EPP mode.

TABLE 13-2: PARALLEL PORT CONNECTOR

Host Connector	Pin Number	Standard	EPP	ECP
1	83	STROBE#	nWrite	STROBE#
2-9	68-75	PD<0:7>	PData<0:7>	PData<0:7>
10	80	ACK#	Intr	ACK#
11	79	BUSY	nWait	Busy, PeriphAck(3)
12	78	PE	(User Defined)	PError, ACK#Reverse (3)
13	77	SLCT	(User Defined)	Select
14	82	ALF#	nDatastb	nAutoFd, HostAck(3)

TABLE 13-2: PARALLEL PORT CONNECTOR (CONTINUED)

Host Connector	Pin Number	Standard	EPP	ECP
15	81	ERROR#	(User Defined)	nFault (1) nPeriphRequest (3)
16	66	INIT#	nRESET	INIT#(1) nReverseRqst(3)
17	67	SLCTIN#	nAddrstrb	nSelectIn(1,3)
(1) = Compatible Mode (3) = High Speed Mode				

Note 13-3 For the cable interconnection required for ECP support and the Slave Connector pin numbers, refer to the *IEEE 1284 Extended Capabilities Port Protocol and ISA Standard, Rev. 1.14*, July 14, 1993. This document is available from Microsoft.

13.1 IBM XT/AT Compatible, Bi-Directional and EPP Modes

13.1.1 DATA PORT

ADDRESS OFFSET = 00H

The Data Port is located at an offset of '00H' from the base address. The data register is cleared at initialization by RESET. During a WRITE operation, the Data Register latches the contents of the internal data bus. The contents of this register are buffered (non inverting) and output onto the PD0 - PD7 ports. During a READ operation in SPP mode, PD0 - PD7 ports are buffered (not latched) and output to the host CPU.

13.1.2 STATUS PORT

ADDRESS OFFSET = 01H

The Status Port is located at an offset of '01H' from the base address. The contents of this register are latched for the duration of a read cycle. The bits of the Status Port are defined as follows:

Bit 0 TMOUT - TIME OUT

This bit is valid in EPP mode only and indicates that a 10 msec time out has occurred on the EPP bus. A logic O means that no time out error has occurred; a logic 1 means that a time out error has been detected. This bit is cleared by a RESET. If the TIMEOUT_SELECT bit (bit 4 of the Parallel Port Mode Register 2, 0xF1 in Logical Device 3 Configuration Registers) is '0', writing a one to this bit clears the TMOUT status bit. Writing a zero to this bit has no effect. If the TIMEOUT_SELECT bit (bit 4 of the Parallel Port Mode Register 2, 0xF1 in Logical Device 3 Configuration Registers) is '1', the TMOUT bit is cleared on the trailing edge of a read of the EPP Status Register.

Bits 1, 2 - are not implemented as register bits, during a read of the Printer Status Register these bits are a low level.

Bit 3 nERR – ERROR#

The level on the ERROR# input is read by the CPU as bit 3 of the Printer Status Register. A logic 0 means an error has been detected; a logic 1 means no error has been detected.

Bit 4 SLT - Printer Selected Status

The level on the SLCT input is read by the CPU as bit 4 of the Printer Status Register. A logic 1 means the printer is on line; a logic 0 means it is not selected.

Bit 5 PE - Paper End

The level on the PE input is read by the CPU as bit 5 of the Printer Status Register. A logic 1 indicates a paper end; a logic 0 indicates the presence of paper.

Bit 6 ACK# - Acknowledge

The level on the ACK# input is read by the CPU as bit 6 of the Printer Status Register. A logic 0 means that the printer has received a character and can now accept another. A logic 1 means that it is still processing the last character or has not received the data.

Bit 7 nBUSY - nBUSY

The complement of the level on the BUSY input is read by the CPU as bit 7 of the Printer Status Register. A logic 0 in this bit means that the printer is busy and cannot accept a new character. A logic 1 means that it is ready to accept the next character.

13.1.3 CONTROL PORT

ADDRESS OFFSET = 02H

The Control Port is located at an offset of ‘02H’ from the base address. The Control Register is initialized by the RESET input, bits 0 to 5 only being affected; bits 6 and 7 are hard wired low.

Bit 0 STROBE - Strobe

This bit is inverted and output onto the STROBE# output.

Bit 1 AUTOFD - Autofeed

This bit is inverted and output onto the nAutoFd output. A logic 1 causes the printer to generate a line feed after each line is printed. A logic 0 means no autofeed.

Bit 2 INIT# - Initiate Output

This bit is output onto the INIT# output without inversion.

Bit 3 SLCTIN - Printer Select Input

This bit is inverted and output onto the SLCTIN# output. A logic 1 on this bit selects the printer; a logic 0 means the printer is not selected.

Bit 4 IRQE - Interrupt Request Enable

The interrupt request enable bit when set to a high level may be used to enable interrupt requests from the Parallel Port to the CPU. An interrupt request is generated on the IRQ port by a positive going ACK# input. When the IRQE bit is programmed low the IRQ is disabled.

Bit 5 PCD - PARALLEL CONTROL DIRECTION

Parallel Control Direction is not valid in printer mode. In printer mode, the direction is always out regardless of the state of this bit. In bi-directional, EPP or ECP mode, a logic 0 means that the printer port is in output mode (write); a logic 1 means that the printer port is in input mode (read).

Bits 6 and 7 during a read are a low level, and cannot be written.

13.1.4 EPP ADDRESS PORT

ADDRESS OFFSET = 03H

The EPP Address Port is located at an offset of ‘03H’ from the base address. The address register is cleared at initialization by RESET. During a WRITE operation, the contents of the internal data bus DB0-DB7 are buffered (non inverting) and output onto the PD0 - PD7 ports. An LPC I/O write cycle causes an EPP ADDRESS WRITE cycle to be performed, during which the data is latched for the duration of the EPP write cycle. During a READ operation, PD0 - PD7 ports are read. An LPC I/O read cycle causes an EPP ADDRESS READ cycle to be performed and the data output to the host CPU, the deassertion of ADDRSTB latches the PData for the duration of the read cycle. This register is only available in EPP mode.

13.1.5 EPP DATA PORT 0

ADDRESS OFFSET = 04H

The EPP Data Port 0 is located at an offset of ‘04H’ from the base address. The data register is cleared at initialization by RESET. During a WRITE operation, the contents of the internal data bus DB0-DB7 are buffered (non inverting) and output onto the PD0 - PD7 ports. An LPC I/O write cycle causes an EPP DATA WRITE cycle to be performed, during which the data is latched for the duration of the EPP write cycle. During a READ operation, PD0 - PD7 ports are read. An LPC I/O read cycle causes an EPP READ cycle to be performed and the data output to the host CPU, the deassertion of DATASTB latches the PData for the duration of the read cycle. This register is only available in EPP mode.

13.1.6 EPP DATA PORT 1

ADDRESS OFFSET = 05H

The EPP Data Port 1 is located at an offset of ‘05H’ from the base address. Refer to EPP DATA PORT 0 for a description of operation. This register is only available in EPP mode.

13.1.7 EPP DATA PORT 2

ADDRESS OFFSET = 06H

The EPP Data Port 2 is located at an offset of '06H' from the base address. Refer to EPP DATA PORT 0 for a description of operation. This register is only available in EPP mode.

13.1.8 EPP DATA PORT 3

ADDRESS OFFSET = 07H

The EPP Data Port 3 is located at an offset of '07H' from the base address. Refer to EPP DATA PORT 0 for a description of operation. This register is only available in EPP mode.

13.1.9 EPP 1.9 OPERATION

bit When the EPP mode is selected in the configuration register, the standard and bi-directional modes are also available. If no EPP Read, Write or Address cycle is currently executing, then the PDx bus is in the standard or bi-directional mode, and all output signals (STROBE, AUTOFD, INIT) are as set by the SPP Control Port and direction is controlled by PCD of the Control port.
bit In EPP mode, the system timing is closely coupled to the EPP timing. For this reason, a watchdog timer is required to prevent system lockup. The timer indicates if more than 10usec have elapsed from the start of the EPP cycle to nWAIT being deasserted (after command). If a time-out occurs, the current EPP cycle is aborted and the time-out condition is indicated in Status bit 0.
bit During an EPP cycle, if STROBE is active, it overrides the EPP write signal forcing the PDx bus to always be in a write mode and the nWRITE signal to always be asserted.

13.1.10 SOFTWARE CONSTRAINTS

Before an EPP cycle is executed, the software must ensure that the control register bit PCD is a logic "0" (i.e., a 04H or 05H should be written to the Control port). If the user leaves PCD as a logic "1", and attempts to perform an EPP write, the chip is unable to perform the write (because PCD is a logic "1") and will appear to perform an EPP read on the parallel bus, no error is indicated.

13.1.11 EPP 1.9 WRITE

The timing for a write operation (address or data) is shown in timing diagram EPP Write Data or Address cycle. The chip inserts wait states into the LPC I/O write cycle until it has been determined that the write cycle can complete. The write cycle can complete under the following circumstances:

- If the EPP bus is not ready (nWAIT is active low) when nDATASTB or nADDRSTB goes active then the write can complete when nWAIT goes inactive high.
- If the EPP bus is ready (nWAIT is inactive high) then the chip must wait for it to go active low before changing the state of nDATASTB, nWRITE or nADDRSTB. The write can complete once nWAIT is determined inactive.

Write Sequence of Operation

1. The host initiates an I/O write cycle to the selected EPP register.
2. If WAIT is not asserted, the chip must wait until WAIT is asserted.
3. The chip places address or data on PData bus, clears PDIR, and asserts nWRITE.
4. Chip asserts nDATASTB or nADDRSTRB indicating that PData bus contains valid information, and the WRITE signal is valid.
5. Peripheral deasserts nWAIT, indicating that any setup requirements have been satisfied and the chip may begin the termination phase of the cycle.
6.

a) The chip deasserts nDATASTB or nADDRSTRB, this marks the beginning of the termination phase. If it has not already done so, the peripheral should latch the information byte now.

b) The chip latches the data from the internal data bus for the PData bus and drives the sync that indicates that no more wait states are required followed by the TAR to complete the write cycle.

7. Peripheral asserts nWAIT, indicating to the host that any hold time requirements have been satisfied and acknowledging the termination of the cycle.
8. Chip may modify nWRITE and nPDATA in preparation for the next cycle.

13.1.12 EPP 1.9 READ

The timing for a read operation (data) is shown in timing diagram EPP Read Data cycle. The chip inserts wait states into the LPC I/O read cycle until it has been determined that the read cycle can complete. The read cycle can complete under the following circumstances:

- If the EPP bus is not ready (nWAIT is active low) when nDATASTB goes active then the read can complete when nWAIT goes inactive high.
- If the EPP bus is ready (nWAIT is inactive high) then the chip must wait for it to go active low before changing the state of nWRITE or before nDATASTB goes active. The read can complete once nWAIT is determined inactive.

Read Sequence of Operation

1. The host initiates an I/O read cycle to the selected EPP register.
2. If WAIT is not asserted, the chip must wait until WAIT is asserted.
3. The chip tri-states the PData bus and deasserts nWRITE.
4. Chip asserts nDATASTB or nADDRSTRB indicating that PData bus is tri-stated, PDIR is set and the nWRITE signal is valid.
5. Peripheral drives PData bus valid.
6. Peripheral deasserts nWAIT, indicating that PData is valid and the chip may begin the termination phase of the cycle.
7. The chip latches the data from the PData bus for the internal data bus and deasserts nDATASTB or nADDRSTRB. This marks the beginning of the termination phase.
8. The chip drives the sync that indicates that no more wait states are required and drives valid data onto the LAD[3:0] signals, followed by the TAR to complete the read cycle.
9. Peripheral tri-states the PData bus and asserts nWAIT, indicating to the host that the PData bus is tri-stated.
10. Chip may modify nWRITE, PDIR and nPDATA in preparation for the next cycle.

13.1.13 EPP 1.7 OPERATION

When the EPP 1.7 mode is selected in the configuration register, the standard and bi-directional modes are also available. If no EPP Read, Write or Address cycle is currently executing, then the PDx bus is in the standard or bi-directional mode, and all output signals (STROBE, AUTOFD, INIT) are as set by the SPP Control Port and direction is controlled by PCD of the Control port.

In EPP mode, the system timing is closely coupled to the EPP timing. For this reason, a watchdog timer is required to prevent system lockup. The timer indicates if more than 10usec have elapsed from the start of the EPP cycle to the end of the cycle. If a time-out occurs, the current EPP cycle is aborted and the time-out condition is indicated in Status bit 0.

13.1.14 SOFTWARE CONSTRAINTS

Before an EPP cycle is executed, the software must ensure that the control register bits D0, D1 and D3 are set to zero. Also, bit D5 (PCD) is a logic “0” for an EPP write or a logic “1” for and EPP read.

13.1.15 EPP 1.7 WRITE

The timing for a write operation (address or data) is shown in timing diagram EPP 1.7 Write Data or Address cycle. The chip inserts wait states into the I/O write cycle when nWAIT is active low during the EPP cycle. This can be used to extend the cycle time. The write cycle can complete when nWAIT is inactive high.

Write Sequence of Operation

- The host sets PDIR bit in the control register to a logic “0”. This asserts nWRITE.
- The host initiates an I/O write cycle to the selected EPP register.
- The chip places address or data on PData bus.
- Chip asserts nDATASTB or nADDRSTRB indicating that PData bus contains valid information, and the WRITE signal is valid.
- If nWAIT is asserted, the chip inserts wait states into I/O write cycle until the peripheral deasserts nWAIT or a time-out occurs.
- The chip drives the final sync, deasserts nDATASTB or nADDRSTRB and latches the data from the internal data bus for the PData bus.
- Chip may modify nWRITE, PDIR and nPDATA in preparation of the next cycle.

13.1.16 EPP 1.7 READ

The timing for a read operation (data) is shown in timing diagram EPP 1.7 Read Data cycle. The chip inserts wait states into the I/O read cycle when nWAIT is active low during the EPP cycle. This can be used to extend the cycle time. The read cycle can complete when nWAIT is inactive high.

Read Sequence of Operation

- The host sets PDIR bit in the control register to a logic “1”. This deasserts nWRITE and tri-states the PData bus.
- The host initiates an I/O read cycle to the selected EPP register.
- Chip asserts nDATASTB or nADDRSTRB indicating that PData bus is tri-stated, PDIR is set and the nWRITE signal is valid.
- If nWAIT is asserted, the chip inserts wait states into the I/O read cycle until the peripheral deasserts nWAIT or a time-out occurs.
- The Peripheral drives PData bus valid.
- The Peripheral deasserts nWAIT, indicating that PData is valid and the chip may begin the termination phase of the cycle.
- The chip drives the final sync and deasserts nDATASTB or nADDRSTRB.
- Peripheral tri-states the PData bus.
- Chip may modify nWRITE, PDIR and nPDATA in preparation of the next cycle.

TABLE 13-3: EPP PIN DESCRIPTIONS

EPP Signal	EPP Name	Type	EPP Description
nWRITE	nWrite	O	This signal is active low. It denotes a write operation.
PD<0:7>	Address/Data	I/O	Bi-directional EPP byte wide address and data bus.
INTR	Interrupt	I	This signal is active high and positive edge triggered. (Pass through with no inversion, Same as SPP).
nWAIT	nWait	I	This signal is active low. It is driven inactive as a positive acknowledgement from the device that the transfer of data is completed. It is driven active as an indication that the device is ready for the next transfer.
nDATASTB	nData Strobe	O	This signal is active low. It is used to denote data read or write operation.
nRESET	nReset	O	This signal is active low. When driven active, the EPP device is reset to its initial operational mode.
nADDRSTB	Address Strobe	O	This signal is active low. It is used to denote address read or write operation.
PE	Paper End	I	Same as SPP mode.
SLCT	Printer Selected Status	I	Same as SPP mode.
nERR	Error	I	Same as SPP mode.
Note 13-4	SPP and EPP can use 1 common register.		
Note 13-5	nWrite is the only EPP output that can be over-ridden by SPP control port during an EPP cycle. For correct EPP read cycles, PCD is required to be a low.		

13.2 Extended Capabilities Parallel Port

ECP provides a number of advantages, some of which are listed below. The individual features are explained in greater detail in the remainder of this section.

High performance half-duplex forward and reverse channel Interlocked handshake, for fast reliable transfer Optional single byte RLE compression for improved throughput (64:1) Channel addressing for low-cost peripherals Maintains link and data layer separation Permits the use of active output drivers permits the use of adaptive signal timing Peer-to-peer capability.

13.2.1 VOCABULARY

The following terms are used in this document:

assert: When a signal asserts it transitions to a “true” state, when a signal deasserts it transitions to a “false” state.

forward: Host to Peripheral communication.

reverse: Peripheral to Host communication

Pword: A port word; equal in size to the width of the LPC interface. For this implementation, PWord is always 8 bits.

1 A high level

0 A low level

These terms may be considered synonymous:

PeriphClk, ACK#

HostAck, nAutoFd

PeriphAck, Busy

nPeriphRequest, nFault

nReverseRequest, INIT#

ACK#Reverse, PError

Xflag, Select

ECPMode, nSelectIn

HostClk, STROBE#

Reference Document: *IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard*, Rev 1.14, July 14, 1993. This document is available from Microsoft.

TABLE 13-4: BIT MAP OF THE EXTENDED PARALLEL PORT REGISTERS

	D7	D6	D5	D4	D3	D2	D1	D0	Note
data	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0	
ecpAFifo	Addr/RLE	Address or RLE field							13-7
dsr	nBusy	ACK#	PError	Select	nFault	0	0	0	13-6
dcr	0	0	Direction	ackIntEn	Selection	INIT#	autofd	strobe	13-6
cFifo	Parallel Port Data FIFO								13-7
ecpDFifo	ECP Data FIFO								13-7
tFifo	Test FIFO								13-7
cnfgA	0	0	0	1	0	0	0	0	
cnfgB	compress	intrValue	Parallel Port IRQ			Parallel Port DMA			
ecr	MODE			nErrIntrEn	dmaEn	serviceIntr	full	empty	

Note 13-6 These registers are available in all modes.

Note 13-7 All FIFOs use one common 16 byte FIFO.

Note 13-8 The ECP Parallel Port Config Reg B reflects the IRQ and DMA channel selected by the Configuration Registers.

13.2.2 ECP IMPLEMENTATION STANDARD

This specification describes the standard interface to the Extended Capabilities Port (ECP). All LPC devices supporting ECP must meet the requirements contained in this section or the port will not be supported by Microsoft. For a description of the ECP Protocol, please refer to the *IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard*, *Rev. 1.14*, July 14, 1993. This document is available from Microsoft.

Description

The port is software and hardware compatible with existing parallel ports so that it may be used as a standard LPT port if ECP is not required. The port is designed to be simple and requires a small number of gates to implement. It does not do any “protocol” negotiation, rather it provides an automatic high burst-bandwidth channel that supports DMA for ECP in both the forward and reverse directions.

Small FIFOs are employed in both forward and reverse directions to smooth data flow and improve the maximum bandwidth requirement. The size of the FIFO is 16 bytes deep. The port supports an automatic handshake for the standard parallel port to improve compatibility mode transfer speed.

The port also supports run length encoded (RLE) decompression (required) in hardware. Compression is accomplished by counting identical bytes and transmitting an RLE byte that indicates how many times the next byte is to be repeated. Decompression simply intercepts the RLE byte and repeats the following byte the specified number of times. Hardware support for compression is optional.

TABLE 13-5: ECP PIN DESCRIPTIONS

Name	Type	Description
STROBE#	O	During write operations STROBE# registers data or address into the slave on the asserting edge (handshakes with Busy).
PData 7:0	I/O	Contains address or data or RLE data.
ACK#	I	Indicates valid data driven by the peripheral when asserted. This signal handshakes with nAutoFd in reverse.
PeriphAck (Busy)	I	This signal deasserts to indicate that the peripheral can accept data. This signal handshakes with STROBE# in the forward direction. In the reverse direction this signal indicates whether the data lines contain ECP command information or data. The peripheral uses this signal to flow control in the forward direction. It is an “interlocked” handshake with STROBE#. PeriphAck also provides command information in the reverse direction.
PError (ACK#Reverse)	I	Used to acknowledge a change in the direction the transfer (asserted = forward). The peripheral drives this signal low to acknowledge nReverseRequest. It is an “interlocked” handshake with nReverseRequest. The host relies upon ACK#Reverse to determine when it is permitted to drive the data bus.
Select	I	Indicates printer on line.
nAutoFd (HostAck)	O	Requests a byte of data from the peripheral when asserted, handshaking with ACK# in the reverse direction. In the forward direction this signal indicates whether the data lines contain ECP address or data. The host drives this signal to flow control in the reverse direction. It is an “interlocked” handshake with ACK#. HostAck also provides command information in the forward phase.
nFault (nPeriphRequest)	I	Generates an error interrupt when asserted. This signal provides a mechanism for peer-to-peer communication. This signal is valid only in the forward direction. During ECP Mode the peripheral is permitted (but not required) to drive this pin low to request a reverse transfer. The request is merely a “hint” to the host; the host has ultimate control over the transfer direction. This signal would be typically used to generate an interrupt to the host CPU.
INIT#	O	Sets the transfer direction (asserted = reverse, deasserted = forward). This pin is driven low to place the channel in the reverse direction. The peripheral is only allowed to drive the bidirectional data bus while in ECP Mode and HostAck is low and nSelectIn is high.
nSelectIn	O	Always deasserted in ECP mode.

13.2.3 REGISTER DEFINITIONS

The register definitions are based on the standard IBM addresses for LPT. All of the standard printer ports are supported. The additional registers attach to an upper bit decode of the standard LPT port definition to avoid conflict with standard ISA devices. The port is equivalent to a generic parallel port interface and may be operated in that mode. The port registers vary depending on the mode field in the ecr. Table 13-6 lists these dependencies. Operation of the devices in modes other that those specified is undefined.

TABLE 13-6: ECP REGISTER DEFINITIONS

Name	Address (13-9)	ECP Modes	Function
data	+000h R/W	000-001	Data Register
ecpAFifo	+000h R/W	011	ECP FIFO (Address)
dsr	+001h R/W	All	Status Register
dcr	+002h R/W	All	Control Register
cFifo	+400h R/W	010	Parallel Port Data FIFO
ecpDFifo	+400h R/W	011	ECP FIFO (DATA)
tFifo	+400h R/W	110	Test FIFO
cnfgA	+400h R	111	Configuration Register A
cnfgB	+401h R/W	111	Configuration Register B
ecr	+402h R/W	All	Extended Control Register

Note 13-9 These addresses are added to the parallel port base address as selected by configuration register or jumpers.

Note 13-10 All addresses are qualified with AEN. Refer to the AEN pin definition.

TABLE 13-7: MODE DESCRIPTIONS

Mode	Description
000	SPP mode
001	PS/2 Parallel Port mode
010	Parallel Port Data FIFO mode
011	ECP Parallel Port mode
100	EPP mode (If this option is enabled in the configuration registers)
101	Reserved
110	Test mode
111	Configuration mode
*Refer to ECR Register Description	

13.2.4 DATA AND ECPAFIFO PORT

ADDRESS OFFSET = 00H

Modes 000 and 001 (Data Port)

The Data Port is located at an offset of ‘00H’ from the base address. The data register is cleared at initialization by RESET. During a WRITE operation, the Data Register latches the contents of the data bus. The contents of this register are buffered (non inverting) and output onto the PD0 - PD7 ports. During a READ operation, PD0 - PD7 ports are read and output to the host CPU.

Mode 011 (ECP FIFO - Address/RLE)

A data byte written to this address is placed in the FIFO and tagged as an ECP Address/RLE. The hardware at the ECP port transmits this byte to the peripheral automatically. The operation of this register is only defined for the forward direction (direction is 0). Refer to TABLE 36-12: on page 349, located in Section 36.0, "Timing Diagrams," on page 339 of this data sheet.

13.2.5 DEVICE STATUS REGISTER (DSR)

ADDRESS OFFSET = 01H

The Status Port is located at an offset of '01H' from the base address. Bits0 - 2 are not implemented as register bits, during a read of the Printer Status Register these bits are a low level. The bits of the Status Port are defined as follows:

Bit 3 nFault

The level on the nFault input is read by the CPU as bit 3 of the Device Status Register.

Bit 4 Select

The level on the Select input is read by the CPU as bit 4 of the Device Status Register.

Bit 5 PError

The level on the PError input is read by the CPU as bit 5 of the Device Status Register. Printer Status Register.

Bit 6 ACK#

The level on the ACK# input is read by the CPU as bit 6 of the Device Status Register.

Bit 7 nBusy

The complement of the level on the BUSY input is read by the CPU as bit 7 of the Device Status Register.

13.2.6 DEVICE CONTROL REGISTER (DCR)

ADDRESS OFFSET = 02H

The Control Register is located at an offset of '02H' from the base address. The Control Register is only valid after activation and is initialized to zero by the RESET input, bits 0 to 5 only being affected; bits 6 and 7 are hard wired low.

Bit 0 STROBE - STROBE

This bit is inverted and output onto the STROBE# output.

Bit 1 AUTOFD - AUTOFEED

This bit is inverted and output onto the nAutoFd output. A logic 1 causes the printer to generate a line feed after each line is printed. A logic 0 means no autofeed.

Bit 2 INIT# - INITIATE OUTPUT

This bit is output onto the INIT# output without inversion.

Bit 3 SELECTIN

This bit is inverted and output onto the SLCTIN# output. A logic 1 on this bit selects the printer; a logic 0 means the printer is not selected.

Bit 4 ackIntEn - INTERRUPT REQUEST ENABLE

The interrupt request enable bit when set to a high level may be used to enable interrupt requests from the Parallel Port to the CPU due to a low to high transition on the ACK# input. Refer to the description of the interrupt under Operation, Interrupts.

Bit 5 DIRECTION

If mode=000 or mode=010, this bit has no effect and the direction is always out regardless of the state of this bit. In all other modes, Direction is valid and a logic 0 means that the printer port is in output mode (write); a logic 1 means that the printer port is in input mode (read).

Bits 6 and 7 during a read are a low level, and cannot be written.

cFifo (Parallel Port Data FIFO)

ADDRESS OFFSET = 400h

Mode = 010

Bytes written or DMAed from the system to this FIFO are transmitted by a hardware handshake to the peripheral using the standard parallel port protocol. Transfers to the FIFO are byte aligned. This mode is only defined for the forward direction.

ecpDFifo (ECP Data FIFO)

ADDRESS OFFSET = 400H

Mode = 011

Bytes written or DMAed from the system to this FIFO, when the direction bit is 0, are transmitted by a hardware handshake to the peripheral using the ECP parallel port protocol. Transfers to the FIFO are byte aligned.

Data bytes from the peripheral are read under automatic hardware handshake from ECP into this FIFO when the direction bit is 1. Reads or DMAs from the FIFO will return bytes of ECP data to the system.

tFifo (Test FIFO Mode)

ADDRESS OFFSET = 400H

Mode = 110

Data bytes may be read, written or DMAed to or from the system to this FIFO in any direction. Data in the tFIFO will not be transmitted to the to the parallel port lines using a hardware protocol handshake. However, data in the tFIFO may be displayed on the parallel port data lines.

The tFIFO will not stall when overwritten or underrun. If an attempt is made to write data to a full tFIFO, the new data is not accepted into the tFIFO. If an attempt is made to read data from an empty tFIFO, the last data byte is re-read again. The full and empty bits must always keep track of the correct FIFO state. The tFIFO will transfer data at the maximum ISA rate so that software may generate performance metrics.

The FIFO size and interrupt threshold can be determined by writing bytes to the FIFO and checking the full and serviceIntr bits.

The writeIntrThreshold can be determined by starting with a full tFIFO, setting the direction bit to 0 and emptying it a byte at a time until serviceIntr is set. This may generate a spurious interrupt, but will indicate that the threshold has been reached.

The readIntrThreshold can be determined by setting the direction bit to 1 and filling the empty tFIFO a byte at a time until serviceIntr is set. This may generate a spurious interrupt, but will indicate that the threshold has been reached.

Data bytes are always read from the head of tFIFO regardless of the value of the direction bit. For example if 44h, 33h, 22h is written to the FIFO, then reading the tFIFO will return 44h, 33h, 22h in the same order as was written.

cnfgA (Configuration Register A)

ADDRESS OFFSET = 400H

Mode = 111

This register is a read only register. When read, 10H is returned. This indicates to the system that this is an 8-bit implementation. (PWord = 1 byte)

cnfgB (Configuration Register B)

ADDRESS OFFSET = 401H

Mode = 111

Bit 7 compress

This bit is read only. During a read it is a low level. This means that this chip does not support hardware RLE compression. It does support hardware de-compression.

Bit 6 intrValue

Returns the value of the interrupt to determine possible conflicts.

Bit [5:3] Parallel Port IRQ (read-only)

to Table 13-9 on page 151.

Bits [2:0] Parallel Port DMA (read-only)

to Table 13-10 on page 151.

ecr (Extended Control Register)

ADDRESS OFFSET = 402H

Mode = all

This register controls the extended ECP parallel port functions.

Bits 7,6,5

These bits are Read/Write and select the Mode.

Bit 4 nErrIntrEn

Read/Write (Valid only in ECP Mode)

1: Disables the interrupt generated on the asserting edge of nFault.

0: Enables an interrupt pulse on the high to low edge of nFault. Note that an interrupt will be generated if nFault is asserted (interrupting) and this bit is written from a 1 to a 0. This prevents interrupts from being lost in the time between the read of the ecr and the write of the ecr.

Bit 3 dmaEn

Read/Write

1: Enables DMA (DMA starts when serviceIntr is 0).

0: Disables DMA unconditionally.

Bit 2 serviceIntr

Read/Write

1: Disables DMA and all of the service interrupts.

0: Enables one of the following 3 cases of interrupts. Once one of the 3 service interrupts has occurred serviceIntr bit shall be set to a 1 by hardware. It must be reset to 0 to re-enable the interrupts. Writing this bit to a 1 will not cause an interrupt.

case dmaEn=1:

During DMA (this bit is set to a 1 when terminal count is reached).

case dmaEn=0 direction=0:

This bit shall be set to 1 whenever there are writeIntrThreshold or more bytes free in the FIFO.

case dmaEn=0 direction=1:

This bit shall be set to 1 whenever there are readIntrThreshold or more valid bytes to be read from the FIFO.

Bit 1 full

Read only

1: The FIFO cannot accept another byte or the FIFO is completely full.

0: The FIFO has at least 1 free byte.

Bit 0 empty

Read only

1: The FIFO is completely empty.

0: The FIFO contains at least 1 byte of data.

TABLE 13-8: EXTENDED CONTROL REGISTER (A)

R/W	Mode
000:	Standard Parallel Port Mode. In this mode the FIFO is reset and common drain drivers are used on the control lines (STROBE#, nAutoFd, INIT# and nSelectIn). Setting the direction bit will not tri-state the output drivers in this mode.
001:	PS/2 Parallel Port Mode. Same as above except that direction may be used to tri-state the data lines and reading the data register returns the value on the data lines and not the value in the data register. All drivers have active pull-ups (push-pull).
010:	Parallel Port FIFO Mode. This is the same as 000 except that bytes are written or DMAed to the FIFO. FIFO data is automatically transmitted using the standard parallel port protocol. Note that this mode is only useful when direction is 0. All drivers have active pull-ups (push-pull).
011:	ECP Parallel Port Mode. In the forward direction (direction is 0) bytes placed into the ecpDFifo and bytes written to the ecpAFifo are placed in a single FIFO and transmitted automatically to the peripheral using ECP Protocol. In the reverse direction (direction is 1) bytes are moved from the ECP parallel port and packed into bytes in the ecpDFifo. All drivers have active pull-ups (push-pull).
100:	Selects EPP Mode: In this mode, EPP is selected if the EPP supported option is selected in configuration register L3-CRF0. All drivers have active pull-ups (push-pull).
101:	Reserved

TABLE 13-8: EXTENDED CONTROL REGISTER (A) (CONTINUED)

R/W	Mode
110:	Test Mode. In this mode the FIFO may be written and read, but the data will not be transmitted on the parallel port. All drivers have active pull-ups (push-pull).
111:	Configuration Mode. In this mode the confgA, confgB registers are accessible at 0x400 and 0x401. All drivers have active pull-ups (push-pull).

TABLE 13-9: EXTENDED CONTROL REGISTER (B)

IRQ Selected	Config Reg B Bits 5:3
15	110
14	101
11	100
10	011
9	010
7	001
5	111
All others	000

APPLICATION NOTE: The cnfgB register reads back the IRQ selected in the Interrupt Select configuration register (offset 70h). This configuration register does not affect the SERIRQ channel on which the Parallel Port interrupt appears. The interrupt channel is assigned to the Parallel Port in the LPC Logical Device as shown in Table 9-6, "SIRQ Interrupt Configuration Register Map". If this IRQ field in cnfgB is required, then software must insure that the Interrupt Select configuration register and the SIRQ Interrupt Configuration table are set to compatible values.

TABLE 13-10: EXTENDED CONTROL REGISTER (C)

IRQ Selected	Config Reg B Bits 5:3
3	011
2	010
1	001
All others	000

APPLICATION NOTE: The cnfgB register reads back the DMA channel selected in the DMA Channel Select configuration register (offset 74h). This configuration register does not affect the DMA channel on which the Parallel Port DMA transfer appears. The DMA channel is assigned to the Parallel Port in the LPC Logical Device as shown in Table 9-4, "DMA Configuration Register Map". If this DMA field in cnfgB is required, then software must insure that the DMA Channel Select configuration register and the DMA Configuration table are set to compatible values.

13.2.7 OPERATION

Mode Switching/Software Control

Software will execute P1284 negotiation and all operation prior to a data transfer phase under programmed I/O control (mode 000 or 001). Hardware provides an automatic control line handshake, moving data between the FIFO and the ECP port only in the data transfer phase (modes 011 or 010).

Setting the mode to 011 or 010 will cause the hardware to initiate data transfer.

If the port is in mode 000 or 001 it may switch to any other mode. If the port is not in mode 000 or 001 it can only be switched into mode 000 or 001. The direction can only be changed in mode 001.

Once in an extended forward mode the software should wait for the FIFO to be empty before switching back to mode 000 or 001. In this case all control signals will be deasserted before the mode switch. In an ecp reverse mode the software waits for all the data to be read from the FIFO before changing back to mode 000 or 001. Since the automatic hardware ecp reverse handshake only cares about the state of the FIFO it may have acquired extra data which will be discarded. It may in fact be in the middle of a transfer when the mode is changed back to 000 or 001. In this case the port will deassert nAutoFd independent of the state of the transfer. The design shall not cause glitches on the handshake signals if the software meets the constraints above.

13.2.7.1 ECP Operation

Prior to ECP operation the Host must negotiate on the parallel port to determine if the peripheral supports the ECP protocol. This is a somewhat complex negotiation carried out under program control in mode 000.

After negotiation, it is necessary to initialize some of the port bits. The following are required:

Set Direction = 0, enabling the drivers.

Set strobe = 0, causing the STROBE# signal to default to the deasserted state.

Set autoFd = 0, causing the nAutoFd signal to default to the deasserted state.

Set mode = 011 (ECP Mode)

ECP address/RLE bytes or data bytes may be sent automatically by writing the ecpAFifo or ecpDFifo respectively.

Note all FIFO data transfers are byte wide and byte aligned. Address/RLE transfers are byte-wide and only allowed in the forward direction.

The host may switch directions by first switching to mode = 001, negotiating for the forward or reverse channel, setting direction to 1 or 0, then setting mode = 011. When direction is 1 the hardware shall handshake for each ECP read data byte and attempt to fill the FIFO. Bytes may then be read from the ecpDFifo as long as it is not empty.

ECP transfers may also be accomplished (albeit slowly) by handshaking individual bytes under program control in mode = 001, or 000.

13.2.8 TERMINATION FROM ECP MODE

Termination from ECP Mode is similar to the termination from Nibble/Byte Modes. The host is permitted to terminate from ECP Mode only in specific well-defined states. The termination can only be executed while the bus is in the forward direction. To terminate while the channel is in the reverse direction, it must first be transitioned into the forward direction.

13.2.9 COMMAND/DATA

ECP Mode supports two advanced features to improve the effectiveness of the protocol for some applications. The features are implemented by allowing the transfer of normal 8 bit data or 8 bit commands.

When in the forward direction, normal data is transferred when HostAck is high and an 8 bit command is transferred when HostAck is low.

The most significant bit of the command indicates whether it is a run-length count (for compression) or a channel address.

When in the reverse direction, normal data is transferred when PeriphAck is high and an 8 bit command is transferred when PeriphAck is low. The most significant bit of the command is always zero. Reverse channel addresses are seldom used and may not be supported in hardware.

TABLE 13-11: CHANNEL/DATA COMMANDS SUPPORTED IN ECP MODE

Forward Channel Commands (HostAck Low) Reverse Channel Commands (PeripAck Low)	
D7	D[6:0]
0	Run-Length Count (0-127) (mode 0011 0X00 only)
1	Channel Address (0-127)

13.2.10 DATA COMPRESSION

The ECP port supports run length encoded (RLE) decompression in hardware and can transfer compressed data to a peripheral. Run length encoded (RLE) compression in hardware is not supported. To transfer compressed data in ECP mode, the compression count is written to the ecpAFifo and the data byte is written to the ecpDFifo.

Compression is accomplished by counting identical bytes and transmitting an RLE byte that indicates how many times the next byte is to be repeated. Decompression simply intercepts the RLE byte and repeats the following byte the specified number of times. When a run-length count is received from a peripheral, the subsequent data byte is replicated the specified number of times. A run-length count of zero specifies that only one byte of data is represented by the next data byte, whereas a run-length count of 127 indicates that the next byte should be expanded to 128 bytes. To prevent data expansion, however, run-length counts of zero should be avoided.

13.2.11 PIN DEFINITION

The drivers for STROBE#, nAutoFd, INIT# and nSelectIn are open-drain in mode 000 and are push-pull in all other modes.

13.2.12 LPC CONNECTIONS

The interface can never stall causing the host to hang. The width of data transfers is strictly controlled on an I/O address basis per this specification. All FIFO-DMA transfers are byte wide, byte aligned and end on a byte boundary. (The PWord value can be obtained by reading Configuration Register A, cnfgA, described in the next section). Single byte wide transfers are always possible with standard or PS/2 mode using program control of the control signals.

13.2.13 INTERRUPTS

The interrupts are enabled by serviceIntr in the ecr register.

serviceIntr = 1 Disables the DMA and all of the service interrupts.

serviceIntr = 0 Enables the selected interrupt condition. If the interrupting condition is valid, then the interrupts generated immediately when this bit is changed from a 1 to a 0. This can occur during Programmed I/O if the number of bytes removed or added from/to the FIFO does not cross the threshold.

An interrupt is generated when:

1. For DMA transfers: When serviceIntr is 0, dmaEn is 1 and the DMA TC cycle is received.
2. For Programmed I/O:
3. When serviceIntr is 0, dmaEn is 0, direction is 0 and there are writeIntrThreshold or more free bytes in the FIFO. Also, an interrupt is generated when serviceIntr is cleared to 0 whenever there are writeIntrThreshold or more free bytes in the FIFO.
4. When serviceIntr is 0, dmaEn is 0, direction is 1 and there are readIntrThreshold or more bytes in the FIFO. Also, an interrupt is generated when serviceIntr is cleared to 0 whenever there are readIntrThreshold or more bytes in the FIFO.
5. When nErrIntrEn is 0 and nFault transitions from high to low or when nErrIntrEn is set from 1 to 0 and nFault is asserted.
6. When ackIntEn is 1 and the ACK# signal transitions from a low to a high.

13.2.14 FIFO OPERATION

The FIFO threshold is set in the chip configuration registers. All data transfers to or from the parallel port can proceed in DMA or Programmed I/O (non-DMA) mode as indicated by the selected mode. The FIFO is used by selecting the Parallel Port FIFO mode or ECP Parallel Port Mode. (FIFO test mode will be addressed separately.) After a reset, the FIFO is disabled. Each data byte is transferred by a Programmed I/O cycle or DMA cycle depending on the selection of DMA or Programmed I/O mode.

The following paragraphs detail the operation of the FIFO flow control. In these descriptions, <threshold> ranges from 1 to 16. The parameter FIFOTHR, which the user programs, is one less and ranges from 0 to 15.

A low threshold value (i.e. 2) results in longer periods of time between service requests, but requires faster servicing of the request for both read and write cases. The host must be very responsive to the service request. This is the desired case for use with a “fast” system. A high value of threshold (i.e. 12) is used with a “sluggish” system by affording a long latency period after a service request, but results in more frequent service requests.

13.2.15 DMA TRANSFERS

DMA transfers are always to or from the ecpDFifo, tFifo or CFifo. DMA utilizes the standard PC DMA services. To use the DMA transfers, the host first sets up the direction and state as in the programmed I/O case. Then it programs the DMA controller in the host with the desired count and memory address. Lastly it sets dmaEn to 1 and serviceIntr to 0. The ECP requests DMA transfers from the host by encoding the LDRQ# pin. The DMA will empty or fill the FIFO using the appropriate direction and mode. When the terminal count in the DMA controller is reached, an interrupt is generated and serviceIntr is asserted, disabling DMA. In order to prevent possible blocking of refresh requests a DMA cycle shall not be requested for more than 32 DMA cycles in a row. The FIFO is enabled directly by the host initiating a DMA cycle for the requested channel, and addresses need not be valid. An interrupt is generated when a TC cycle is received. (Note: The only way to properly terminate DMA transfers is with a TC cycle.)

DMA may be disabled in the middle of a transfer by first disabling the host DMA controller. Then setting serviceIntr to 1, followed by setting dmaEn to 0, and waiting for the FIFO to become empty or full. Restarting the DMA is accomplished by enabling DMA in the host, setting dmaEn to 1, followed by setting serviceIntr to 0.

13.2.16 DMA MODE - TRANSFERS FROM THE FIFO TO THE HOST

In the reverse mode, the peripheral may not continue to fill the FIFO if it runs out of data to transfer, even if the chip continues to request more data from the peripheral.)

The ECP requests a DMA cycle whenever there is data in the FIFO. The DMA controller must respond to the request by reading data from the FIFO. The ECP stops requesting DMA cycles when the FIFO becomes empty or when a TC cycle is received, indicating that no more data is required. If the ECP stops requesting DMA cycles due to the FIFO going empty, then a DMA cycle is requested again as soon as there is one byte in the FIFO. If the ECP stops requesting DMA cycles due to the TC cycle, then a DMA cycle is requested again when there is one byte in the FIFO, and serviceIntr has been re-enabled.

13.2.17 PROGRAMMED I/O MODE OR NON-DMA MODE

The ECP or parallel port FIFOs may also be operated using interrupt driven programmed I/O. Software can determine the writeIntrThreshold, readIntrThreshold, and FIFO depth by accessing the FIFO in Test Mode.

Programmed I/O transfers are to the ecpDFifo at 400H and ecpAFifo at 000H or from the ecpDFifo located at 400H, or to/from the tFifo at 400H. To use the programmed I/O transfers, the host first sets up the direction and state, sets dmaEn to 0 and serviceIntr to 0.

The ECP requests programmed I/O transfers from the host by activating the interrupt. The programmed I/O will empty or fill the FIFO using the appropriate direction and mode.

Note 13-11 A threshold of 16 is equivalent to a threshold of 15. These two cases are treated the same.

13.2.18 PROGRAMMED I/O - TRANSFERS FROM THE FIFO TO THE HOST

In the reverse direction an interrupt occurs when serviceIntr is 0 and readIntrThreshold bytes are available in the FIFO. If at this time the FIFO is full it can be emptied completely in a single burst, otherwise readIntrThreshold bytes may be read from the FIFO in a single burst.

readIntrThreshold =(16-<threshold>) data bytes in FIFO

An interrupt is generated when serviceIntr is 0 and the number of bytes in the FIFO is greater than or equal to (16-<threshold>). (If the threshold = 12, then the interrupt is set whenever there are 4-16 bytes in the FIFO). The host must respond to the request by reading data from the FIFO. This process is repeated until the last byte is transferred out of the FIFO. If at this time the FIFO is full, it can be completely emptied in a single burst, otherwise a minimum of (16-<threshold>) bytes may be read from the FIFO in a single burst.

13.2.19 PROGRAMMED I/O - TRANSFERS FROM THE HOST TO THE FIFO

In the forward direction an interrupt occurs when serviceIntr is 0 and there are writeIntrThreshold or more bytes free in the FIFO. At this time if the FIFO is empty it can be filled with a single burst before the empty bit needs to be re-read. Otherwise it may be filled with writeIntrThreshold bytes.

writeIntrThreshold = (16-<threshold>) free bytes in FIFO

An interrupt is generated when serviceIntr is 0 and the number of bytes in the FIFO is less than or equal to <threshold>. (If the threshold = 12, then the interrupt is set whenever there are 12 or less bytes of data in the FIFO.) The host must respond to the request by writing data to the FIFO. If at this time the FIFO is empty, it can be completely filled in a single burst, otherwise a minimum of (16-<threshold>) bytes may be written to the FIFO in a single burst. This process is repeated until the last byte is transferred into the FIFO.

14.0 FLOPPY DISK CONTROLLER

The Floppy Disk controller (FDC) provides the interface between a host microprocessor and the floppy disk drives. The FDC integrates the functions of the Formatter/Controller, Digital data Separator, Write Precompensation and Data Rate Selection logic for an IBM XT/AT compatible FDC. The true CMOS 765B core maintains 100% IBM PC XT/AT compatibility in addition to providing data overflow and underflow protection. SCH5636 supports one floppy drive directly (see Section 14.12, "Floppy Drive Presence Detection," on page 190).

The FDC is compatible to the 82077AA using Microchip's proprietary floppy disk controller core.

Note:	Although the SCH5636 supports only a single floppy drive, references to the second drive are retained in this chapter for reference.

14.1 FDC Internal Registers

The Floppy Disk Controller contains eight internal registers which facilitate the interfacing between the host microprocessor and the disk drive. Table 14-1 shows the addresses required to access these registers. Registers other than the ones shown are not supported. The rest of the description assumes that the primary addresses have been selected.

(Shown with base addresses of 3F0 and 370.)

TABLE 14-1: STATUS, DATA AND CONTROL REGISTERS

Primary Address	Secondary Address	R/W	Register
3F0	370	R	Status Register A (SRA)
3F1	371	R	Status Register B (SRB)
3F2	372	R/W	Digital Output Register (DOR)
3F3	373	R/W	Tape Drive Register (TDR)
3F4	374	R	Main Status Register (MSR)
3F4	374	W	Data Rate Select Register (DSR)
3F5	375	R/W	Data (FIFO)
3F6	376		Reserved
3F7	377	R	Digital Input Register (DIR)
3F7	377	W	Configuration Control Register (CCR)

14.1.1 STATUS REGISTER A (SRA)

Address 3F0 READ ONLY

This register is read-only and monitors the state of the internal interrupt signal and several disk interface pins in PS/2 and Model 30 modes. The SRA can be accessed at any time when in PS/2 mode. In the PC/AT mode the data bus pins D0 – D7 are held in a high impedance state for a read of address 3F0.

14.1.1.1 PS/2 Mode

	7	6	5	4	3	2	1	0
	INT PENDING	nDRV2	STEP	TRK0#	HDSEL	nINDX	nWP	DIR
RESET COND.	0	1	0	N/A	0	N/A	N/A	0

Bit 0 DIRECTION

Active high status indicating the direction of head movement. A logic "1" indicates inward direction; a logic "0" indicates outward direction.

Bit 1 nWRITE PROTECT

Active low status of the WRITE PROTECT disk interface input. A logic "0" indicates that the disk is write protected.

Bit 2 INDEX#

Active low status of the INDEX disk interface input.

Bit 3 HEAD SELECT

Active high status of the HDSEL disk interface input. A logic "1" selects side 1 and a logic "0" selects side 0.

Bit 4 nTRACK 0

Active low status of the TRK0 disk interface input.

Bit 5 STEP

Active high status of the STEP output disk interface output pin.

Bit 6 nDRV2

Note:	This function is not supported. This bit is always read as “1”.

Bit 7 INTERRUPT PENDING

Active high bit indicating the state of the Floppy Disk Interrupt output.

14.1.1.2 PS/2 Model 30 Mode

	7	6	5	4	3	2	1	0
	INT PENDING	DRQ	STEP F/F	TRK0	HDSEL#HDSEL#HDSEL#	INDX	WP	DIR#
RESET COND.	0	0	0	N/A	1	N/A	N/A	1

Bit 0 DIRECTION

Active low status indicating the direction of head movement. A logic “0” indicates inward direction; a logic “1” indicates outward direction.

Bit 1 WRITE PROTECT

Active high status of the WRITE PROTECT disk interface input. A logic “1” indicates that the disk is write protected.

Bit 2 INDEX

Active high status of the INDEX disk interface input.

Bit 3 HEAD SELECT

Active low status of the HDSEL disk interface input. A logic “0” selects side 1 and a logic “1” selects side 0.

Bit 4 TRACK 0

Active high status of the TRK0 disk interface input.

Bit 5 STEP

Active high status of the latched STEP disk interface output pin. This bit is latched with the STEP output going active, and is cleared with a read from the DIR register, or with a hardware or software reset.

Bit 6 DMA REQUEST

Active high status of the DMA request pending.

Bit 7 INTERRUPT PENDING

Active high bit indicating the state of the Floppy Disk Interrupt.

14.1.2 STATUS REGISTER B (SRB)

Address 3F1 READ ONLY

This register is read-only and monitors the state of several disk interface pins in PS/2 and Model 30 modes. The SRB can be accessed at any time when in PS/2 mode. In the PC/AT mode the data bus pins D0 – D7 are held in a high impedance state for a read of address 3F1.

14.1.2.1 PS/2 Mode

	7	6	5	4	3	2	1	0
	1	1	DRIVE SEL0	WDATA TOGGLE	RDATA TOGGLE	WGATE	MOT EN1	MOT EN0
RESET COND.	1	1	0	0	0	0	0	0

Bit 0 MOTOR ENABLE 0

Active high status of the MTR0 disk interface output pin. This bit is low after a hardware reset and unaffected by a software reset.

Bit 1 MOTOR ENABLE 1

Active high status of the MTR1 disk interface output pin. This bit is low after a hardware reset and unaffected by a software reset. **Note:** This function is not supported.

Bit 2 WRITE GATE

Active high status of the WGATE disk interface output.

Bit 3 READ DATA TOGGLE

Every inactive edge of the RDATA input causes this bit to change state.

Bit 4 WRITE DATA TOGGLE

Every inactive edge of the WDATA input causes this bit to change state.

Bit 5 DRIVE SELECT 0

Reflects the status of the Drive Select 0 bit of the DOR (address 3F2 bit 0). This bit is cleared after a hardware reset and it is unaffected by a software reset.

Bit 6 RESERVED

Always read as a logic “1”.

Bit 7 RESERVED

Always read as a logic “1”.

14.1.2.2 PS/2 Model 30 Mode

	7	6	5	4	3	2	1	0
	nDRV2	DS1#	DS0#	WDATA F/F	RDATA F/F	WGATE F/F	nDS3	nDS2
RESET COND.	N/A	1	1	0	0	0	1	1

Bit 0 nDRIVE SELECT 2

The DS2 disk interface is not supported.

Bit 1 nDRIVE SELECT 3

The DS3 disk interface is not supported.

Bit 2 WRITE GATE

Active high status of the latched WGATE output signal. This bit is latched by the active going edge of WGATE and is cleared by the read of the DIR register.

Bit 3 READ DATA

Active high status of the latched RDATA output signal. This bit is latched by the inactive going edge of RDATA and is cleared by the read of the DIR register.

Bit 4 WRITE DATA

Active high status of the latched WDATA output signal. This bit is latched by the inactive going edge of WDATA and is cleared by the read of the DIR register. This bit is not gated with WGATE.

Bit 5 nDRIVE SELECT 0

Active low status of the DS0 disk interface output.

Bit 6 nDRIVE SELECT 1

Active low status of the DS1 disk interface output. **Note**: This function is not supported.

Bit 7 nDRV2

Active low status of the DRV2 disk interface input. **Note**: This function is not supported.

14.1.3 DIGITAL OUTPUT REGISTER (DOR)

Address 3F2 READ/WRITE

The DOR controls the drive select and motor enables of the disk interface outputs. It also contains the enable for the DMA logic and a software reset bit. The contents of the DOR are unaffected by a software reset. The DOR can be written to at any time.

	7	6	5	4	3	2	1	0
	MOT EN3	MOT EN2	MOT EN1	MOT EN0	DMAEN	nRESET	DRIVE SEL1	DRIVE SEL0
RESET COND.	0	0	0	0	0	0	0	0

Bit 0 and 1 DRIVE SELECT

These two bits are binary encoded for the drive selects, thereby allowing only one drive to be selected at one time. **Note**: Only one drive is supported.

Bit 2 nRESET

A logic “0” written to this bit resets the Floppy disk controller. This reset will remain active until a logic “1” is written to this bit. This software reset does not affect the DSR and CCR registers, nor does it affect the other bits of the DOR register. The minimum reset duration required is 100ns, toggling this bit by consecutive writes to this register is a valid method of issuing a software reset.

Bit 3 DMAEN

PC/AT and Model 30 Mode:

Writing this bit to logic “1” will enable the DMA and interrupt functions. This bit being a logic “0” will disable the DMA and interrupt functions. This bit is a logic “0” after a reset and in these modes.

PS/2 Mode: In this mode the DMA and interrupt functions are always enabled. During a reset, this bit will be cleared to a logic “0”.

Bit 4 MOTOR ENABLE 0

This bit controls the MTR0 disk interface output. A logic “1” in this bit will cause the output pin to go active.

Bit 5 MOTOR ENABLE 1

This bit controls the MTR1 disk interface output. A logic “1” in this bit will cause the output pin to go active. **Note**: MTR1 output is not supported.

Note: In the following tables, DS1# and MTR1# are not supported. They are included for reference to previous designs.

Drive	DOR Value
0	1CH
1	2DH

TABLE 14-2: INTERNAL 2 DRIVE DECODE – NORMAL

Digital Output Register				Drive Select Outputs (Active Low)		Motor on Outputs (Active Low)	
Bit 5	Bit 4	Bit1	Bit 0	DS1#	DS0#	MTR1#	MTR0#
X	1	0	0	1	0	nBIT 5	nBIT 4
1	X	0	1	0	1	nBIT 5	nBIT 4
0	0	X	X	1	1	nBIT 5	nBIT 4

TABLE 14-3: INTERNAL 2 DRIVE DECODE – DRIVES 0 AND 1 SWAPPED

Digital Output Register				Drive Select Outputs (Active Low)		Motor on Outputs (Active Low)	
Bit 5	Bit 4	Bit1	Bit 0	DS1#	DS0#	MTR1#	MTR0#
X	1	0	0	0	1	nBIT 4	nBIT 5
1	X	0	1	1	0	nBIT 4	nBIT 5
0	0	X	X	1	1	nBIT 4	nBIT 5

Bit 6 MOTOR ENABLE 2

The MTR2 disk interface output is not supported in the SCH5636.

Bit 7 MOTOR ENABLE 3

The MTR3 disk interface output is not supported in the SCH5636.

14.1.4 TAPE DRIVE REGISTER (TDR)

Address 3F3 READ/WRITE

The Tape Drive Register (TDR) is included for 82077 software compatibility and allows the user to assign tape support to a particular drive during initialization. Any future references to that drive automatically invokes tape support. The TDR Tape Select bits TDR.[1:0] determine the tape drive number. Table 14-4 illustrates the Tape Select Bit encoding. Note that drive 0 is the boot device and cannot be assigned tape support. The remaining Tape Drive Register bits TDR.[7:2] are tristated when read. The TDR is unaffected by a software reset.

TABLE 14-4: TAPE SELECT BITS

Tape SEL1 (TDR.1)	Tape SEL0 (TDR.0)	Drive Selected
0	0	None
0	1	1
1	0	2
1	1	3

14.1.4.1 Normal Floppy Mode

Normal mode. Register 3F3 contains only bits 0 and 1. When this register is read, bits 2 – 7 are ‘0’.

	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
REG 3F3	0	0	0	0	0	0	tape sel1	tape sel0

14.1.4.2 Enhanced Floppy Mode 2 (OS2)

Register 3F3 for Enhanced Floppy Mode 2 operation.

	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
REG 3F3	Reserved	Reserved	Drive Type ID		Floppy Boot Drive		tape sel1	tape sel0

Bit 3:2 Floppy Boot Drive

Read only. Always returns 0.

Bit 5:4 Drive Type ID

TABLE 14-5: DRIVE TYPE ID

Digital Output Register		Register 3F3 – Drive Type ID	
Bit 1	Bit 0	Bit 5	Bit 4
0	0	L0-CRF2 – B1	L0-CRF2 – B0
0	1	L0-CRF2 – B3	L0-CRF2 – B2
1	0	L0-CRF2 – B5	L0-CRF2 – B4
1	1	L0-CRF2 – B7	L0-CRF2 – B6

Note: L0-CRF2-Bx = Logical Device 0, Configuration Register F2, Bit x.

14.1.5 DATA RATE SELECT REGISTER (DSR)

Address 3F4 WRITE ONLY

This register is write only. It is used to program the data rate, amount of write precompensation, power down status, and software reset. The data rate is programmed using the Configuration Control Register (CCR) not the DSR, for PC/AT and PS/2 Model 30.

	7	6	5	4	3	2	1	0
	S/W RESET	POWER DOWN	0	PRE-COMP2	PRE-COMP1	PRE-COMP0	DRATE SEL1	DRATE SEL0
RESET COND.	0	0	0	0	0	0	1	0

This register is write only. It is used to program the data rate, amount of write precompensation, power down status, and software reset. The data rate is programmed using the Configuration Control Register (CCR) not the DSR, for PC/AT and PS/2 Model 30.

Other applications can set the data rate in the DSR. The data rate of the floppy controller is the most recent write of either the DSR or CCR. The DSR is unaffected by a software reset. A hardware reset will set the DSR to 02H, which corresponds to the default precompensation setting and 250 Kbps.

Bit 0 and 1 DATA RATE SELECT

These bits control the data rate of the floppy controller. See Table 14-7 for the settings corresponding to the individual data rates. The data rate select bits are unaffected by a software reset, and are set to 250 Kbps after a hardware reset.

Bit 2 through 4 PRECOMPENSATION SELECT

These three bits select the value of write precompensation that will be applied to the WDATA output signal. Table 14-6 shows the precompensation values for the combination of these bits settings. Track 0 is the default starting track number to start precompensation. This starting track number can be changed by the configure command.

TABLE 14-6: PRECOMPENSATION DELAYS

Precomp 432	Precompensation Delay (nsec)	
	<2Mbps	2Mbps
111 001 010 011 100 101 110 000	0.00 41.67 83.34 125.00 166.67 208.33 250.00 Default	0 20.8 41.7 62.5 83.3 104.2 125 Default
Default: See Table 14-9 on page 162.		

Bit 5 UNDEFINED

Should be written as a logic “0”.

Bit 6 LOW POWER

A logic “1” written to this bit will put the floppy controller into manual low power mode. The floppy controller clock and data separator circuits will be turned off. The controller will come out of manual low power mode after a software reset or access to the Data Register or Main Status Register.

Bit 7 SOFTWARE RESET

This active high bit has the same function as the DOR RESET (DOR bit 2) except that this bit is self clearing.

Note: The DSR is Shadowed in the Floppy Data Rate Select Shadow Register, located at the offset 0x1F in the runtime register block.

TABLE 14-7: DATA RATES

Drive Rate	Data Rate		Data Rate		DENSEL	DRATE(14-1)	
DRT0	SEL1	SEL0	MFM	FM		1	0
0	1	1	1Meg	---	1	1	1
0	0	0	500	250	1	0	0
0	0	1	300	150	0	0	1
0	1	0	250	125	0	1	0
1	1	1	1Meg	---	1	1	1
1	0	0	500	250	1	0	0
1	0	1	500	250	0	0	1
1	1	0	250	125	0	1	0

Drive Rate Table (Recommended) 00 = 360K, 1.2M, 720K, 1.44M and 2.88M Vertical Format

01 = 3-Mode Drive

10 = 2 Meg Tape

Note 14-1 The DRATE and DENSEL values are mapped onto the DRVDEN pin.

TABLE 14-8: DRVDEN MAPPING

DT1	DT0	DRVDEN0 (1)	Drive Type
0	0	DENSEL	4/2/1 MB 3.5” 2/1 MB 5.25” FDDS 2/1.6/1 MB 3.5” (3-MODE)
1	0	DRATE1	
0	1	nDENSEL	PS/2
1	1	DRATE0	

TABLE 14-9: DEFAULT PRECOMPENSATION DELAYS

Data Rate	Precompensation Delay
2 Mbps	20.8 ns
1 Mbps	41.67 ns
500 Kbps	125 ns
300 Kbps	125 ns
250 Kbps	125 ns

14.1.6 MAIN STATUS REGISTER

Address 3F4 READ ONLY

The Main Status Register is a read-only register and indicates the status of the disk controller. The Main Status Register can be read at any time. The MSR indicates when the disk controller is ready to receive data via the Data Register. It should be read before each byte transferring to or from the data register except in DMA mode. No delay is required when reading the MSR after a data transfer.

7	6	5	4	3	2	1	0
RQM	DIO	NON DMA	CMD BUSY	Reserved	Reserved	DRV1 BUSY	DRV0 BUSY

Bit 0 – 1 DRV x BUSY

These bits are set to 1s when a drive is in the seek portion of a command, including implied and overlapped seeks and recalibrates.

Bit 4 COMMAND BUSY

This bit is set to a 1 when a command is in progress. This bit will go active after the command byte has been accepted and goes inactive at the end of the results phase. If there is no result phase (Seek, Recalibrate commands), this bit is returned to a 0 after the last command byte.

Bit 5 NON-DMA

Reserved, read ‘0’. This part does not support non-DMA mode.

Bit 6 DIO

Indicates the direction of a data transfer once a RQM is set. A 1 indicates a read and a 0 indicates a write is required.

Bit 7 RQM

Indicates that the host can transfer data if set to a 1. No access is permitted if set to a 0.

14.1.7 DATA REGISTER (FIFO)

Address 3F5 READ/WRITE

All command parameter information, disk data and result status are transferred between the host processor and the floppy disk controller through the Data Register.

Data transfers are governed by the RQM and DIO bits in the Main Status Register.

The Data Register defaults to FIFO disabled mode after any form of reset. This maintains PC/AT hardware compatibility. The default values can be changed through the Configure command (enable full FIFO operation with threshold control). The advantage of the FIFO is that it allows the system a larger DMA latency without causing a disk error. Table 14-10 gives several examples of the delays with a FIFO.

The data is based upon the following formula:

$$\text{Threshold \# x} \quad \left| \; 1/(\text{DATA RATE}) \quad \text{x } 8 \; \right| \quad \text{- } 1.5\ \mu\text{s} = \text{DELAY}$$

At the start of a command, the FIFO action is always disabled and command parameters must be sent based upon the RQM and DIO bit settings. As the command execution phase is entered, the FIFO is cleared of any data to ensure that invalid data is not transferred.

An overrun or underrun will terminate the current command and the transfer of data. Disk writes will complete the current sector by generating a 00 pattern and valid CRC. Reads require the host to remove the remaining data so that the result phase may be entered.

TABLE 14-10: FIFO SERVICE DELAY

FIFO THRESHOLD EXAMPLES	MAXIMUM DELAY TO SERVICING AT 2 MBPS DATA RATE
1 byte 2 bytes 8 bytes 15 bytes	1 x 4 µs - 1.5 µs = 2.5 µs 2 x 4 µs - 1.5 µs = 6.5 µs 8 x 4 µs - 1.5 µs = 30.5 µs 15 x 4 µs - 1.5 µs = 58.5 µs
FIFO THRESHOLD EXAMPLES	**MAXIMUM DELAY TO SERVICING AT 1 MBPS DATA RATE**
1 byte 2 bytes 8 bytes 15 bytes	1 x 8 µs - 1.5 µs = 6.5 µs 2 x 8 µs - 1.5 µs = 14.5 µs 8 x 8 µs - 1.5 µs = 62.5 µs 15 x 8 µs - 1.5 µs = 118.5 µs
FIFO THRESHOLD EXAMPLES	**MAXIMUM DELAY TO SERVICING AT 500 KBPS DATA RATE**
1 byte 2 bytes 8 bytes 15 bytes	1 x 16 µs - 1.5 µs = 14.5 µs 2 x 16 µs - 1.5 µs = 30.5 µs 8 x 16 µs - 1.5 µs = 126.5 µs 15 x 16 µs - 1.5 µs = 238.5 µs

14.1.8 DIGITAL INPUT REGISTER (DIR)

Address 3F7 READ ONLY

This register is read-only in all modes.

14.1.8.1 PC-AT Mode

	7	6	5	4	3	2	1	0
	DSK CHG	0	0	0	0	0	0	0
RESET COND.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Bit 0 – 6 UNDEFINED

The data bus outputs D0 – 6 are read as ‘0’.

Bit 7 DSKCHG

This bit monitors the pin of the same name and reflects the opposite value seen on the disk cable or the value programmed in the Force Disk Change Register (see the Runtime Register at offset 0x1E).

14.1.8.2 PS/2 Mode

	7	6	5	4	3	2	1	0
	DSK CHG	1	1	1	1	DRATE SEL1	DRATE SEL0	nHIGH DENS
RESET COND.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1

Bit 0 nHIGH DENS

This bit is low whenever the 500 Kbps or 1 Mbps data rates are selected, and high when 250 Kbps and 300 Kbps are selected.

Bits 1 – 2 DATA RATE SELECT

These bits control the data rate of the floppy controller. See Table 14-7 on page 161 for the settings corresponding to the individual data rates. The data rate select bits are unaffected by a software reset, and are set to 250 Kbps after a hardware reset.

Bits 3 – 6 UNDEFINED

Always read as a logic “1”

Bit 7 DSKCHG

This bit monitors the pin of the same name and reflects the opposite value seen on the disk cable or the value programmed in the Force Disk Change Register (see Runtime Register at offset 0x1E).

14.1.8.3 Model 30 Mode

	7	6	5	4	3	2	1	0
	DSK CHG	0	0	0	DMAEN	NOPRE C	DRATE SEL1	DRATE SEL0
RESET COND.	N/A	0	0	0	0	0	1	0

Bits 0 – 1 DATA RATE SELECT

These bits control the data rate of the floppy controller. See Table 14-7 for the settings corresponding to the individual data rates. The data rate select bits are unaffected by a software reset, and are set to 250 Kbps after a hardware reset.

Bit 2 NOPREC

This bit reflects the value of NOPREC bit set in the CCR register.

Bit 3 DMAEN

This bit reflects the value of DMAEN bit set in the DOR register bit 3.

Bits 4 – 6 UNDEFINED

Always read as a logic “0”

Bit 7 DSKCHG

This bit monitors the pin of the same name and reflects the opposite value seen on the disk cable or the value programmed in the Force Disk Change Register (see Runtime Register at offset 0x1E).

14.1.9 CONFIGURATION CONTROL REGISTER (CCR)

Address 3F7 WRITE ONLY

14.1.9.1 PC/AT and PS/2 Modes

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	DRATE SEL1	DRATE SEL0
RESET COND.	N/A	N/A	N/A	N/A	N/A	N/A	1	0

Bit 0 and 1 DATA RATE SELECT 0 and 1

These bits determine the data rate of the floppy controller. See Table 14-7 on page 161 for the appropriate values.

Bit 2 – 7 RESERVED

Should be set to a logical “0”

14.1.9.2 PS/2 Model 30 Mode

	7	6	5	4	3	2	1	0
	0	0	0	0	0	NOPREC	DRATE SEL1	DRATE SEL0
RESET COND.	N/A	N/A	N/A	N/A	N/A	N/A	1	0

Bit 0 and 1 DATA RATE SELECT 0 and 1

These bits determine the data rate of the floppy controller. See Table 14-7 on page 161 for the appropriate values.

Bit 2 NO PRECOMPENSATION

This bit can be set by software, but it has no functionality. It can be read by bit 2 of the DSR when in Model 30 register mode. Unaffected by software reset.

Bit 3 – 7 RESERVED

Should be set to a logical “0”

Table 14-8 on page 161 shows the state of the DENSEL pin. The DENSEL pin is set high after a hardware reset and is unaffected by the DOR and the DSR resets.

14.2 Status Register Encoding

During the Result Phase of certain commands, the Data Register contains data bytes that give the status of the command just executed.

TABLE 14-11: STATUS REGISTER 0

Bit No.	Symbol	Name	Description
7,6	IC	Interrupt Code	00 - Normal termination of command. The specified command was properly executed and completed without error. 01 - Abnormal termination of command. Command execution was started, but was not successfully completed. 10 - Invalid command. The requested command could not be executed. 11 - Abnormal termination caused by Polling.
5	SE	Seek End	The FDC completed a Seek, Relative Seek or Recalibrate command (used during a Sense Interrupt Command).
4	EC	Equipment Check	The TRK0 pin failed to become a “1” after: 1. 80 step pulses in the Recalibrate command. 2. The Relative Seek command caused the FDC to step outward beyond Track 0.
3			Unused. This bit is always “0”.
2	H	Head Address	The current head address.
1,0	DS1,0	Drive Select	The current selected drive.

TABLE 14-12: STATUS REGISTER 1

Bit No.	Symbol	Name	Description
7	EN	End of Cylinder	The FDC tried to access a sector beyond the final sector of the track (255D). Will be set if TC is not issued after Read or Write Data command.
6			Unused. This bit is always “0”.
5	DE	Data Error	The FDC detected a CRC error in either the ID field or the data field of a sector.
4	OR	Overrun/ Underrun	Becomes set if the FDC does not receive CPU or DMA service within the required time interval, resulting in data overrun or underrun.
3			Unused. This bit is always “0”.
2	ND	No Data	Any one of the following: 1. Read Data, Read Deleted Data command - the FDC did not find the specified sector. 2. Read ID command - the FDC cannot read the ID field without an error. 3. Read A Track command - the FDC cannot find the proper sector sequence.
1	NW	Not Writable	WP pin became a “1” while the FDC is executing a Write Data, Write Deleted Data, or Format A Track command.

TABLE 14-12: STATUS REGISTER 1 (CONTINUED)

Bit No.	Symbol	Name	Description
0	MA	Missing Address Mark	Any one of the following: 1. The FDC did not detect an ID address mark at the specified track after encountering the index pulse from the INDEX# pin twice. 2. The FDC cannot detect a data address mark or a deleted data address mark on the specified track.

TABLE 14-13: STATUS REGISTER 2

Bit No.	Symbol	Name	Description
7			Unused. This bit is always “0”.
6	CM	Control Mark	Any one of the following: Read Data command - the FDC encountered a deleted data address mark. Read Deleted Data command - the FDC encountered a data address mark.
5	DD	Data Error in Data Field	The FDC detected a CRC error in the data field.
4	WC	Wrong Cylinder	The track address from the sector ID field is different from the track address maintained inside the FDC.
3			Unused. This bit is always “0”.
2			Unused. This bit is always “0”.
1	BC	Bad Cylinder	The track address from the sector ID field is different from the track address maintained inside the FDC and is equal to FF hex, which indicates a bad track with a hard error according to the IBM soft-sectored format.
0	MD	Missing Data Address Mark	The FDC cannot detect a data address mark or a deleted data address mark.

TABLE 14-14: STATUS REGISTER 3

Bit No.	Symbol	Name	Description
7			Unused. This bit is always “0”.
6	WP	Write Protected	Indicates the status of the WRTPRT pin.
5			Unused. This bit is always “1”.
4	T0	Track 0	Indicates the status of the TRK0 pin.
3			Unused. This bit is always “1”.
2	HD	Head Address	Indicates the status of the HDSEL pin.
1,0	DS1,0	Drive Select	Indicates the status of the DS1, DS0 pins. **Note:** DS1 is not supported.

14.3 Reset

There are three sources of system reset on the FDC: the PCI RESET# pin, a reset generated via a bit in the DOR, and a reset generated via a bit in the DSR. At power on, a Power On Reset initializes the FDC. All resets take the FDC out of the power down state.

All operations are terminated upon a PCI RESET#, and the FDC enters an idle state. A reset while a disk write is in progress will corrupt the data and CRC.

On exiting the reset state, various internal registers are cleared, including the Configure command information, and the FDC waits for a new command. Drive polling will start unless disabled by a new Configure command.

PCI RESET# Pin (Hardware Reset)

The PCI RESET# pin is a global reset and clears all registers except those programmed by the Specify command. The DOR reset bit is enabled and must be cleared by the host to exit the reset state.

DOR Reset vs. DSR Reset (Software Reset)

These two resets are functionally the same. Both will reset the FDC core, which affects drive status information and the FIFO circuits. The DSR reset clears itself automatically while the DOR reset requires the host to manually clear it. DOR reset has precedence over the DSR reset. The DOR reset is set automatically upon a pin reset. The user must manually clear this reset bit in the DOR to exit the reset state.

14.4 Modes of Operation

The FDC has three modes of operation, PC/AT mode, PS/2 mode and Model 30 mode. These are determined by the state of the Interface Mode bits in LD0-CRF0[3,2].

PC/AT Mode

The PC/AT register set is enabled, the DMA enable bit of the DOR becomes valid (controls the interrupt and DMA functions), and DENSEL is an active high signal.

PS/2 Mode

This mode supports the PS/2 models 50/60/80 configuration and register set. The DMA bit of the DOR becomes a “don’t care”. The DMA and interrupt functions are always enabled, and DENSEL is active low.

Model 30 mode

This mode supports PS/2 Model 30 configuration and register set. The DMA enable bit of the DOR becomes valid (controls the interrupt and DMA functions), and DENSEL is active low.

14.5 DMA Transfers

DMA transfers are enabled with the Specify command and are initiated by the FDC by activating a DMA request cycle. DMA read, write and verify cycles are supported. The FDC supports two DMA transfer modes: Single Transfer and Burst Transfer. Burst mode is enabled via Logical Device 0-CRF0-Bit[1] (LD0-CRF0[1]).

14.6 Controller Phases

For simplicity, command handling in the FDC can be divided into three phases: Command, Execution, and Result. Each phase is described in the following sections.

14.6.1 COMMAND PHASE

After a reset, the FDC enters the command phase and is ready to accept a command from the host. For each of the commands, a defined set of command code bytes and parameter bytes has to be written to the FDC before the command phase is complete. (Please refer to Table 14-15 on page 168 for the command set descriptions). These bytes of data must be transferred in the order prescribed.

Before writing to the FDC, the host must examine the RQM and DIO bits of the Main Status Register. RQM and DIO must be equal to “1” and “0” respectively before command bytes may be written. RQM is set false by the FDC after each write cycle until the received byte is processed. The FDC asserts RQM again to request each parameter byte of the command unless an illegal command condition is detected. After the last parameter byte is received, RQM remains “0” and the FDC automatically enters the next phase as defined by the command definition.

The FIFO is disabled during the command phase to provide for the proper handling of the “Invalid Command” condition.

14.6.2 EXECUTION PHASE

All data transfers to or from the FDC occur during the execution phase, which can proceed in DMA mode as indicated in the Specify command.

After a reset, the FIFO is disabled. Each data byte is transferred by a read/write or DMA cycle depending on the DMA mode. The Configure command can enable the FIFO and set the FIFO threshold value.

The following paragraphs detail the operation of the FIFO flow control. In these descriptions, <threshold> is defined as the number of bytes available to the FDC when service is requested from the host and ranges from 1 to 16. The parameter FIFOTHR, which the user programs, is one less and ranges from 0 to 15.

A low threshold value (i.e. 2) results in longer periods of time between service requests, but requires faster servicing of the request for both read and write cases. The host reads (writes) from (to) the FIFO until empty (full), then the transfer request goes inactive. The host must be very responsive to the service request. This is the desired case for use with a “fast” system.

A high value of threshold (i.e. 12) is used with a “sluggish” system by affording a long latency period after a service request, but results in more frequent service requests.

Non-DMA Mode – Transfers from the FIFO to the Host

This part does not support non-DMA mode.

Non-DMA Mode – Transfers from the Host to the FIFO

This part does not support non-DMA mode.

DMA Mode – Transfers from the FIFO to the Host

The FDC generates a DMA request cycle when the FIFO contains (16 - <threshold>) bytes, or the last byte of a full sector transfer has been placed in the FIFO. The DMA controller must respond to the request by reading data from the FIFO. The FDC will deactivate the DMA request when the FIFO becomes empty by generating the proper sync for the data transfer.

DMA Mode – Transfers from the Host to the FIFO.

The FDC generates a DMA request cycle when entering the execution phase of the data transfer commands. The DMA controller must respond by placing data in the FIFO. The DMA request remains active until the FIFO becomes full. The DMA request cycle is reasserted when the FIFO has <threshold> bytes remaining in the FIFO. The FDC will terminate the DMA cycle after a TC, indicating that no more data is required.

Data Transfer Termination

The FDC supports terminal count explicitly through the TC pin and implicitly through the underrun/overrun and end-of-track (EOT) functions. For full sector transfers, the EOT parameter can define the last sector to be transferred in a single or multi-sector transfer.

If the last sector to be transferred is a partial sector, the host can stop transferring the data in mid-sector, and the FDC will continue to complete the sector as if a TC cycle was received. The only difference between these implicit functions and TC cycle is that they return “abnormal termination” result status. Such status indications can be ignored if they were expected.

Note that when the host is sending data to the FIFO of the FDC, the internal sector count will be complete when the FDC reads the last byte from its side of the FIFO. There may be a delay in the removal of the transfer request signal of up to the time taken for the FDC to read the last 16 bytes from the FIFO. The host must tolerate this delay.

14.6.3 RESULT PHASE

The generation of the interrupt determines the beginning of the result phase. For each of the commands, a defined set of result bytes has to be read from the FDC before the result phase is complete. These bytes of data must be read out for another command to start.

RQM and DIO must both equal “1” before the result bytes may be read. After all the result bytes have been read, the RQM and DIO bits switch to “1” and “0” respectively, and the CB bit is cleared, indicating that the FDC is ready to accept the next command.

14.7 Command Set/Descriptions

Commands can be written whenever the FDC is in the command phase. Each command has a unique set of needed parameters and status results. The FDC checks to see that the first byte is a valid command and, if valid, proceeds with the command. If it is invalid, an interrupt is issued. The user sends a Sense Interrupt Status command which returns an invalid command error. Refer to Table 14-15 for explanations of the various symbols used. Table 14-16 lists the required parameters and the results associated with each command that the FDC is capable of performing.

TABLE 14-15: DESCRIPTION OF COMMAND SYMBOLS

Symbol	Name	Description
C	Cylinder Address	The currently selected address; 0 to 255.
D	Data Pattern	The pattern to be written in each sector data field during formatting.
D0, D1	Drive Select 0-1	Designates which drives are perpendicular drives on the Perpendicular Mode Command. A “1” indicates a perpendicular drive.
DIR	Direction Control	If this bit is 0, then the head will step out from the spindle during a relative seek. If set to a 1, the head will step in toward the spindle.

TABLE 14-15: DESCRIPTION OF COMMAND SYMBOLS (CONTINUED)

Symbol	Name	Description
DS0, DS1	Disk Drive Select	DS1 DS0 DRIVE 0 0 Drive 0 0 1 Drive 1 **Note:** Drive 1 is not supported
DTL	Special Sector Size	By setting N to zero (00), DTL may be used to control the number of bytes transferred in disk read/write commands. The sector size (N = 0) is set to 128. If the actual sector (on the diskette) is larger than DTL, the remainder of the actual sector is read but is not passed to the host during read commands; during write commands, the remainder of the actual sector is written with all zero bytes. The CRC check code is calculated with the actual sector. When N is not zero, DTL has no meaning and should be set to FF HEX.
EC	Enable Count	When this bit is “1” the “DTL” parameter of the Verify command becomes SC (number of sectors per track).
EFIFO	Enable FIFO	This active low bit when a 0, enables the FIFO. A “1” disables the FIFO (default).
EIS	Enable Implied Seek	When set, a seek operation will be performed before executing any read or write command that requires the C parameter in the command phase. A “0” disables the implied seek.
EOT	End of Track	The final sector number of the current track.
GAP		Alters Gap 2 length when using Perpendicular Mode.
GPL	Gap Length	The Gap 3 size. (Gap 3 is the space between sectors excluding the VCO synchronization field).
H/HDS	Head Address	Selected head: 0 or 1 (disk side 0 or 1) as encoded in the sector ID field.
HLT	Head Load Time	The time interval that FDC waits after loading the head and before initializing a read or write operation. Refer to the Specify command for actual delays.
HUT	Head Unload Time	The time interval from the end of the execution phase (of a read or write command) until the head is unloaded. Refer to the Specify command for actual delays.
LOCK		Lock defines whether EFIFO, FIFOTHR, and PRETRK parameters of the CONFIGURE COMMAND can be reset to their default values by a “software Reset”. (A reset caused by writing to the appropriate bits of either the DSR or DOR)
MFM	MFM/FM Mode Selector	A one selects the double density (MFM) mode. A zero selects single density (FM) mode.
MT	Multi-Track Selector	When set, this flag selects the multi-track operating mode. In this mode, the FDC treats a complete cylinder under head 0 and 1 as a single track. The FDC operates as this expanded track started at the first sector under head 0 and ended at the last sector under head 1. With this flag set, a multitrack read or write operation will automatically continue to the first sector under head 1 when the FDC finishes operating on the last sector under head 0.
N	Sector Size Code	This specifies the number of bytes in a sector. If this parameter is “00”, then the sector size is 128 bytes. The number of bytes transferred is determined by the DTL parameter. Otherwise the sector size is (2 raised to the “Nth” power) times 128. All values up to “07” hex are allowable. “07”h would equal a sector size of 16k. It is the user's responsibility to not select combinations that are not possible with the drive. N SECTOR SIZE 00 128 Bytes 01 256 Bytes 02 512 Bytes 03 1024 Bytes … … 07 16K Bytes
NCN	New Cylinder Number	The desired cylinder number.
ND	Non-DMA Mode Flag	Write ‘0’. This part does not support non-DMA mode.

TABLE 14-15: DESCRIPTION OF COMMAND SYMBOLS (CONTINUED)

Symbol	Name	Description
OW	Overwrite	The bits D0-D3 of the Perpendicular Mode Command can only be modified if OW is set to 1. OW id defined in the Lock command.
PCN	Present Cylinder Number	The current position of the head at the completion of Sense Interrupt Status command.
POLL	Polling Disable	When set, the internal polling routine is disabled. When clear, polling is enabled.
PRETRK	Precompensation Start Track Number	Programmable from track 00 to FFH.
R	Sector Address	The sector number to be read or written. In multi-sector transfers, this parameter specifies the sector number of the first sector to be read or written.
RCN	Relative Cylinder Number	Relative cylinder offset from present cylinder as used by the Relative Seek command.
SC	Number of Sectors Per Track	The number of sectors per track to be initialized by the Format command. The number of sectors per track to be verified during a Verify command when EC is set.
SK	Skip Flag	When set to 1, sectors containing a deleted data address mark will automatically be skipped during the execution of Read Data. If Read Deleted is executed, only sectors with a deleted address mark will be accessed. When set to “0”, the sector is read or written the same as the read and write commands.
SRT	Step Rate Interval	The time interval between step pulses issued by the FDC. Programmable from 0.5 to 8 milliseconds in increments of 0.5 ms at the 1 Mbit data rate. Refer to the SPECIFY command for actual delays.
ST0 ST1 ST2 ST3	Status 0 Status 1 Status 2 Status 3	Registers within the FDC which store status information after a command has been executed. This status information is available to the host during the result phase after command execution.
WGATE	Write Gate	Alters timing of WE to allow for pre-erase loads in perpendicular drives.

14.8 Instruction Set

TABLE 14-16: INSTRUCTION SET

READ DATA									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	MT	MFM	SK	0	0	1	1	0	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	C								Sector ID information prior to Command execution.
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data transfer between the FDD and system.
Result	R	ST0								Status information after Command execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID information after Command execution.
	R	H								
	R	R								
	R	N								

WRITE DATA									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	MT	MFM	0	0	0	1	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	C								Sector ID information prior to Command execution.
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data transfer between the FDD and system.
Result	R	ST0								Status information after Command execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID information after Command execution.
	R	H								
	R	R								
	R	N								

WRITE DELETED DATA									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	MT	MFM	0	0	1	0	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	C								Sector ID information prior to Command execution.
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data transfer between the FDD and system.
Result	R	ST0								Status information after Command execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID information after Command execution.
	R	H								
	R	R								
	R	N								

READ A TRACK									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	MFM	0	0	0	0	1	0	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	C								Sector ID information prior to Command execution.
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data transfer between the FDD and system. FDC reads all of cylinders' contents from index hole to EOT.
Result	R	ST0								Status information after Command execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID information after Command execution.
	R	H								
	R	R								
	R	N								

VERIFY									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	MT	MFM	SK	1	0	1	1	0	Command Codes
	W	EC	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	C								Sector ID information prior to Command execution.
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL/SC								
Execution										No data transfer takes place.
Result	R	ST0								Status information after Command execution.
	R	ST1								
	R	ST2								
	R	C								Sector ID information after Command execution.
	R	H								
	R	R								
	R	N								

VERSION									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	0	0	1	0	0	0	0	Command Code
Result	R	1	0	0	1	0	0	0	0	Enhanced Controller

FORMAT A TRACK									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	MFM	0	0	1	1	0	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	N								Bytes/Sector
	W	SC								Sectors/Cylinder
	W	GPL								Gap 3
	W	D								Filler Byte
Execution for Each Sector Repeat:	W	C								Input Sector Parameters
	W	H								
	W	R								
	W	N								
										FDC formats an entire cylinder
Result	R	ST0								Status information after Command execution
	R	ST1								
	R	ST2								
	R	Undefined								
	R	Undefined								
	R	Undefined								
	R	Undefined								

RECALIBRATE									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	0	0	0	0	0	0	DS1	DS0	Drive 1 not supported
Execution										Head retracted to Track 0 Interrupt.

SENSE INTERRUPT STATUS									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	0	0	0	1	0	0	0	Command Codes
Result	R	ST0								Status information at the end of each seek operation.
	R	PCN								

SPECIFY									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Command	W	0	0	0	0	0	0	1	1	Command Codes
	W	SRT				HUT				
	W	HLT							ND	

SENSE DRIVE STATUS										
		DATA BUS								
PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Command	W	0	0	0	0	0	1	0	0	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
Result	R	ST3								Status information about FDD

SEEK									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	0	0	0	0	1	1	1	1	Command Codes
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	NCN								
Execution										Head positioned over proper cylinder on diskette.

CONFIGURE									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	0	0	0	1	0	0	1	1	Configure Information
	W	0	0	0	0	0	0	0	0	
	W	0	EIS	EFIFO	POLL	FIFOTHR				
Execution	W	PRETRK								

RELATIVE SEEK									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	1	DIR	0	0	1	1	1	1	
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
	W	RCN								

DUMPREG									
PHASE	**R/W**	**DATA BUS**							**REMARKS**
		D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	
Command	W	0	0	0	0	1	1	1	0	**Note:** Registers placed in FIFO
Execution										
Result	R	PCN-Drive 0								
	R	PCN-Drive 1								
	R	PCN-Drive 2								
	R	PCN-Drive 3								
	R	SRT				HUT				
	R	HLT							ND	
	R	SC/EOT								
	R	LOCK	0	D3	D2	D1	D0	GAP	WGATE	
	R	0	EIS	EFIFO	POLL	FIFOTHR				
	R	PRETRK								

READ ID									
		DATA BUS							
PHASE	**R/W**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**REMARKS**
Command	W	0	MFM	0	0	1	0	1	0	Commands
	W	0	0	0	0	0	HDS	DS1	DS0	Drive 1 not supported
Execution										The first correct ID information on the Cylinder is stored in Data Register
Result	R	ST0								Status information after Command execution. Disk status after the Command has completed.
	R	ST1								
	R	ST2								
	R	C								
	R	H								
	R	R								
	R	N								

PERPENDICULAR MODE										
		DATA BUS								
PHASE	**R/W**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**REMARKS**
Command	W	0	0	0	1	0	0	1	0	Command Codes
		OW	0	D3	D2	D1	D0	GAP	WGATE	

INVALID CODES										
		DATA BUS								
PHASE	**R/W**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**REMARKS**
Command	W	Invalid Codes								Invalid Command Codes (NoOp – FDC goes into Standby State)
Result	R	ST0								ST0 = 80H

LOCK										
		DATA BUS								
PHASE	R/W	D7	D6	D5	D4	D3	D2	D1	D0	REMARKS
Command	W	LOCK	0	0	1	0	1	0	0	Command Codes
Result	R	0	0	0	LOCK	0	0	0	0	

SC is returned if the last command that was issued was the Format command. EOT is returned if the last command was a Read or Write.

Note 14-2 These bits are used internally only. They are not reflected in the Drive Select pins. It is the user's responsibility to maintain correspondence between these bits and the Drive Select pins (DOR).

14.9 Data Transfer Commands

All of the Read Data, Write Data and Verify type commands use the same parameter bytes and return the same results information, the only difference being the coding of bits 0-4 in the first byte.

An implied seek will be executed if the feature was enabled by the Configure command. This seek is completely transparent to the user. The Drive Busy bit for the drive will go active in the Main Status Register during the seek portion of the command. If the seek portion fails, it is reflected in the results status normally returned for a Read/Write Data command. Status Register 0 (ST0) would contain the error code and C would contain the cylinder on which the seek failed.

14.9.1 READ DATA

A set of nine (9) bytes is required to place the FDC in the Read Data Mode. After the Read Data command has been issued, the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify command), and begins reading ID Address Marks and ID fields. When the sector address read off the diskette matches with the sector address specified in the command, the FDC reads the sector's data field and transfers the data to the FIFO.

After completion of the read operation from the current sector, the sector address is incremented by one and the data from the next logical sector is read and output via the FIFO. This continuous read function is called "Multi-Sector Read Operation". Upon receipt of the TC cycle, or an implied TC (FIFO overrun/underrun), the FDC stops sending data but will continue to read data from the current sector, check the CRC bytes, and at the end of the sector, terminate the Read Data Command.

N determines the number of bytes per sector (see Table 14-17). If N is set to zero, the sector size is set to 128. The DTL value determines the number of bytes to be transferred. If DTL is less than 128, the FDC transfers the specified number of bytes to the host. For reads, it continues to read the entire 128-byte sector and checks for CRC errors. For writes, it completes the 128-byte sector by filling in zeros. If N is not set to 00 Hex, DTL should be set to FF Hex and has no impact on the number of bytes transferred.

TABLE 14-17: SECTOR SIZES

N	Sector Size
00	128 bytes
01	256 bytes
02	512 bytes
03	1024 bytes
…	…
07	16 Kbytes

The amount of data which can be handled with a single command to the FDC depends upon MT (multi-track) and N (number of bytes/sector).

The Multi-Track function (MT) allows the FDC to read data from both sides of the diskette. For a particular cylinder, data will be transferred starting at Sector 1, Side 0 and completing the last sector of the same track at Side 1.

If the host terminates a read or write operation in the FDC, the ID information in the result phase is dependent upon the state of the MT bit and EOT byte. Refer to **Table 14-18**.

At the completion of the Read Data command, the head is not unloaded until after the Head Unload Time Interval (specified in the Specify command) has elapsed. If the host issues another command before the head unloads, then the head settling time may be saved between subsequent reads.

If the FDC detects a pulse on the INDEX# pin twice without finding the specified sector (meaning that the diskette's index hole passes through index detect logic in the drive twice), the FDC sets the IC code in Status Register 0 to "01" indicating abnormal termination, sets the ND bit in Status Register 1 to "1" indicating a sector not found, and terminates the Read Data Command.

After reading the ID and Data Fields in each sector, the FDC checks the CRC bytes. If a CRC error occurs in the ID or data field, the FDC sets the IC code in Status Register 0 to "01" indicating abnormal termination, sets the DE bit flag in Status Register 1 to "1", sets the DD bit in Status Register 2 to "1" if CRC is incorrect in the ID field, and terminates the Read Data Command. Table 14-19 describes the effect of the SK bit on the Read Data command execution and results. Except where noted in Table 14-19, the C or R value of the sector address is automatically incremented (see Table 14-21 on page 183).

TABLE 14-18: EFFECTS OF MT AND N BITS

MT	N	Maximum Transfer Capacity	Final Sector Read from Disk
0	1	256 x 26 = 6,656	26 at side 0 or 1
1	1	256 x 52 = 13,312	26 at side 1
0	2	512 x 15 = 7,680	15 at side 0 or 1
1	2	512 x 30 = 15,360	15 at side 1
0	3	1024 x 8 = 8,192	8 at side 0 or 1
1	3	1024 x 16 = 16,384	16 at side 1

TABLE 14-19: SKIP BIT VS. READ DATA COMMAND

SK Bit Value	Data Address Mark Type Encountered	Results		
		Sector Read?	CM Bit of ST2 Set?	Description of Results
0	Normal Data	Yes	No	Normal termination.
0	Deleted Data	Yes	Yes	Address not incremented. Next sector not searched for. Normal termination.
1	Normal Data	Yes	No	Normal termination.
1	Deleted Data	No	Yes	Sector not read ("skipped").

14.9.2 READ DELETED DATA

This command is the same as the Read Data command, only it operates on sectors that contain a Deleted Data Address Mark at the beginning of a Data Field.

Table 14-20 describes the effect of the SK bit on the Read Deleted Data command execution and results. Except where noted in Table 14-20, the C or R value of the sector address is automatically incremented (see Table 14-21).

TABLE 14-20: SKIP BIT VS. READ DELETED DATA COMMAND

SK Bit Value	Data Address Mark Type Encountered	Results		
		Sector Read?	CM Bit of ST2 Set?	Description of Results
0	Normal Data	Yes	Yes	Address not incremented. Next sector not searched for.
0	Deleted Data	Yes	No	Normal termination.
1	Normal Data	No	Yes	Normal termination. Sector not read ("skipped").
1	Deleted Data	Yes	No	Normal termination.

14.9.3 READ A TRACK

This command is similar to the Read Data command except that the entire data field is read continuously from each of the sectors of a track. Immediately after encountering a pulse on the INDEX# pin, the FDC starts to read all data fields on the track as continuous blocks of data without regard to logical sector numbers. If the FDC finds an error in the ID or DATA CRC check bytes, it continues to read data from the track and sets the appropriate error bits at the end of the command. The FDC compares the ID information read from each sector with the specified value in the command and sets the ND flag of Status Register 1 to a “1” if there no comparison. Multi-track or skip operations are not allowed with this command. The MT and SK bits (bits D7 and D5 of the first command byte respectively) should always be set to “0”.

This command terminates when the EOT specified number of sectors has not been read. If the FDC does not find an ID Address Mark on the diskette after the second occurrence of a pulse on the INDEX# pin, then it sets the IC code in Status Register 0 to “01” (abnormal termination), sets the MA bit in Status Register 1 to “1”, and terminates the command.

TABLE 14-21: RESULT PHASE

MT	Head	Final Sector Transferred to	ID Information at Result Phase			
		HOST	C	H	R	N
0	0	Less than EOT	NC	NC	R + 1	NC
		Equal to EOT	C + 1	NC	01	NC
	1	Less than EOT	NC	NC	R + 1	NC
		Equal to EOT	C + 1	NC	01	NC
1	0	Less than EOT	NC	NC	R + 1	NC
		Equal to EOT	NC	LSB	01	NC
	1	Less than EOT	NC	NC	R + 1	NC
		Equal to EOT	C + 1	LSB	01	NC

NC: No Change, the same value as the one at the beginning of command execution.

LSB: Least Significant Bit, the LSB of H is complemented.

14.9.4 WRITE DATA

After the Write Data command has been issued, the FDC loads the head (if it is in the unloaded state), waits the specified head load time if unloaded (defined in the Specify command), and begins reading ID fields. When the sector address read from the diskette matches the sector address specified in the command, the FDC reads the data from the host via the FIFO and writes it to the sector’s data field.

After writing data into the current sector, the FDC computes the CRC value and writes it into the CRC field at the end of the sector transfer. The Sector Number stored in “R” is incremented by one, and the FDC continues writing to the next data field. The FDC continues this “Multi-Sector Write Operation”. Upon receipt of a terminal count signal or if a FIFO over/under run occurs while a data field is being written, then the remainder of the data field is filled with zeros. The FDC reads the ID field of each sector and checks the CRC bytes. If it detects a CRC error in one of the ID fields, it sets the IC code in Status Register 0 to “01” (abnormal termination), sets the DE bit of Status Register 1 to “1”, and terminates the Write Data command.

The Write Data command operates in much the same manner as the Read Data command. The following items are the same. Please refer to the Read Data Command for details:

Transfer Capacity

EN (End of Cylinder) bit

ND (No Data) bit

Head Load, Unload Time Interval

ID information when the host terminates the command

Definition of DTL when N = 0 and when N does not = 0

14.9.5 WRITE DELETED DATA

This command is almost the same as the Write Data command except that a Deleted Data Address Mark is written at the beginning of the Data Field instead of the normal Data Address Mark. This command is typically used to mark a bad sector containing an error on the floppy disk.

14.9.6 VERIFY

The Verify command is used to verify the data stored on a disk. This command acts exactly like a Read Data command except that no data is transferred to the host. Data is read from the disk and CRC is computed and checked against the previously-stored value.

Because data is not transferred to the host, the TC cycle cannot be used to terminate this command. By setting the EC bit to “1”, an implicit TC will be issued to the FDC. This implicit TC will occur when the SC value has decremented to 0 (an SC value of 0 will verify 256 sectors). This command can also be terminated by setting the EC bit to “0” and the EOT value equal to the final sector to be checked. If EC is set to “0”, DTL/SC should be programmed to 0FFH. Refer to Table 14-21 on page 183 and Table 14-22 on page 184 for information concerning the values of MT and EC versus SC and EOT value.

Definitions:

Sectors Per Side = Number of formatted sectors per each side of the disk.

Sectors Remaining = Number of formatted sectors left which can be read, including side 1 of the disk if MT is set to “1”.

TABLE 14-22: VERIFY COMMAND RESULT PHASE

MT	EC	SC/EOT Value	Termination Result
0	0	SC = DTL EOT <= # Sectors Per Side	Success Termination Result Phase Valid
0	0	SC = DTL EOT > # Sectors Per Side	Unsuccessful Termination Result Phase Invalid
0	1	SC <= # Sectors Remaining AND EOT <= # Sectors Per Side	Successful Termination Result Phase Valid
0	1	SC > # Sectors Remaining OR EOT > # Sectors Per Side	Unsuccessful Termination Result Phase Invalid
1	0	SC = DTL EOT <= # Sectors Per Side	Successful Termination Result Phase Valid
1	0	SC = DTL EOT > # Sectors Per Side	Unsuccessful Termination Result Phase Invalid
1	1	SC <= # Sectors Remaining AND EOT <= # Sectors Per Side	Successful Termination Result Phase Valid
1	1	SC > # Sectors Remaining OR EOT > # Sectors Per Side	Unsuccessful Termination Result Phase Invalid

Note 14-3 If MT is set to “1” and the SC value is greater than the number of remaining formatted sectors on Side 0, verifying will continue on Side 1 of the disk.

14.9.7 FORMAT A TRACK

The Format command allows an entire track to be formatted. After a pulse from the INDEX# pin is detected, the FDC starts writing data on the disk including gaps, address marks, ID fields, and data fields per the IBM System 34 or 3740 format (MFM or FM respectively). The particular values that will be written to the gap and data field are controlled by the values programmed into N, SC, GPL, and D which are specified by the host during the command phase. The data field of the sector is filled with the data byte specified by D. The ID field for each sector is supplied by the host; that is, four data bytes per sector are needed by the FDC for C, H, R, and N (cylinder, head, sector number and sector size respectively).

After formatting each sector, the host must send new values for C, H, R and N to the FDC for the next sector on the track. The R value (sector number) is the only value that must be changed by the host after each sector is formatted. This allows the disk to be formatted with nonsequential sector addresses (interleaving). Incrementing and formatting continues for the whole track until the FDC encounters a pulse on the INDEX# pin again and terminates the command.

Table 14-24 on page 185 contains typical values for gap fields which are dependent upon the size of the sector and the number of sectors on each track. Actual values can vary due to drive electronics.

TABLE 14-23: FORMAT FIELDS

SYSTEM 34 (DOUBLE DENSITY) FORMAT																			
GAP4a 80x 4E	SYNC 12x 00	IAM		GAP 1 50x 4E	SYNC 12x 00	IDAM		CYL	HD	SEC	NO	CRC	GAP 2 22x 4E	SYNC 12x 00	DATA AM		DATA	CRC	GAP 3	GAP 4b
		3x C2	FC			3x A1	FE								3x A1	FB F8				
SYSTEM 3740 (SINGLE DENSITY) FORMAT																			
GAP4a 40x FF	SYNC 6x 00	IAM		GAP 1 26x FF	SYNC 6x 00	IDAM		CYL	HD	SEC	NO	CRC	GAP 2 11x FF	SYNC 6x 00	DATA AM		DATA	CRC	GAP 3	GAP 4b
		FC				FE									FB or F8					
PERPENDICULAR FORMAT																			
GAP4a 80x 4E	SYNC 12x 00	IAM		GAP 1 50x 4E	SYNC 12x 00	IDAM		CYL	HD	SEC	NO	CRC	GAP 2 41x 4E	SYNC 12x 00	DATA AM		DATA	CRC	GAP 3	GAP 4b
		3x C2	FC			3x A1	FE								3x A1	FB F8				

TABLE 14-24: TYPICAL VALUES FOR FORMATTING

	Format	Sector Size	N	SC	GPL1	GPL2
5.25” Drives	FM	128	00	12	07	09
		128	00	10	10	19
		512	02	08	18	30
		1024	03	04	46	87
		2048	04	02	C8	FF
		4096	05	01	C8	FF
		...	...			
	MFM	256	01	12	0A	0C
		256	01	10	20	32
		512*	02	09	2A	50
		1024	03	04	80	F0
		2048	04	02	C8	FF
		4096	05	01	C8	FF
		...	...			
3.5” Drives	FM	128	0	0F	07	1B
		256	1	09	0F	2A
		512	2	05	1B	3A
3.5” Drives	MFM	256	1	0F	0E	36
		512**	2	09	1B	54
		1024	3	05	35	74

GPL1 = suggested GPL values in Read and Write commands to avoid splice point between data field and ID field of contiguous sections.
GPL2 = suggested GPL value in Format A Track command.
*PC/AT values (typical)
**PS/2 values (typical). Applies with 1.0 MB and 2.0 MB drives.

Note: All values except sector size are in hex.

14.9.8 CONTROL COMMANDS

Control commands differ from the other commands in that no data transfer takes place. Three commands generate an interrupt when complete: Read ID, Recalibrate, and Seek. The other control commands do not generate an interrupt.

Read ID

The Read ID command is used to find the present position of the recording heads. The FDC stores the values from the first ID field it is able to read into its registers. If the FDC does not find an ID address mark on the diskette after the second occurrence of a pulse on the INDEX# pin, it then sets the IC code in Status Register 0 to "01" (abnormal termination), sets the MA bit in Status Register 1 to "1", and terminates the command.

The following commands will generate an interrupt upon completion. They do not return any result bytes. It is highly recommended that control commands be followed by the Sense Interrupt Status command. Otherwise, valuable interrupt status information will be lost.

Recalibrate

The Recalibrate Command causes the read/write head within the FDC to retract to the track 0 position. The FDC clears the contents of the PCN counter and checks the status of the TRK0# pin from the FDD. As long as the TRK0# pin is low, the DIR pin remains 0 and step pulses are issued. When the TRK0# pin goes high, the SE bit in Status Register 0 is set to "1" and the command is terminated. If the TRK0# pin is still low after 79 step pulses have been issued, the FDC sets the SE and the EC bits of Status Register 0 to "1" and terminates the command. Disks capable of handling more than 80 tracks per side may require more than one Recalibrate command to return the head back to physical Track 0.

This command does not have a result phase. The Sense Interrupt Status command must be issued after the Recalibrate command to effectively terminate it and provide verification of the head position (PCN). During the command phase of the recalibrate operation, the FDC is in the BUSY state, but during the execution phase it is in a NON-BUSY state. At this time, another Recalibrate command may be issued, and in this manner parallel Recalibrate operations may be done on up to four drives at once. Upon power up, the software must issue a Recalibrate command to properly initialize all drives and the controller.

Seek

The read/write head within the drive is moved from track to track under the control of the Seek command. The FDC compares the PCN, which is the current head position, with the NCN and performs the following operation if there is a difference:

- PCN < NCN: Direction signal to drive set to "1" (step in) and issues step pulses.
- PCN > NCN: Direction signal to drive set to "0" (step out) and issues step pulses.

The rate at which step pulses are issued is controlled by SRT (Stepping Rate Time) in the Specify command. After each step pulse is issued, NCN is compared against PCN, and when NCN = PCN the SE bit in Status Register 0 is set to "1" and the command is terminated. During the command phase of the seek or recalibrate operation, the FDC is in the BUSY state, but during the execution phase it is in the NON-BUSY state. At this time, another Seek or Recalibrate command may be issued, and in this manner, parallel seek operations may be done on up to four drives at once.

Note that if implied seek is not enabled, the read and write commands should be preceded by:

1. Seek command - Step to the proper track
2. Sense Interrupt Status command - Terminate the Seek command
3. Read ID - Verify head is on proper track
4. Issue Read/Write command.

The Seek command does not have a result phase. It is highly recommended that the Sense Interrupt Status command is issued after the Seek command to terminate it and provide verification of the head position (PCN). The H bit (Head Address) in ST0 will always return to a "0". When exiting POWERDOWN mode, the FDC clears the PCN value and the status information to zero. Prior to issuing the POWERDOWN command, it is highly recommended that the user service all pending interrupts through the Sense Interrupt Status command.

14.10 Sense Interrupt Status

An interrupt signal is generated by the FDC for one of the following reasons:

1. Upon entering the Result Phase of:

a) Read Data command
b) Read A Track command
c) Read ID command

d) Read Deleted Data command
e) Write Data command
f) Format A Track command
g) Write Deleted Data command
h) Verify command
2. End of Seek, Relative Seek, or Recalibrate command

The Sense Interrupt Status command resets the interrupt signal and, via the IC code and SE bit of Status Register 0, identifies the cause of the interrupt.

TABLE 14-25: INTERRUPT IDENTIFICATION

SE	IC	Interrupt Due to
0	11	Polling
1	00	Normal termination of Seek or Recalibrate command
1	01	Abnormal termination of Seek or Recalibrate command

The Seek, Relative Seek, and Recalibrate commands have no result phase. The Sense Interrupt Status command must be issued immediately after these commands to terminate them and to provide verification of the head position (PCN). The H (Head Address) bit in ST0 will always return a “0”. If a Sense Interrupt Status is not issued, the drive will continue to be BUSY and may affect the operation of the next command.

Sense Drive Status

Sense Drive Status obtains drive status information. It has not execution phase and goes directly to the result phase from the command phase. Status Register 3 contains the drive status information.

Specify

The Specify command sets the initial values for each of the three internal times. The HUT (Head Unload Time) defines the time from the end of the execution phase of one of the read/write commands to the head unload state. The SRT (Step Rate Time) defines the time interval between adjacent step pulses. Note the spacing between the first and second step pulses may be shorter than the remaining step pulses. The HLT (Head Load Time) defines the time between when the Head Load signal goes high and the read/write operation starts. The values change with the data rate speed selection and are documented in Table 14-26. The values are the same for MFM and FM.

DMA operation is selected by the ND bit. When ND is “0”, the DMA mode is selected. This part does not support non-DMA mode. In DMA mode, data transfers are signaled by the DMA request cycles.

Configure

The Configure command is issued to select the special features of the FDC. A Configure command need not be issued if the default values of the FDC meet the system requirements.

TABLE 14-26: DRIVE CONTROL DELAYS (MS)

	HUT					SRT				
	2M	1M	500K	300K	250K	2M	1M	500K	300K	250K
0	64	128	256	426	512	4	8	16	26.7	32
1	4	8	16	26.7	32	3.75	7.5	15	25	30
...	...	...	...	...	...	...	...	...	...	...
E	56	112	224	373	448	0.5	1	2	3.33	4
F	60	120	240	400	480	0.25	0.5	1	1.67	2

	HLT				
	2M	1M	500K	300K	250K
00	64	128	256	426	512
01	0.5	1	2	3.3	4
02	1	2	4	6.7	8
...	...	...	...	...	...
7F	63	126	252	420	504
7F	63.5	127	254	423	508

Configure Default Values:

EIS - No Implied Seeks

EFIFO - FIFO Disabled

POLL - Polling Enabled

FIFOTHR - FIFO Threshold Set to 1 Byte

PRETRK - Pre-Compensation Set to Track 0

EIS - Enable Implied Seek. When set to “1”, the FDC will perform a Seek operation before executing a read or write command. Defaults to no implied seek.

EFIFO - A “1” disables the FIFO (default). This means data transfers are asked for on a byte-by-byte basis. Defaults to “1”, FIFO disabled. The threshold defaults to “1”.

POLL - Disable polling of the drives. Defaults to “0”, polling enabled. When enabled, a single interrupt is generated after a reset. No polling is performed while the drive head is loaded and the head unload delay has not expired.

FIFOTHR - The FIFO threshold in the execution phase of read or write commands. This is programmable from 1 to 16 bytes. Defaults to one byte. A “00” selects one byte; “0F” selects 16 bytes.

PRETRK - Pre-Compensation Start Track Number. Programmable from track 0 to 255. Defaults to track 0. A “00” selects track 0; “FF” selects track 255.

Version

The Version command checks to see if the controller is an enhanced type or the older type (765A). A value of 90 H is returned as the result byte.

Relative Seek

The command is coded the same as for Seek, except for the MSB of the first byte and the DIR bit.

DIR Head Step Direction Control

RCN Relative Cylinder Number that determines how many tracks to step the head in or out from the current track number.

DIR	Action
0 1	Step Head Out Step Head In

The Relative Seek command differs from the Seek command in that it steps the head the absolute number of tracks specified in the command instead of making a comparison against an internal register. The Seek command is good for drives that support a maximum of 256 tracks. Relative Seeks cannot be overlapped with other Relative Seeks. One Relative Seek can be active at a time. Relative Seeks may be overlapped with Seeks and Recalibrates. Bit 4 of Status Register 0 (EC) will be set if Relative Seek attempts to step outward beyond Track 0.

As an example, assume that a floppy drive has 300 usable tracks. The host needs to read track 300 and the head is on any track (0-255). If a Seek command is issued, the head will stop at track 255. If a Relative Seek command is issued, the FDC will move the head the specified number of tracks, regardless of the internal cylinder position register (but will increment the register). If the head was on track 40 (d), the maximum track that the FDC could position the head on using Relative Seek will be 295 (D), the initial track + 255 (D). The maximum count that the head can be moved with a single Relative Seek command is 255 (D).

The internal register, PCN, will overflow as the cylinder number crosses track 255 and will contain 39 (D). The resulting PCN value is thus (RCN + PCN) mod 256. Functionally, the FDC starts counting from 0 again as the track number goes above 255 (D). It is the user’s responsibility to compensate FDC functions (precompensation track number) when accessing tracks greater than 255. The FDC does not keep track that it is working in an “extended track area” (greater than 255). Any command issued will use the current PCN value except for the Recalibrate command, which only looks for the TRACK0 signal. Recalibrate will return an error if the head is farther than 79 due to its limitation of issuing a maximum of 80 step pulses. The user simply needs to issue a second Recalibrate command. The Seek command and implied seeks will function correctly within the 44 (D) track (299-255) area of the “extended track area”. It is the user’s responsibility not to issue a new track position that will exceed the maximum track that is present in the extended area.

To return to the standard floppy range (0-255) of tracks, a Relative Seek should be issued to cross the track 255 boundary.

A Relative Seek can be used instead of the normal Seek, but the host is required to calculate the difference between the current head location and the new (target) head location. This may require the host to issue a Read ID command to ensure that the head is physically on the track that software assumes it to be. Different FDC commands will return different cylinder results which may be difficult to keep track of with software without the Read ID command.

Perpendicular Mode

The Perpendicular Mode command should be issued prior to executing Read/Write/Format commands that access a disk drive with perpendicular recording capability. With this command, the length of the Gap2 field and VCO enable timing can be altered to accommodate the unique requirements of these drives. Table 14-27 on page 190 describes the effects of the WGATE and GAP bits for the Perpendicular Mode command. Upon a reset, the FDC will default to the conventional mode (WGATE = 0, GAP = 0).

Selection of the 500 Kbps and 1 Mbps perpendicular modes is independent of the actual data rate selected in the Data Rate Select Register. The user must ensure that these two data rates remain consistent.

The Gap2 and VCO timing requirements for perpendicular recording type drives are dictated by the design of the read/write head. In the design of this head, a pre-erase head precedes the normal read/write head by a distance of 200 micrometers. This works out to about 38 bytes at a 1 Mbps recording density. Whenever the write head is enabled by the Write Gate signal, the pre-erase head is also activated at the same time. Thus, when the write head is initially turned on, flux transitions recorded on the media for the first 38 bytes will not be preconditioned with the pre-erase head since it has not yet been activated. To accommodate this head activation and deactivation time, the Gap2 field is expanded to a length of 41 bytes. The Format Fields table illustrates the change in the Gap2 field size for the perpendicular format.

On the read back by the FDC, the controller must begin synchronization at the beginning of the sync field. For the conventional mode, the internal PLL VCO is enabled (VCOEN) approximately 24 bytes from the start of the Gap2 field. But, when the controller operates in the 1 Mbps perpendicular mode (WGATE = 1, GAP = 1), VCOEN goes active after 43 bytes to accommodate the increased Gap2 field size. For both cases, and approximate two-byte cushion is maintained from the beginning of the sync field for the purposes of avoiding write splices in the presence of motor speed variation.

For the Write Data case, the FDC activates Write Gate at the beginning of the sync field under the conventional mode. The controller then writes a new sync field, data address mark, data field, and CRC. With the pre-erase head of the perpendicular drive, the write head must be activated in the Gap2 field to insure a proper write of the new sync field. For the 1 Mbps perpendicular mode (WGATE = 1, GAP = 1), 38 bytes will be written in the Gap2 space. Since the bit density is proportional to the data rate, 19 bytes will be written in the Gap2 field for the 500 Kbps perpendicular mode (WGATE = 1, GAP =0).

It should be noted that none of the alterations in Gap2 size, VCO timing, or Write Gate timing affect normal program flow. The information provided here is just for background purposes and is not needed for normal operation. Once the Perpendicular Mode command is invoked, FDC software behavior from the user standpoint is unchanged.

The perpendicular mode command is enhanced to allow specific drives to be designated Perpendicular recording drives. This enhancement allows data transfers between Conventional and Perpendicular drives without having to issue Perpendicular mode commands between the accesses of the different drive types, nor having to change write pre-compensation values.

When both GAP and WGATE bits of the PERPENDICULAR MODE COMMAND are both programmed to “0” (Conventional mode), then D0, D1, D2, D3, and D4 can be programmed independently to “1” for that drive to be set automatically to Perpendicular mode. In this mode the following set of conditions also apply:

- The GAP2 written to a perpendicular drive during a write operation will depend upon the programmed data rate.
- The write pre-compensation given to a perpendicular mode drive will be 0ns.
- For D0-D3 programmed to “0” for conventional mode drives any data written will be at the currently programmed write pre-compensation.

Note 14-4 Bits D0-D3 can only be overwritten when OW is programmed as a “1”.If either GAP or WGATE is a “1” then D0-D3 are ignored.

Software and hardware resets have the following effect on the PERPENDICULAR MODE COMMAND:

1. “Software” resets (via the DOR or DSR registers) will only clear GAP and WGATE bits to “0”. D0-D3 are unaffected and retain their previous value.
2. “Hardware” resets will clear all bits (GAP, WGATE and D0-D3) to “0”, i.e all conventional mode.

TABLE 14-27: EFFECTS OF WGATE AND GAP BITS

WGATE	Gap	Mode	Length of GAP2 Format Field	Portion of GAP 2 Written by Write Data Operation
0	0	Conventional	22 Bytes	0 Bytes
0	1	Perpendicular (500 Kbps)	22 Bytes	19 Bytes
1	0	Reserved (Conventional)	22 Bytes	0 Bytes
1	1	Perpendicular (1 Mbps)	41 Bytes	38 Bytes

Lock

In order to protect systems with long DMA latencies against older application software that can disable the FIFO the LOCK Command has been added. This command should only be used by the FDC routines, and application software should refrain from using it. If an application calls for the FIFO to be disabled then the CONFIGURE command should be used.

The LOCK command defines whether the EFIFO, FIFOTHR, and PRETRK parameters of the CONFIGURE command can be RESET by the DOR and DSR registers. When the LOCK bit is set to logic “1” all subsequent “software RESETS by the DOR and DSR registers will not change the previously set parameters to their default values. All “hardware” RESET from the PCI RESET# pin will set the LOCK bit to logic “0” and return the EFIFO, FIFOTHR, and PRETRK to their default values. A status byte is returned immediately after issuing a LOCK command. This byte reflects the value of the LOCK bit set by the command byte.

Enhanced Dumpreg

The DUMPREG command is designed to support system run-time diagnostics and application software development and debug. To accommodate the LOCK command and the enhanced PERPENDICULAR MODE command the eighth byte of the DUMPREG command has been modified to contain the additional data from these two commands.

14.11 Compatibility

The SCH5636 was designed with software compatibility in mind. It is a fully backwards- compatible solution with the older generation 765A/B disk controllers. The FDC also implements on-board registers for compatibility with the PS/2, as well as PC/AT and PC/XT, floppy disk controller subsystems. After a hardware reset of the FDC, all registers, functions and enhancements default to a PC/AT, PS/2 or PS/2 Model 30 compatible operating mode, depending on how the IDENT and MFM bits are configured by the system BIOS.

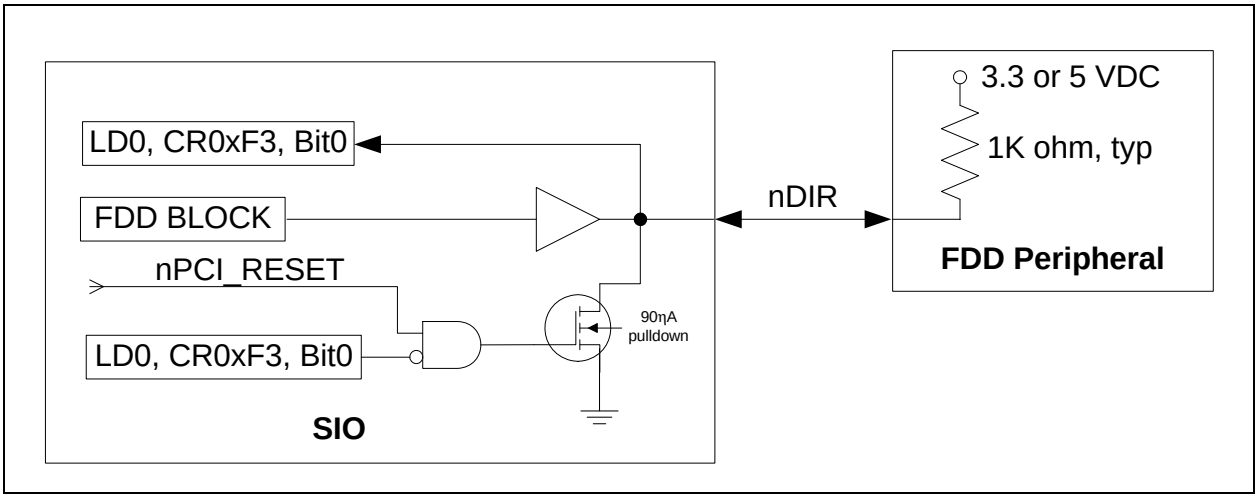
14.12 Floppy Drive Presence Detection

The Floppy Drive Presence Detection Circuit is illustrated in Figure 14-1. The nDIR pin is an output pin on the SCH5636 Floppy Interface. Standard Floppy Peripherals have a strong pull-ups on this pin. The SCH5636 has an on chip 90µA weak pull-down which is selectively activated to detect the presence of the nDIR pull-up from the Floppy peripheral. The 90µA weak pull-down is deactivated when PCI RESET# is asserted or once the Floppy activate bit (LD0,CR0x30-bit0) is set to ‘1’.

The state of the nDIR pin is reported to the FDSTAT bit at LD0,CR0xF3-bit0. See Section 14.12.1, "FDSTAT Register," on page 191.

APPLICATION NOTE: The programmer can read the state of the FDSTAT bit at LD0,CR0xF3-bit0, If this bit is set to ‘1’, the a FDD is present and in the normal course of initializing the floppy, the programmer will set the Floppy activate bit (LD0,CR0xF3-bit0) is set to ‘1’.

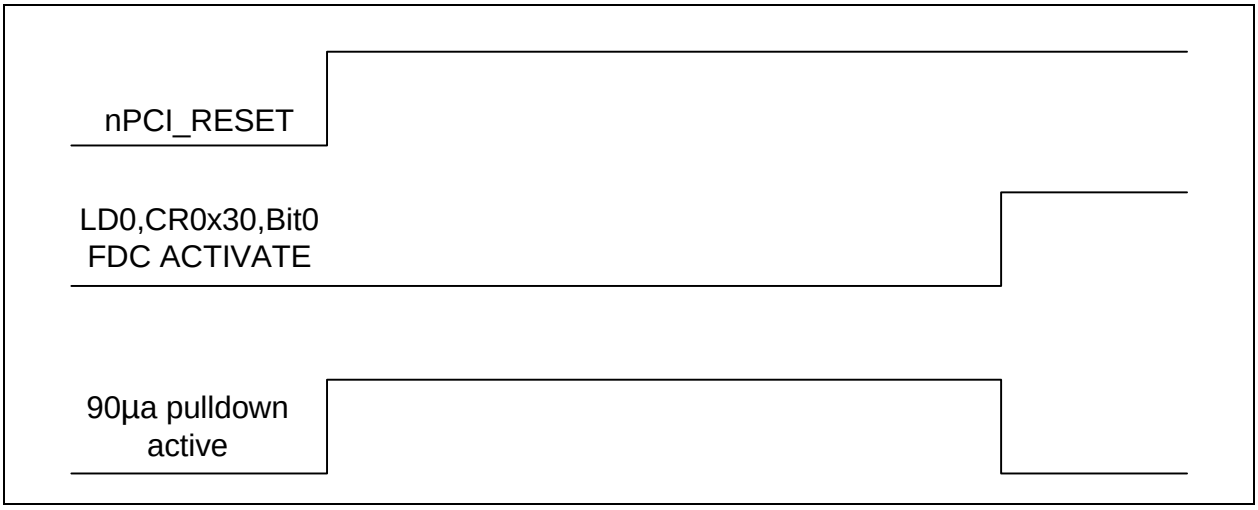
FIGURE 14-1: FLOPPY DRIVE PRESENCE DETECTION CIRCUIT



Note: Figure 14-1 is intended for illustration purposes only and does not portray specific implementation details.

Note 14-5 The nDIR buffer defaults to open drain. The nDIR can be configured to be push-pull via LD0,CR0xF0-bit6. The FDSTAT bit at LD0,CR0xF3-bit0 is reliable only if nDIR is open drain; therefore, LD0,CR0xF0-bit6 should remain clear until after the Floppy has been detected.

FIGURE 14-2: NDIR WEAK PULL-DOWN ACTIVATION TIMING



14.12.1 FDSTAT REGISTER

The register FDSTAT is in the Configuration space at Logical Device 0, Index 0xF3.

- Bit 0: FDPRES indicates whether a floppy drive has been detected: ‘1’ means that at least one drive is present, and ‘0’ means that no drive is present.
- Bits 1 through 7, reserved for future use. Read-only, read as zero always.

15.0 SERIAL PERIPHERAL INTERFACE

15.1 General Description

The SPI interface may be used to load the internal SRAM with program code and data from an external serial flash device. For a serial flash device to be compatible with the SCH5636, it must meet the following minimum requirements:

- Serial flash devices must ignore the upper address bits such that an address of FF_FFFFh aliases to the top of the memory
- SPI Compatible Mode 0 support (clock phase is 0 and data is latched on the rising edge of the clock).
- The SPI Read command is 03h.
- If the device receives a command that is not supported or incomplete, the device must complete the cycle gracefully without altering the internal flash contents.

An example serial flash device is the SST 8Mbit SPI Serial Flash, part number SST25VF080B.

16.0 PME SUPPORT

The SCH5636 offers support for power management events (PMEs), also referred to as a System Control Interrupt (SCI) events in an ACPI system. A power management event is indicated to the chipset via the assertion of the IO_PME# signal and functions as a wake event (that is, a PME can wake a system that is in a sleep state). In the SCH5636, the IO_PME# is asserted by active transitions on the ring indicator inputs RI1# and RI2#, active keyboard-data edges, active mouse-data edges, programmable edges on GPIO pins and temperature events. The GP041/IO_PME# pin, when selected for the IO_PME# function, can be programmed to be active high or active low via the polarity bit in the GP041 Pin Control register. The output buffer type of the pin can also be programmed to be open-drain or push-pull.

PMEs can also be enabled onto the Serial IRQ stream, by configuring a SERIRQ channel for the Runtime Registers Logical Device, with the SELECT function set to 0. See Section 9.6, "SERIRQ Interrupts".

Note:	If the host enables UART2 and requires the RI2# pin for wakeup, the host should configure the Pin Control register for GP062/RI2# so that the Mux Control field selects RI2# and the Interrupt Detection selects falling edge. If the host enables UART1 and requires the RI1# pin for wakeup, the host should configure the Pin Control register for GP052/RI1# so that the Mux Control field selects RI1# and the Interrupt Detection selects falling edge.

The PME functionality is controlled by the PME_STS Register and the PME_EN Register in the Runtime Registers block, in Logical Device A. The PME Enable bit, PME_EN, globally controls PME Wake-up events. When PME_EN is inactive, the IO_PME# signal can not be asserted. When PME_EN is asserted, any wake source whose individual PME Wake Enable register bit is asserted can cause IO_PME# to become asserted.

The PME Status register indicates that an enabled wake source has occurred, and if the PME_EN bit is set, asserted the IO_PME# signal. The PME Status bit is asserted by when any PME Status bit is asserted. PME Status registers are described in Section 19.0, "Runtime Registers", which defines whether PME Status bits are set on edge events or level events. The PME_Status bit in the PME_STS Register will become asserted independent of the state of the global PME enable bit, PME_EN.

The following pertains to the PME status bits for each event:

- The output of the status bit for each event is combined with the corresponding enable bit to set the PME status bit.
- The status bit for any pending events must be cleared in order to clear the PME_STS bit. Status bits are cleared on a write of ‘1’. Read-only status bits must be cleared at their source.

See Section 11.5.8, "Keyboard and Mouse PME Generation" in Section 11.0, "Keyboard Controller" for information about using the keyboard and mouse signals to generate a PME.

The PME registers are located in system I/O space at an offset from the Base Address programmed for Logical Device Ah.

See PME register descriptions in Section 19.0, "Runtime Registers".

16.1 GPIO Events

Eight GPIO pins can be used to generate Power Management Events. For GPIO events, the polarity of the edge used to set the status bit and generate a PME is controlled by the Interrupt Detection field in the Pin Control register associated with the GPIO. These GPIOs will only generate a PME if the Interrupt Detection field is configured for edge-sensitive interrupts. If the GPIO is configured for a level-sensitive interrupt, or edge detection is disabled, then the GPIO cannot generate a PME. Status bits are cleared on a write of ‘1’. These eight GPIOs can also be used to generate System Management Interrupts.

16.2 Enabling SMI Events onto the PME Pin

There is a bit in the PME_STS2 Register to show the status of the internal “group” SMI signal in the PME logic (if bit 5 of the SMI_EN2 Register is set). This bit, Devint_Status, is at bit 0 of the PME_STS2 Register register. This bit is defined as follows:

0= The group SMI output is inactive.

1= The group SMI output is active.

Devint_Status, when asserted, can generate a PME if the Devint_Status Enable bit in PME_EN2 Register is 1.

16.3 Low Battery Warning PME Event

See Section 18.0, "Low Battery Detection," on page 196 for a description of Low Battery PME events.

The Low_Bat PME event is indicated and enabled via the PME_STS2 Register and the PME_EN2 Register. See Section 19.0, "Runtime Registers," on page 198 for a description of these registers.

16.4 “Wake on Specific Key” Event

See Section 11.6, "'Wake on Specific Key' Option," on page 117 in Section 11.0, "Keyboard Controller" for a description of this event.

17.0 SMI SUPPORT

The SCH5636 implements a “group” IO_SMI# output pin. The System Management Interrupt is a non-maskable interrupt with the highest priority level used for OS transparent power management. The SMI group interrupt output consists of the enabled interrupts from each of the functional blocks in the chip and several of the GPIOs and the temperature monitoring. The GP071/IO_SMI# pin, when selected for the IO_SMI# function, can be programmed to be active high or active low via the polarity bit in the GP071 pin control register. The output buffer type of the pin can also be programmed to be open-drain or push-pull via the pin control register.

The interrupts are enabled onto the group SMI# signal as defined in the description of the SMI_STS Register. The group SMI# signal is then enabled onto the IO_SMI# output pin via bit[7] in the SMI Enable Register 2. The group SMI# signal can also be enabled onto the serial IRQ stream via Bit[6] in the SMI Enable Register 2. The configuration logic for SERIRQ can route the SMI# signal onto any SERIRQ channel. In addition, the internal SMI can be enabled onto the IO_PME# pin. Bit[5] of the SMI Enable Register 2 is used to enable the SMI output onto the IO_PME# pin (GP041). This bit will enable the internal SMI output into the PME logic through the DEVINT_STS bit in PME_STS3. See Section 16.0, "PME Support," on page 193 for more details.

17.1 SMI Registers

The event bits for generating System Management Interrupts are located in the SMI Status and Enable registers. The SMI logic for these events is implemented such that the output of the status bit for each event is combined with the corresponding enable bit in order to generate an SMI.

The SMI registers are accessed at an offset from the base I/O address for logical device A (see Section 19.0, "Runtime Registers," on page 198 for more information).

Status bits for events for super I/O devices are located in the SMI_STS1 Register and the SMI_STS2 Register. All of these status bits are cleared at the source. The SMI logic for these events is implemented such that each event is directly combined with the corresponding enable bit in order to generate an SMI.

SMI events for Low Bat, the EC Watchdog Timer and the GPIO events must be cleared in the SMI Status registers.

SMI_EN2 Register also contains the bit to enable the group SMI onto the IO_SMI# output pin via (bit[7]), the bit to enable the group SMI onto the serial IRQ stream (Bit[6]) and the bit to enable the group SMI signal onto the IO_PME# pin (Bit[5]).

17.2 GPIO SMI Events

Eight GPIO pins can be used to generate SMI events. For GPIO events, the polarity of the edge used to set the status bit and generate an SMI is controlled by the Interrupt Detection field in the Pin Control register associated with the GPIO. These GPIOs will only generate an SMI if the Interrupt Detection field is configured for edge-sensitive interrupts. If the GPIO is configured for a level-sensitive interrupt, or edge detection is disabled, then the GPIO cannot generate a SMI.These same eight GPIOs can also be used to generate Power Management Events.

17.3 Low Battery Warning SMI Event

See Section 18.0, "Low Battery Detection," on page 196 for a description of Low Battery SMI events.

The Low_Bat SMI event is indicated and enabled via the SMI_STS2 Register and the SMI_EN2 Register. See Section 19.0, "Runtime Registers," on page 198 for a description of these registers.

18.0 LOW BATTERY DETECTION

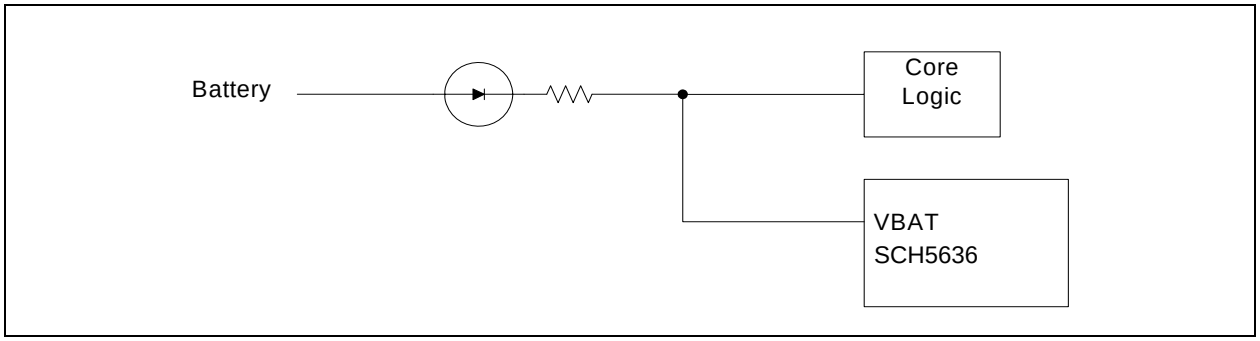
The SCH5636 can be used with an external battery to provide battery backup for the VBAT Powered RAM as well as several internal status bits. The status of the battery voltage is checked at three different times by the SCH5636:

- When the battery is replaced while VTR power is off
- On the rising edge of VTR
- On the rising edge of VCC

bit Two state bits in Runtime space, Low_Bat in the PME_STS2 Register and Low_Bat in the SMI_STS2 Register, track the status of the external battery. These bits can be accessed by the Host and used to generate Host events.

Figure 18-1, "External Battery Circuit", illustrates the external battery circuit:

FIGURE 18-1: EXTERNAL BATTERY CIRCUIT



The battery voltage measured at the VBAT pin, not at the source, so battery voltages described in this chapter incorporate the approximate 0.3V voltage drop caused by the diode in the figure.

18.1 Battery Voltage Detection

Battery voltage is monitored in the three conditions defined in the following sections:

18.1.1 VTR POWER OFF

If, while the VTR power supply is off, the external battery is removed and replaced with a battery that delivers at least approximately 1.2V, a VBAT Power On Reset (VBAT POR) is signaled. A VBAT POR will cause the internal battery-backed status bits to their default state. The VBAT Powered RAM is not reset, however, a VBAT POR indicates that the contents of the VBAT Powered RAM are indeterminate. A VBAT POR sets an internal state bit that is readable and clearable by the EC.

Removing and replacing the external battery has no effect if VTR power is on.

18.1.2 RISING EDGE OF VTR

When VTR rises above its operational threshold, the battery voltage is checked to see if it is less than approximately 1.2V. If the battery voltage is below the 1.2V threshold, the internal VBAT POR state is set; If the battery voltage is above the 1.2V threshold, the internal VBAT POR state is not modified.

18.1.3 RISING EDGE OF VCC

When VCC rises above its operational threshold, the VBAT pin is measured and compared to a threshold of approximately 2.2V. This threshold is only checked on the rising edge of VCC.

18.2 Low Battery Events

Both the Power Management Event logic and the System Management Interrupt logic have state bits that report the status of VBAT voltage. These bits are Low_Bat in the PME_STS2 Register and Low_Bat in the SMI_STS2 Register. On the rising edge of VTR, these two bits are both initialized to '1' if either the battery was replaced while VTR was off, or the battery was below threshold when VTR reached its operational threshold. In addition, both bits are set to '1' if the battery voltage is below approximately 2.2V when VCC rises above its operational threshold. Neither Low_Bat bit is modified if the battery voltage is above approximately 2.2V when VCC rises.

Both Low_Bat status bits can only be cleared when written with a '1' by software.

Since the PME enable bit is not battery backed up and is cleared on VTR POR, the Low_Bat PME status bit is not normally a wakeup event since it cannot be enabled when VTR is on but VCC is off. When VCC returns, if the PME or SMI enable bit (and other associated enable bits) are set while the respective Low_Bat bits are '1', then the corresponding event will be generated.

Both the PME and SMI Low_Bat status bits can be enabled while VCC is active. No event will be generated if neither status bit is set when enabled. If VCC goes away, the Low_Bat enable bits will remain active, but no event will be triggered because neither Low_Bat status bit can be triggered while operating on VTR power alone. When VCC returns, if the PME or SMI enable bit (and other associated enable bits) are set and the battery voltage is below approximately 2.2V, then a corresponding event will be generated.

18.3 Synchronization with the EC

The EC maintains the internal battery-back status bits that are used to configure the Low_Bat status bits in the PME and SMI status registers. This status bit retains its value across VTR power cycles, so that the EC and the Host can properly service battery events even if VTR power goes down after a low voltage event but before the EC or the Host can react to the event. In order to insure that the EC and the Host are in synch with respect to processing low voltage events, the Host should inform the EC that it has processed the Low_Bat events in the PME and SMI status registers.

19.0 RUNTIME REGISTERS

19.1 Power, Clocks and Reset

19.1.1 RESET

This block is reset when nSYS_RST is asserted. The Force Disk Change Register is also reset on nSIO_RESET.

19.2 Interrupts

19.2.1 HOST INTERRUPTS

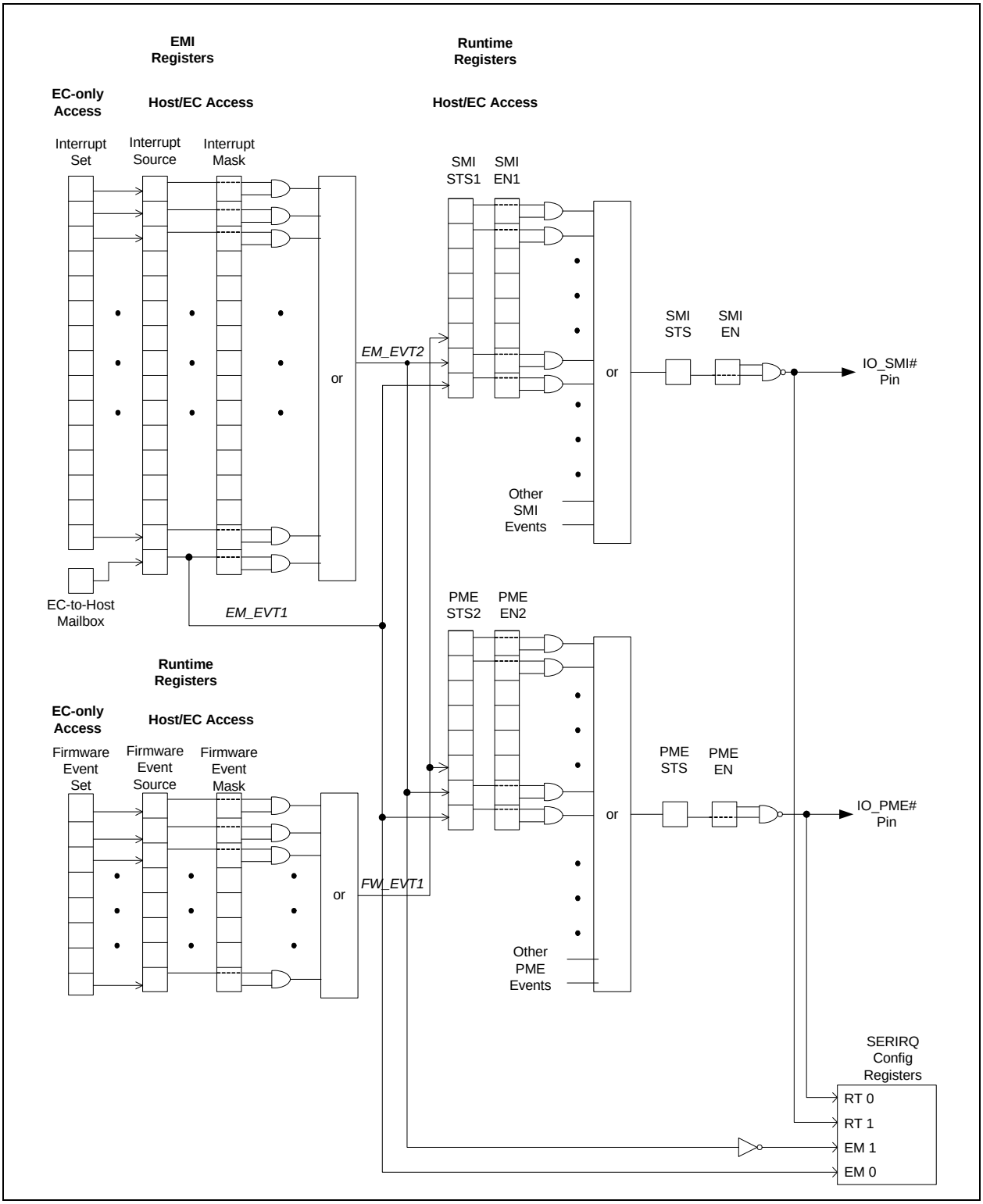
The PME_STS/PME_EN registers can assert the IO_PME# pin and thus can be used to generate a PME to the Host. The SMI_STS/SMI_EN registers can assert the IO_SMI# pin and can be used to generate an SMI to the Host. The PME can also be routed to a SERIRQ channel in the LPC Configuration block, as the first of the two interrupts from the Runtime Register block (SELECT=0). The SMI can be routed to the SERIRQ as the second interrupt from the Runtime Register block (SELECT=1).

The SMI event produces a standard active low on the serial IRQ stream, and active low on the open drain IO_SMI# pin. The PME event also produces a standard active low on the serial IRQ stream, and active low on the open drain IO_PME# pin.

Firmware can generate a PME or an SMI through three EVT event bits. The EMI generates the EM_EVT1 and EM_EVT2 signals, and the Firmware Event Status Register/Firmware Event Enable Register pair generates the FW_EVT3 signal. The three EVT signals are routed to both the PME generation logic (in PME_STS2 Register) and the SMI generation logic (in SMI_STS1 Register).

The event signal routing is illustrated in Figure 19-1, "SMI/PME Interrupt Routing":

FIGURE 19-1: SMI/PME INTERRUPT ROUTING



Note: Unless noted, all PME and SMI status bits are level-sensitive.

19.3 Registers

Table 19-1 is a register summary for the Runtime Registers. The LPC I/O address for each Run-Time Register is described below as an offset from the Base Address Register. Each Configuration register access through the Host Access Port is via its LDN indicated in Table 4-2, “Host Logical Devices on SCH5636,” on page 26 and its Host Access Port index which is described as “Host Config Index” in the tables below.

TABLE 19-1: RUNTIME REGISTERS REGISTER SUMMARY

Register Name	I/O Offset	Size	Type	Notes
RUNTIME REGISTERS				
PME_STS Register	00h	8	R/WC	
PME_EN Register	01h	8	R/W	
PME_STS1 Register	02h	8	R/WC	
PME_STS2 Register	03h	8	R/WC	
PME_STS3 Register	04h	8	R/WC	
PME_EN1 Register	05h	8	R/W	
PME_EN2 Register	06h	8	R/W	
PME_EN3 Register	07h	8	R/W	
SMI_STS Register	10h	8	R/WC	
SMI_EN Register	11h	8	R/W	
SMI_STS1 Register	12h	8	R/WC	
SMI_STS2 Register	13h	8	R/WC	
SMI_STS3 Register	14h	8	R/WC	
SMI_EN1 Register	15h	8	R/W	
SMI_EN2 Register	16h	8	R/W	
SMI_EN3 Register	17h	8	R/W	
Force Disk Change Register	20h	8	R/W	
Floppy Data Rate Select Shadow Register	21h	8	R	
UART 1 FIFO Control Shadow Register	22h	8	R	
UART 2 FIFO Control Shadow Register	23h	8	R	
Device Disable Register	24h	8	R/W	
LED Register	25h	8	R/W	
Keyboard Scan Register	26h	8	R/W	
Power Good Register	27h	8	R/W	
GPIO Select Register	28h	8	R/W	
GPIO Read Register	29h	8	R/W	
GPIO Write Register	2Ah	8	R/W	
Firmware Event Status Register	30h	8	R/WC	
Firmware Event Enable Register	31h	8	R/W	
Power Recovery Modes Register	32h	8	R/W	
Intruder RegisterIntruder Register	34h	8	R/W	
EC-ONLY REGISTERS				
Device Disable Enable Register	100h	32	R/W	
GPIO Access Enable 1 Register	104h	32	R/W	
GPIO Access Enable 2 Register	108h	32	R/W	
Event Set Register	10Ch	32	R/W	

TABLE 19-1: RUNTIME REGISTERS REGISTER SUMMARY (CONTINUED)

Register Name	I/O Offset	Size	Type	Notes
Host Clear Enable Register	110h	32	R/W	
CONFIGURATION REGISTERS				
SPEKEY	F0h	8	R/W	Note 19-1

Note 19-1 See Section 11.6, "'Wake on Specific Key' Option," on page 117.

19.3.1 PME_STS REGISTER

TABLE 19-2: PME_STS REGISTER

HOST OFFSET	00h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/WC
BIT NAME	Reserved							PME_ Status

PME_Status

PME Status. This bit is set to 1 if any bit in a PME_STS*i* register is a 1 and enabled by the corresponding bit in the PME_EN*i* register. If this bit is a 1 and PME_Enable in the PME_EN Register is a 1, the IO_PME# signal will be asserted. This bit is cleared by writing it with a 1. Writing a 0 to this bit has no effect.

0= No PME event (default)

1= PME Event active.

19.3.2 PME_EN REGISTER

TABLE 19-3: PME_EN REGISTER

HOST OFFSET	01h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							PME_ Enable

PME_Enable

PME Enable. This bit enables assertion of the IO_PME# pin.

0= IO_PME# signal assertion is disabled (default)

1= Enables assertion of the IO_PME# signal

19.3.3 PME_STS1 REGISTER

PME Wake Status 1 register. The PME_Status bit is asserted if any bit in this register is 1 and the corresponding bit in the PME_EN1 Register is also 1. All bits are cleared by writing with a 1. Writes of a 0 to any bit have no effect.

Wake events are events that can wake up the host if it is in a sleep state.

All status bits in this register are set on signal edges.

TABLE 19-4: PME_STS1 REGISTER

HOST OFFSET	02h					8-bit	HOST SIZE	
POWER	VTR					00h	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R/WC	R/WC	R/WC	R/WC	R/WC	R
BIT NAME	Reserved		SPEKEY	MOUSE	KBD	RI1	RI2	Reserved

RI2

This bit is set to 1 if the Ring Indicator Wake event from UART 2 is asserted. The wake event is asserted on the falling edge of the RI2# pin.

Note: In order to generate an RI2 event, the Interrupt Detection field in the Pin Control register for GP062, the GPIO associated with RI2# as an alternate function, must be configured for falling edge interrupts.

RI1

This bit is set to 1 if the Ring Indicator Wake event from UART 1 is asserted. The wake event is asserted on the falling edge of the RI1# pin.

Note: In order to generate an RI1 event, the Interrupt Detection field in the Pin Control register for GP052, the GPIO associated with RI1# as an alternate function, must be configured for falling edge interrupts.

KBD

This bit is set to 1 if the KBD Wake event from the Keyboard PS/2 is asserted. The wake event is asserted on the falling edge of the KDAT pin.

MOUSE

This bit is set to 1 if the Mouse Wake event from the Mouse PS/2 is asserted. The wake event is asserted on the falling edge of the MDAT pin.

SPEKEY

This bit is set to 1 if the wake on specific key wake event is asserted.

19.3.4 PME_STS2 REGISTER

PME Wake Status 2 register. All bits are cleared by writing with a 1. Writes of a 0 to any bit have no effect.

TABLE 19-5: PME_STS2 REGISTER

HOST OFFSET	03h					8-bit	HOST SIZE	
POWER	VTR					0XX0_0000b	DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R/WC	R	R	R	R/WC	R/WC
BIT NAME	Reserved	Intrusion	Low_ Bat	FW_ EVT1	EM_ EVT2	EM_ EVT1	WDT	Devint_ Status

Devint_Status

This bit is set to 1 if the IO_SMI# signal is asserted by the SMI logic.

WDT

This bit is set to 1 if a Watchdog Timer event is asserted when the watchdog times out.

EM_EVT1

This bit is set to 1 if the EC_WR bit (bit 0) of the Interrupt Source Register in the EM Interface is asserted.

EM_EVT2

This bit is set to 1 if any of the 16 bits in the Interrupt Source Register in the EM Interface is asserted and enabled by the corresponding bit in the Interrupt Enable Register in the EM Interface.

FW_EVT1

This bit is set to 1 if any of the bits in the Firmware Event Status Register in the Runtime Register block is asserted and enabled by the corresponding bit in the Firmware Event Enable Register.

Low_Bat

When VTR reaches the its operating threshold, Low_Bat is set to the value of the VBAT_POR bit in the battery-backed internal status register. When VCC RESET is de-asserted, this bit is also set to 1 if the VBAT_Low bit in the battery-backed internal status register is asserted and not modified otherwise. The low battery event will not normally be a PME wakeup event because the Low_Bat Enable bit will be cleared on VTR POR, when Low_Bat is set. See Section 18.0, "Low Battery Detection," on page 196 for more details on the battery monitoring logic.

Intrusion

This bit is a copy of INTRUSION in the Intruder Register and follows its behavior. If enabled, a PME is generated when INTRUSION is high. INTRUSION in the Intruder Register must be cleared in order for this bit to be cleared.

19.3.5 PME_STS3 REGISTER

PME Wake Status 3 register. All bits are cleared by writing with a 1. Writes of a 0 to any bit have no effect.

For each GPIO monitored by this register, a PME is asserted if the GPIO pin value changes according to the Interrupt Detection field in the Pin Control Register associated with the GPIO. PME events are only asserted if the Interrupt Detection field is set to an edge-triggered events. If the field is set to a level-sensitive event, no PME will be generated even if the pin matches the selected level.

Note:	If the Interrupt Detection field for one of the GPIO pins monitored by this register is changed from no-edge-detection to edge-triggered interrupts while the pin is high, the status bit will be set. Host software should clear the GPIO status bit in this register whenever it reconfigures the Interrupt Detection field in the Pin Control Register for the GPIO, in order to avoid a spurious event.

TABLE 19-6: PME_STS3 REGISTER

HOST OFFSET	04h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	GPIO62	GPIO54	GPIO53	GPIO35	GPIO31	GPIO25	GPIO24	GPIO21

GPIO21

This bit is set to 1 if a wake event for GPIO21 is asserted.

GPIO24

This bit is set to 1 if a wake event for GPIO24 is asserted.

GPIO25

This bit is set to 1 if a wake event for GPIO25 is asserted.

GPIO31

bit This bit is set to 1 if a wake event for GPIO31 is asserted.

GPIO35

This bit is set to 1 if a wake event for GPIO35 is asserted.

GPIO53

This bit is set to 1 if a wake event for GPIO53 is asserted.

GPIO54

This bit is set to 1 if a wake event for GPIO54 is asserted.

GPIO62

This bit is set to 1 if a wake event for GPIO62 is asserted.

19.3.6 PME_EN1 REGISTER

PME Wake Enable 1 register. This register is used to enable individual PME wake sources from the PME_STS1 Register onto the PME_Status bit in the PME_STS Register. Bit fields correspond to the fields in the PME_STS1 Register.

TABLE 19-7: PME_EN1 REGISTER

HOST OFFSET	05h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R/W	R/W	R/W	R/W	R/W	R
BIT NAME	Reserved		SPEKEY	MOUSE	KBD	RI1	RI2	Reserved

19.3.7 PME_EN2 REGISTER

PME Wake Enable 1 register. This register is used to enable individual PME wake sources from the PME_STS2 Register onto the PME_Status bit in the PME_STS Register. Bit fields correspond to the fields in the PME_STS2 Register.

TABLE 19-8: PME_EN2 REGISTER

HOST OFFSET	06h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved	Intrusion	Low_ Bat	FW_ EVT1	EM_ EVT2	EM_ EVT1	WDT	Devint_ Status

19.3.8 PME_EN3 REGISTER

PME Wake Enable 1 register. This register is used to enable individual PME wake sources from the PME_STS3 Register onto the PME_Status bit in the PME_STS Register. Bit fields correspond to the fields in the PME_STS3 Register.

TABLE 19-9: PME_EN3 REGISTER

HOST OFFSET	07h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO62	GPIO54	GPIO53	GPIO35	GPIO31	GPIO25	GPIO24	GPIO21

19.3.9 SMI_STS REGISTER

TABLE 19-10: SMI_STS REGISTER

HOST OFFSET	10h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/WC
BIT NAME	Reserved							SMI_ Status

SMI_Status

SMI Status. This bit is set to 1 if any bit in a SMI_STS*i* register is a 1 and enabled by the corresponding bit in the SMI_EN*i* register. If this bit is a 1 and SMI_Enable in the SMI_EN Register is a 1, an SMI will be asserted. An SMI will be asserted on the IO_SMI# pin, on the SERIRQ, or in the Devint_Status bit in the PME_STS2 Register, according to the respective enable bits in the SMI_EN2 Register. This bit is cleared by writing it with a 1. Writing a 0 to this bit has no effect.

0= No SMI event (default)

1= SMI Event active.

19.3.10 SMI_EN REGISTER

TABLE 19-11: SMI_EN REGISTER

HOST OFFSET	11h					8-bit	**HOST SIZE**	
POWER	VTR			00h			**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							SMI_ Enable

SMI_Enable

This bit enables assertion of the SMI function.

0= SMI function assertion is disabled (default)

1= Enables assertion of the SMI function

19.3.11 SMI_STS1 REGISTER

SMI Status 1 register. The IO_SMI# pin is asserted if any bit in this register is 1 and the corresponding bit in is also 1. All bits must be cleared at the source except as shown.

TABLE 19-12: SMI_STS1 REGISTER

HOST OFFSET	12h					8-bit	**HOST SIZE**	
POWER	VTR			0000_001Xb			**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R/WC
BIT NAME	FW_ EVT1	EM_ EVT2	EM_ EVT1	FINT	U1INT	U2INT	PINT	Low_ Bat

Low_Bat

When VTR reaches the its operating threshold, Low_Bat is set to the value of the VBAT_POR bit in the battery-backed internal status register. When VCC RESET is de-asserted, this bit is also set to 1 if the VBAT_Low bit in the battery-backed internal status register is asserted and not modified otherwise.

In normal use, the low battery event is not a PME wakeup event because the Low_Bat Enable bit will be cleared on VTR POR, when Low_Bat is set.

See Section 18.0, "Low Battery Detection," on page 196 for more details on the battery monitoring logic.

PINT

The parallel port interrupt defaults to ‘1b’ when the parallel port activate bit is cleared. When the parallel port is activated, PINT follows the ACK# input.

U2INT

This bit follows the interrupt bit from UART 2.

UINT1

This bit follows the interrupt bit from UART 1.

FINT

This bit follows the interrupt bit from Floppy Disk Controller.

EM_EVT1

This bit is set to 1 if the EC_WR bit (bit 0) of the Interrupt Source Register in the EM Interface is asserted.

EM_EVT2

This bit is set to 1 if any of the 16 bits in the Interrupt Source Register in the EM Interface is asserted and enabled by the corresponding bit in the Interrupt Enable Register in the EM Interface.

FW_EVT1

This bit is set to 1 if any of the bits in the Firmware Event Status Register in the Runtime Register block is asserted and enabled by the corresponding bit in the Firmware Event Enable Register.

19.3.12 SMI_STS2 REGISTER

SMI Status 2 register. The IO_SMI# pin is asserted if any bit in this register is 1 and the corresponding bit in the SMI_EN2 Register is also 1. All bits must be cleared at the source unless otherwise noted.

TABLE 19-13: SMI_STS2 REGISTER

HOST OFFSET	13h					8-bit	**HOST SIZE**	
POWER	VTR			000X01XXb			**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R/WC	R	R	R
BIT NAME	Reserved			Intrusion	WDT	Reserved (Note 19-2)	KINT	MINT

MINT

This bit follows the interrupt bit from Mouse port of the 8042 Logical Device.

KINT

This bit follows the interrupt bit from Keyboard port of the 8042 Logical Device.

WDT

This bit is set to 1 if a Watchdog Timer event is asserted when the watchdog times out.

This bit is cleared by a write of ‘1b’. Writes of ‘0b’ have no effect.

Intrusion

This bit is a copy of INTRUSION in the Intruder Register and follows its behavior. If enabled, an SMI is generated when INTRUSION is high. INTRUSION in the Intruder Register must be cleared in order for this bit to be cleared.

Note 19-2 This bit is reserved. Writes are ignored. Reads always return ‘1’.

19.3.13 SMI_STS3 REGISTER

SMI Status 3 register. All bits are cleared by writing with a 1. Writes of a 0 to any bit have no effect.

For each GPIO monitored by this register, a SMI is asserted if the GPIO pin value changes according to the Interrupt Detection field in the Pin Control Register associated with the GPIO. SMI events are only asserted if the Interrupt Detection field is set to an edge-triggered events. If the field is set to a level-sensitive event, no SMI will be generated even if the pin matches the selected level.

Note: If the Interrupt Detection field for one of the GPIO pins monitored by this register is changed from no-edge-detection to edge-triggered interrupts while the pin is high, the status bit will be set. Host software should clear the GPIO status bit in this register whenever it reconfigures the Interrupt Detection field in the Pin Control Register for the GPIO, in order to avoid a spurious event.

TABLE 19-14: SMI_STS3 REGISTER

HOST OFFSET	14h					8-bit	HOST SIZE	
POWER	VTR					00h	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	GPIO62	GPIO54	GPIO53	GPIO35	GPIO31	GPIO25	GPIO24	GPIO21

GPIO21

This bit is set to 1 if a wake event for GPIO21 is asserted.

GPIO24

This bit is set to 1 if a wake event for GPIO24 is asserted.

GPIO25

This bit is set to 1 if a wake event for GPIO25 is asserted.

GPIO31

This bit is set to 1 if a wake event for GPIO31 is asserted.

GPIO35

This bit is set to 1 if a wake event for GPIO35 is asserted.

GPIO53

This bit is set to 1 if a wake event for GPIO53 is asserted.

GPIO54

This bit is set to 1 if a wake event for GPIO54 is asserted.

GPIO62

This bit is set to 1 if a wake event for GPIO62 is asserted.

19.3.14 SMI_EN1 REGISTER

SMI Enable 1 register. This register is used to enable individual SMI sources from the SMI_STS1 Register onto the IO_SMI# pin. Bit fields correspond to the fields in the SMI_STS1 Register.

TABLE 19-15: SMI_EN1 REGISTER

HOST OFFSET	15h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	FW_ EVT1	EM_ EVT2	EM_ EVT1	FINT	U1INT	U2INT	PINT	Low_ Bat

19.3.15 SMI_EN2 REGISTER

SMI Enable 2 register. This register is used to enable individual SMI sources from the SMI_STS2 Register onto the IO_SMI# pin. Bit fields correspond to the fields in the SMI_STS2 Register except as noted.

TABLE 19-16: SMI_EN2 REGISTER

HOST OFFSET	16h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
BIT NAME	EN_SMI_ PIN	EN_SMI_ S	EN_SMI_ PME	Intrusion	WDT	Reserved	KINT	MINT

EN_SMI_PME

If this bit is 1, the Devint_Status bit in PME_STS2 Register is asserted if the SMI logic asserts the SMI function (the SMI_Status bit in the SMI_STS Register is 1 and the SMI_Enable bit in the SMI_EN Register is 1).

EN_SMI_S

If this bit is 1, the SMI logic is enabled onto serial IRQ if the SMI logic asserts the SMI function (the SMI_Status bit in the SMI_STS Register is 1 and the SMI_Enable bit in the SMI_EN Register is 1).

EN_SMI_PIN

If this bit is 1, the SMI logic is enabled onto the IO_SMI# pin if the SMI logic asserts the SMI function (the SMI_Status bit in the SMI_STS Register is 1 and the SMI_Enable bit in the SMI_EN Register is 1).

19.3.16 SMI_EN3 REGISTER

SMI Enable 3 register. This register is used to enable individual SMI sources from the SMI_STS3 Register onto the IO_SMI# pin. Bit fields correspond to the fields in the SMI_STS3 Register.

TABLE 19-17: SMI_EN3 REGISTER

HOST OFFSET	17h					8-bit	HOST SIZE	
POWER	VTR					00h	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO62	GPIO54	GPIO53	GPIO35	GPIO31	GPIO25	GPIO24	GPIO21

19.3.17 FORCE DISK CHANGE REGISTER

TABLE 19-18: FORCE DISK CHANGE REGISTER

HOST OFFSET	20h					8-bit	HOST SIZE	
POWER	VTR					03h 03h	nSYS_RST DEFAULT nSIO_RESET DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R/WS
BIT NAME	Reserved							FDC0

FDC0

Force Disk Change for FDC0. Setting this bit to '1b' forces the FDD DSKCHG# input active when drive 0 has been selected. This field can be written to a 1, but cannot be cleared by software. This bit is cleared on STEP# and DS0#.

0= Inactive

1= Active

Note 1: DSKCHG (FDC DIR Register, bit 7) = (DS0# AND FDC0) OR DSKCHG#.

2: This register is reset on VTR POR, VCC RESET and PCIRESET.

19.3.18 FLOPPY DATA RATE SELECT SHADOW REGISTER

This register is a readable copy of the write-only Floppy Data Rate Select register.

TABLE 19-19: FLOPPY DATA RATE SELECT SHADOW REGISTER

HOST OFFSET	21h					8-bit	HOST SIZE	
POWER	VTR					n/a	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R
BIT NAME	SRST	PD	Reserved	PRECOMP			DRS[1:0]	

DRS

Data Rate Select 0 and Data Rate Select 1

PRECOMP

PRECOMP 0, PRECOMP 1 and PRECOMP 2.

PD

Power Down

SRST

Soft Reset

19.3.19 UART 1 FIFO CONTROL SHADOW REGISTER

This register is a readable copy of the write-only UART 1 FIFO Control register.

TABLE 19-20: UART 1 FIFO CONTROL SHADOW REGISTER

HOST OFFSET	22h					8-bit	HOST SIZE	
POWER	VTR					n/a	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R
BIT NAME	RTM	RTL	Reserved		DMS	XFR	RFR	FE

FE

FIFO Enable

Rfr

Receiver FIFO Reset.

Xfr

Transmit FIFO Reset.

DMS

DMA Mode Select

RTL

Receiver Trigger (LSB)

RTM

Receiver Trigger (MSB)

19.3.20 UART 2 FIFO CONTROL SHADOW REGISTER

This register is a readable copy of the write-only UART 2 FIFO Control register.

TABLE 19-21: UART 2 FIFO CONTROL SHADOW REGISTER

HOST OFFSET	23h					8-bit	HOST SIZE	
POWER	VTR					n/a	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R
BIT NAME	RTM	RTL	Reserved		DMS	XFR	RFR	FE

FE

FIFO Enable

Rfr

Receiver FIFO Reset.

Xfr

Transmit FIFO Reset.

DMS

DMA Mode Select

RTL

Receiver Trigger (LSB)

RTM

Receiver Trigger (MSB)

19.3.21 DEVICE DISABLE REGISTER

The enable bits in this register disable access to LPC logical devices by overriding the Valid bit in the Base Address Registers for each device.

TABLE 19-22: DEVICE DISABLE REGISTER

HOST OFFSET	24h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R	R/W	R	R	R/W
BIT NAME	PP_ Disable	SP1_ Disable	SP2_ Disable	Reserved	Floppy_ Disable	Reserved		FWP

FWP

Floppy Write Protect

0=This bit has no effect: floppy write protection is controlled by the write protect pin or the forced write protect bit (bit 0 of register F1h in the Floppy Configuration space)

1= In this state, FWP overrides the write protect pin on the part and the forced write protect bit

WRTPRT# (to the FDC Core) = (DS0# AND Force Write Protect) OR (DS1# AND Force Write Protect) OR WRTPRT# (from the FDD Interface) OR Floppy Write Protect

Floppy Disable

Floppy Disable

0=No effect: FDC access controlled by the Valid bit in the FDC BAR.

1= Floppy access disabled

SP2_Disable

Serial Port 2 Disable.

0=No effect: Serial Port 2 access controlled by the Valid bit in the Serial Port 2 BAR.

1= Serial Port 2 access disabled

SP1_Disable

Serial Port Disable

0=No effect: Serial Port 1 access controlled by the Valid bit in the Serial Port 1 BAR.

1= Serial Port 1 access disabled

PP_Disable

Parallel Port Disable

0=No effect: Parallel Port access controlled by the Valid bit in the Parallel Port BAR.

1= Parallel Port access disabled

19.3.22LED REGISTER

This register controls the operation of the YELLOW# and GREEN# pins This register is read-only unless the LED controlled by the YELLOW# and GREEN# pins is off or the system is in the S0 state with power on (SLP_S3# and SLP_S5# both de-asserted and PWRGD_PS asserted).

TABLE 19-23: LED REGISTER

HOST OFFSET	25h					8-bit	**HOST SIZE**	
POWER	VTR			00000000b			**nSYS_RST DEFAULT**	
				00000xxxb			**VCC RESET DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				Code_ Fetch	Color	Blink	

Blink

Yellow/Green LED blink pattern

0=The LED is off

1= The LED blinks at a 1Hz rate with a 50% duty cycle (0.5 seconds on, 0.5 seconds off)

2=Reserved

3=The LED is on

This field is controlled by logic based on the SLP_S3#, SLP_S5# and PWRGD_PS pins. See Section 21.9, "LED Pins," on page 233.

Color

Affects the color of the LED1 and LED2. The action is state dependent; see Section 21.9, "LED Pins," on page 233.

Code_Fetch

This bit indicates BIOS progress.

0=BIOS has not reached code-fetch state

1= BIOS has reached code-fetch state

This bit is forced to 0 on VCC RESET.

Note:	When Code_Fetch is set to ‘1b’ by software, Color must be set to ‘1b’ as well.

19.3.23 KEYBOARD SCAN REGISTER

TABLE 19-24: KEYBOARD SCAN REGISTER

HOST OFFSET	26h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Scan_Code							

Scan_Code

Keyboard scan code, used in the Wake on Specific Key logic. See Section 11.6, "'Wake on Specific Key' Option," on page 117.

19.3.24 POWER GOOD REGISTER

TABLE 19-25: POWER GOOD REGISTER

HOST OFFSET	27h				8-bit	**HOST SIZE**	
POWER	VTR				F1h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R	R	R/W	R/W
BIT NAME	PCIRST_ OUT2_EN	PCIRST_ OUT1_EN	PCIRST_ OUT3_EN	PCIRST_ OUT4_EN	Reserved		PWRGD_ Lock	PWRGD_ Delay

PWRGD_Delay

Selects an optional delay for the PWR_GOOD_3V signal.

0=Select no delay for PWR_GOOD_3V

1= Select PWR_GOOD_3V delay (default)

PWRGD_Lock

0=No lock operation (default)

1= Bit[0] and bit[1] of this register become read-only. They remain read-only until a VTR POR

PCIRST_OUT4_EN

0=Pin PCI_RST_OUT4# forced low

1= Pin PCI_RST_OUT4# is a buffered copy of LRESET# (default)

PCIRST_OUT3_EN

0=Pin PCI_RST_OUT3# forced low

1= Pin PCI_RST_OUT3# is a buffered copy of LRESET# (default)

PCIRST_OUT1_EN

0=Pin PCI_RST_OUT3# forced low

1= Pin PCI_RST_OUT3# is a buffered copy of LRESET# (default)

PCIRST_OUT2_EN

0=Pin PCI_RST_OUT4# forced low

1= Pin PCI_RST_OUT4# is a buffered copy of LRESET# (default)

19.3.25 GPIO SELECT REGISTER

TABLE 19-26: GPIO SELECT REGISTER

HOST OFFSET	28h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved		GPIO_Select					

GPIO_Select

This field selects which GPIO is readable in the GPIO Read Register. Table 19-27, "Select to GPIO Mapping" shows the mapping between the value in GPIO_Select (in decimal) and the GPIO selected.

TABLE 19-27: SELECT TO GPIO MAPPING

Select	GPIO	Select	GPIO	Select	GPIO	Select	GPIO
0	GP000	16	GP020	32	GP040	48	GP060
1	GP001	17	GP021	33	GP041	49	GP061
2	GP002	18	GP022	34	GP042	50	GP062
3	GP003	19	GP023	35	GP043	51	GP063
4	GP004	20	GP024	36	GP044	52	GP064
5	GP005	21	GP025	37	GP045	53	GP065
6	GP006	22	GP026	38	GP046	54	GP066
7	GP007	23	GP027	39	GP047	55	GP067
8	GP010	24	GP030	40	GP050	56	GP070
9	GP011	25	GP031	41	GP051	57	GP071
10	GP012	26	GP032	42	GP052	58	N/A
11	GP013	27	GP033	43	GP053	59	N/A
12	GP014	28	GP034	44	GP054	60	N/A
13	GP015	29	GP035	45	GP055	61	N/A

TABLE 19-27: SELECT TO GPIO MAPPING (CONTINUED)

Select	GPIO	Select	GPIO	Select	GPIO	Select	GPIO
14	GP016	30	GP036	46	GP056	62	N/A
15	GP017	31	N/A	47	GP057	63	N/A

19.3.26 GPIO READ REGISTER

TABLE 19-28: GPIO READ REGISTER

HOST OFFSET	29h					8-bit	**HOST SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							GPIO_ In

GPIO_In

This field returns the value of the GPIO pin selected by GPIO_Select, if enabled.

19.3.27 GPIO WRITE REGISTER

This register is used to implement GPIO writes by the Host. If enabled by the GPIO Access Enable 1 Register and GPIO Access Enable 2 Register, writes to this register set bit GPIO_Write in the GIRQ22 Source register, which can generate an EC interrupt of GPIO_Write is enabled for interrupts.

By convention, the Host and the EC synchronize GPIO writes with this register. When processing the GPIO_Write interrupt, the EC copies bit D0 of this register into the GPIO specified in the GPIO_Select register. Once the write is complete, the EC sets this register to FFh. The Host should only write either GPIO_Select or GPIO_Write when GPIO_Write is FFh.

TABLE 19-29: GPIO WRITE REGISTER

HOST OFFSET	2Ah					8-bit	**HOST SIZE**	
POWER	VTR					FFh	**nSYS_RST DEFAULT**	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Write							

GPIO_Write

If enabled by GPIO_Select and GPIO Access Enable 1 Register and GPIO Access Enable 2 Register, writes to this register cause a GPIO_Write interrupt in the EC Interrupt Status Register GIRQ22.

19.3.28 FIRMWARE EVENT STATUS REGISTER

TABLE 19-30: FIRMWARE EVENT STATUS REGISTER

HOST OFFSET	30h					8-bit	HOST SIZE	
POWER	VTR					00h	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC	R/WC
BIT NAME	Firmware_Event							

Firmware_Event

If any of these bits is ‘1b’ and the corresponding bit in the Firmware Event Enable Register is set to ‘1b’, the Firmware Event bit (FWE) in the PME and SMI Status registers is set to 1.

Each bit in this field is cleared when written with a ‘1b’. The ability to clear the bit can be disabled by the EC.

The EC can generate an interrupt to the Host by setting any bit in this field to ‘1b’. The EC can set bits to ‘1b’.

19.3.29 FIRMWARE EVENT ENABLE REGISTER

TABLE 19-31: FIRMWARE EVENT ENABLE REGISTER

HOST OFFSET	31h					8-bit	HOST SIZE	
POWER	VTR					00h	nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Firmware_Event_Enable							

Firmware_Event_Enable

Each bit that is set to ‘1b’ in this field enables the generation of an SMI or PME event by the corresponding bit in the Firmware_Event field in the Firmware Event Status Register.

19.3.30 POWER RECOVERY MODES REGISTER

TABLE 19-32: POWER RECOVERY MODES REGISTER

HOST OFFSET	32h					8-bit	HOST SIZE	
POWER	VTR					00h	VBAT POR DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R
BIT NAME	PWR_ State	Reserved						

PWR_State

This bit saves the last state of the SLP_S3# input when the power supply goes off (that is, this bit is set to the state of SLP_S3# on the falling edge of the PWRGD_PS input). The bit is used to determine last state of the system prior to a power failure:

0: Power supply was off (SLP_S3# was asserted) (VBAT POR default)

1: Power supply was on (SLP_S3# was de-asserted)

19.3.31 INTRUDER REGISTER

See Section 20.0, "Intruder Detection Support," on page 223 for details on Intrusion Detection.

TABLE 19-33: INTRUDER REGISTER

HOST OFFSET	34h					8-bit	HOST SIZE	
POWER	VTR			See Note 19-3			nSYS_RST DEFAULT	
BYTE0 BIT	D7	D6	D5	D4	D3	D2	D1	D0
TYPE	R	R	R	R	R	R	R	R/WC
BIT NAME	Reserved						INTRD_ STS	INTRUSION

INTRUSION

When the INTRUDER# input goes high-to-low or low-to-high, this bit will be set. This bit is also set on VBAT POR. Software must write a ‘1’ to clear this bit. Writes of ‘0’ to this bit are ignored.

INTRD_STS

This bit indicates the current state of the INTRUDER# pin.

0: INTRUDER# pin is 0

1: INTRUDER# pin is 1

Note 19-3 The VTR POR default value of this register depends on the current value of INTRD_STS, which is derived directly from the INTRUDER# pin, and the VBAT-backed INTRUSION state bit. Possible combinations are:

00: No intrusion event; INTRUDER# pin currently 0

01: An intrusion event occurred; INTRUDER# pin currently 0

10: No intrusion event; INTRUDER# pin currently 1

11: An intrusion event occurred; INTRUDER# pin currently 1

19.4 EC-Only Registers

19.4.1 DEVICE DISABLE ENABLE REGISTER

TABLE 19-34: DEVICE DISABLE ENABLE REGISTER

HOST OFFSET	-					-	**HOST SIZE**	
EC OFFSET	100h					8-Bit	**EC SIZE**	
POWER	VTR					0001h	**nSYS_RST DEFAULT**	
BUS	Host SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Enable

Enable

If this bit is 1, the Device Disable Register is read-write. If this bit is 0, the Device Disable Register is read-only.

19.4.2 GPIO ACCESS ENABLE 1 REGISTER

This register enables Host access to GPIOs. Each bit corresponds to one GPIO in the range GP000 through GP036.

TABLE 19-35: GPIO ACCESS ENABLE 1 REGISTER

HOST OFFSET	-					-	**HOST SIZE**	
EC OFFSET	104h					32-Bit	**EC SIZE**	
POWER	VTR					0000_0000h	**nSYS_RST DEFAULT**	
BUS	Host SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved	GPIO_Enable[036:030]						
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[027:020]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[017:010							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[007:000]							

GPIO_Enable

If a bit in this register is 1, then the corresponding GPIO is accessible by the HOST in the GPIO Read Register and the GPIO Write Register. The bits in this register correspond to values in the GPIO Select Register from 0 through 31.

19.4.3 GPIO ACCESS ENABLE 2 REGISTER

This register enables Host access to GPIOs. Each bit corresponds to one GPIO in the range GP040 through GP071.

TABLE 19-36: GPIO ACCESS ENABLE 2 REGISTER

HOST OFFSET	-					-	**HOST SIZE**	
EC OFFSET	108h					32-Bit	**EC SIZE**	
POWER	VTR					0000_0000h	**nSYS_RST DEFAULT**	
BUS	Host SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R	R	R	R	R	R	R/W	R/W
BIT NAME	Reserved						GPIO_Enable [071:070]	
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[067:060]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[057:050							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	GPIO_Enable[047:040]							

GPIO_Enable

If a bit in this register is 1, then the corresponding GPIO is accessible by the HOST in the GPIO Read Register and the GPIO Write Register. The bits in this register correspond to values in the GPIO Select Register from 32 through 57.

19.4.4 EVENT SET REGISTER

TABLE 19-37: EVENT SET REGISTER

HOST OFFSET	-					-	**HOST SIZE**	
EC OFFSET	10Ch					8-Bit	**EC SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BUS	Host SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS	R/WS
BIT NAME	Event_Set							

Event_Set

This register provides the EC with a means of updating the Firmware Event Status Register. Writing a bit in this field with a ‘1b’ sets the corresponding bit in the Firmware Event Status Register to ‘1b’. Writing a bit in this field with a ‘0b’ has no effect. Reading this field returns the current contents of the Firmware Event Status Register.

19.4.5 HOST CLEAR ENABLE REGISTER

TABLE 19-38: HOST CLEAR ENABLE REGISTER

HOST OFFSET	-					-	**HOST SIZE**	
EC OFFSET	110h					8-Bit	**EC SIZE**	
POWER	VTR					00h	**nSYS_RST DEFAULT**	
BUS	Host SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Host_Clr_Enable							

Host_Clr_Enable

When a bit in this field is ‘0b’, the corresponding bit in the Firmware Event Status Register cannot be cleared by writes to the Firmware Event Status Register. When a bit in this field is ‘1b’, the corresponding bit in the Firmware Event Status Register can be cleared when that register bit is written with a ‘1b’.

This register is only accessible to the EC. These bits allow the EC to emulate whether the status bits in the Firmware Event Status Register are based on an edge or level firmware event.

20.0 INTRUDER DETECTION SUPPORT

A switch connected to the chassis cover indicates if the cover is on or off. When the cover is removed, the INTRUDER# input will transition from high-to-low or low-to-high depending on the type of switch used (normally open or normally closed).

Whenever the INTRUDER# input goes high-to-low or low-to-high, the INTRUSION bit is set in the Intruder Register in the Runtime Registers LPC Logical Device (Logical Device Ah) and in the Power-Fail and Reset Status Register in the Power, Clocks and Resets EC Logical Device (Logical Device 33h). The INTRUSION bit will remain set until cleared by software. This bit and input logic are powered by VBAT so that an intrusion condition is detected and stored even if VTR is removed. Reads and writes of the INTRUSION bit in the Intruder Register are directed to the INTRUSION bit in the Power-Fail and Reset Status Register.

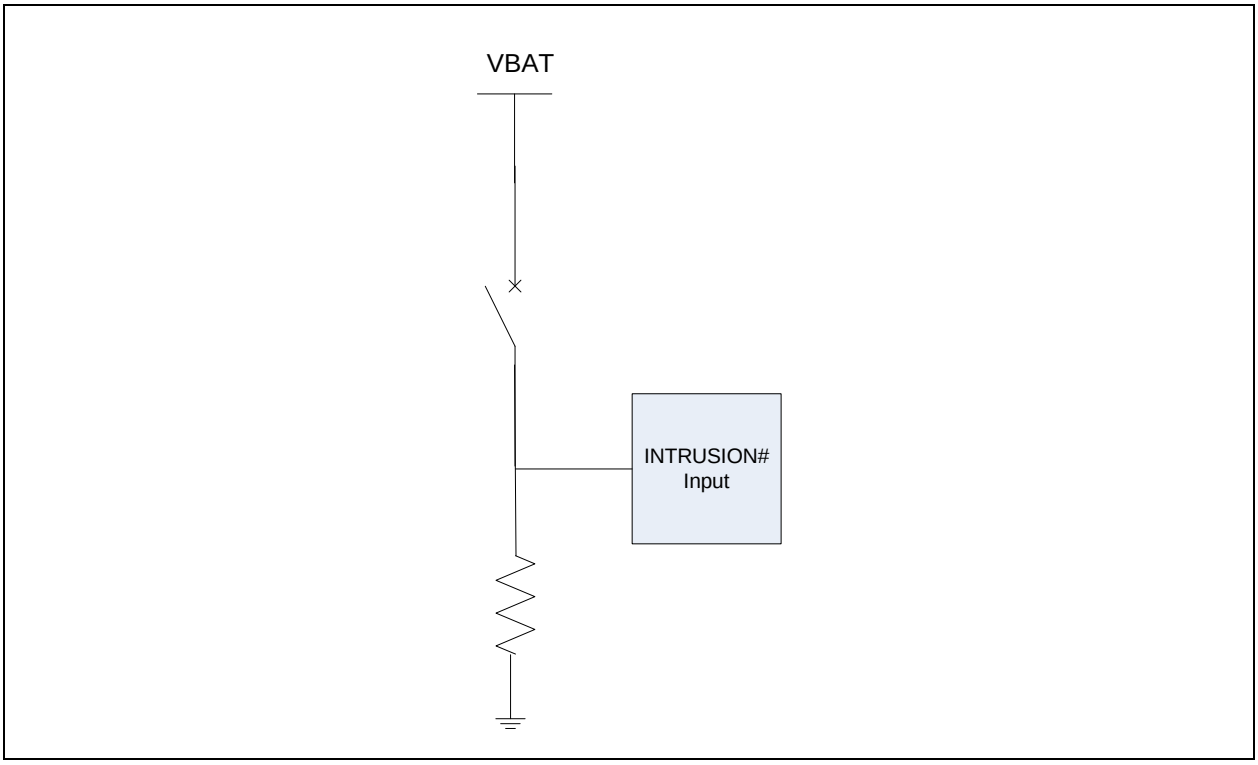
The INTRD_STS bit indicates the current state of the INTRUDER# pin. This bit is in the Intruder Register in the Runtime Registers LPC Logical Device (Logical Device Ah).

20.1 Intrusion Bit

An intrusion event occurs when there is any transition of INTRUDER# (low-to-high or high-to-low). Any intrusion event will set the INTRUSION bit and also changes the PME and SMI status bits.

For minimal current drain on the battery, the recommended use of this pin is with a normally open switch as shown in Figure 20-1, "Recommended Intruder Pin Connection". When the cover is closed the input will be externally pulled down to ground. When the cover is opened this input will be pulled up to VBAT.

FIGURE 20-1: RECOMMENDED INTRUDER PIN CONNECTION



The INTRUSION bit will default to ‘1’ on VBAT POR (battery removed and replaced or battery voltage below approximately 1.2V on VTR POR). The INTRUSION bit will therefore be set to ‘1’ if an intrusion event occurs or if a VBAT POR occurs.

Writing ‘1’ to the INTRUSION bit will clear it, regardless of the state of the INTRUDER# pin. Writing ‘0’ to the INTRUSION bit has no effect.

20.2 PME and SMI Generation

The SMI and PME status bits for intrusion detection are set under VCC power, VTR power or on VTR POR, as they “shadow” the INTRUDER# bit. The SMI and PME status bits are cleared on a write of '1'. These bits cannot be cleared until the INTRUDER# pin goes low.

These bits function in one of three cases:

Case 1. An intrusion occurs under battery power only or a VBAT POR occurs. In this case, the event will be latched under battery power and the INTRUSION PME and SMI status bits will be set when VTR returns. Therefore, the PME and SMI status bits will have two possible default values on VTR POR, depending on whether or not the intrusion event occurred under battery power. When VTR returns, no enable bits are set, so there will be no PME or SMI generated. When VCC goes active, and the OS sets the enable bits, a PME and/or SMI will be generated. If the corresponding PME enable bit is set, a PME will be generated under VCC power. If the corresponding SMI enable bit is set, an SMI will be generated under VCC power. Therefore, in this case, setting the enable bit (low-to-high edge) will trigger the generation of the PME and SMI.

Case 2. An intrusion occurs under VTR power (VCC=0). In this case, the INTRUSION PME and SMI status bits will be set. If the corresponding PME enable bit is set, a PME will be generated under VTR power. If the corresponding SMI enable bit is set, an SMI will be generated under VTR power. In this case, setting the status bit (low-to-high edge) will trigger the generation of the PME and SMI.

Case 3. An intrusion occurs under VCC power. In this case, the INTRUSION PME and SMI status bits will be set. If the corresponding PME enable bit is set, a PME will be generated under VCC power. If the corresponding SMI enable bit is set, an SMI will be generated under VCC power. In this case, setting the status bit (low-to-high edge) will trigger the generation of the PME and SMI.

21.0 GLUE LOGIC HARDWARE

21.1 General Description

The Glue Logic Hardware provides several special purpose hardware blocks. These blocks include functions for:

- LRESET# buffering
- Power Supply Turn On
- PWR_GOOD_3V Signal
- Resume Reset Signal Generation
- PWRBTN# Debounce
- Backfeed Cut Functionality
- LED Pins
- Performance Timer

21.2 Power, Clocks and Reset

21.2.1 RESET

Glue logic is reset when nSYS_RST is asserted.

21.3 LRESET# buffering

The LRESET# input (Pin 16) can be used as an alternate function for a number of GPIO pins. When the LRESET# alternate function is selected, the GPIO pin follows the state of the LRESET# pin, and thus functions as a buffered copy of LRESET#.

The PCIRST_OUT4#, PCIRST_OUT3#, PCIRST_OUT2# and PCIRST_OUT1#functions are each associated with an enable bit in the Power Good Register located in Runtime Register offset 05h. The bits are VTR powered bit, and default to enabled (‘1’) on VTR POR. The bits operate as follows when the pins are configured for the LRESET# buffer function:

- When the associated enable bit is set to ‘1’, the pins will operate as in Table 21-1.
- When the associated enable bit is cleared to ‘0’, the pins will be low (‘0’).

TABLE 21-1: BUFFERED PCI RESET TRUTH TABLE

Input	Enable	Output
PCI_RESET#	**PCIRST_OUT1_EN, PCIRST_OUT2_EN PCIRST_OUT3_EN, PCIRST_OUT4_EN**	**PCIRST_OUT1#, PCIRST_OUT2# PCIRST_OUT3#, PCIRST_OUT4#**
X	0	0
0	1	0
1	1	1

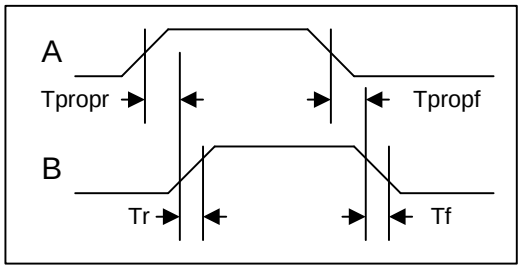
When VTR power is applied, and VCC RESET is asserted, the buffered LRESET# outputs are low.

The timing values for the PCIRST_OUT4#, PCIRST_OUT3#, PCIRST_OUT2# and PCIRST_OUT1# signals are shown in Table 21-2, "Buffered LRESET# Timing". This values reference Figure 21-1.

TABLE 21-2: BUFFERED LRESET# TIMING

Name	Description	MIN	TYP	MAX	Units
Tr	Buffered LRESET# signal low to high rise time. Measured from 90% to 10%			53	ns
Tpropf	Buffered LRESET# signal low to high propagation time. Measured from LRESET# to Buffered LRESET# signal.			30	ns
CO	Output Capacitance			25	pF
CL	Load Capacitance			40	pF

FIGURE 21-1: LRESET# RISE, FALL AND PROPAGATION TIMING



21.4 Power Supply Turn On

The PS_ON# signal is used to turn on the power supply. It is a function of SLP_S3# according to the truth table below.

The PS_ON# is used as the power down signal for the power supply. Since PS_ON# is an open drain output, it will need to be pulled up external to the chip if such a pull-up is not provided on the power supply. The power supply turn-on circuit behaves according to the table below.

TABLE 21-3: PS_ON# TRUTH TABLE

Input	Output
SLP_S3#	**PS_ON#**
0	Hi-Z
1	0

21.5 PWR_GOOD_3V Signal

The PWR_GOOD_3V signal has a selectable delay to insert delay from ACPI power sequencing events to software runtime.

- **Negative edge (S0->S3/S5):** The 1-0 transition of SLP_S3# input or the 1-0 transition (or 0 level) of PWRGD_PS input would cause an immediate 1-0 transition (or 0 level) of PWR_GOOD_3V.
- **Positive edge (S3/S5->S0):** The 0-1 transition of PWRGD_PS input would cause a 0-1 transition of PWRGD_3V. The PWR_GOOD_3V transition is either immediate (no delay) or after a 100 ms (min) to 120 msec (max) delay from the 0-1 transition of PWRGD_PS.
- The delay is optional and will be governed by a lockable select bit in the PWRGD DELAY register (Runtime Register at offset 51h). Default operation selects the delay. An internal delay indicator signal is used to indicate whether the 100-120 msec delay time has elapsed.
- All affected pins will retain the same electrical characteristics as they have now.
- PWR_GOOD_3V is forced to 0 and glitch-protected while VTR is rising.
- PWR_GOOD_3V is always 0 when the Resume Reset signal is asserted.

TABLE 21-4: PWR_GOOD_3V OUTPUT

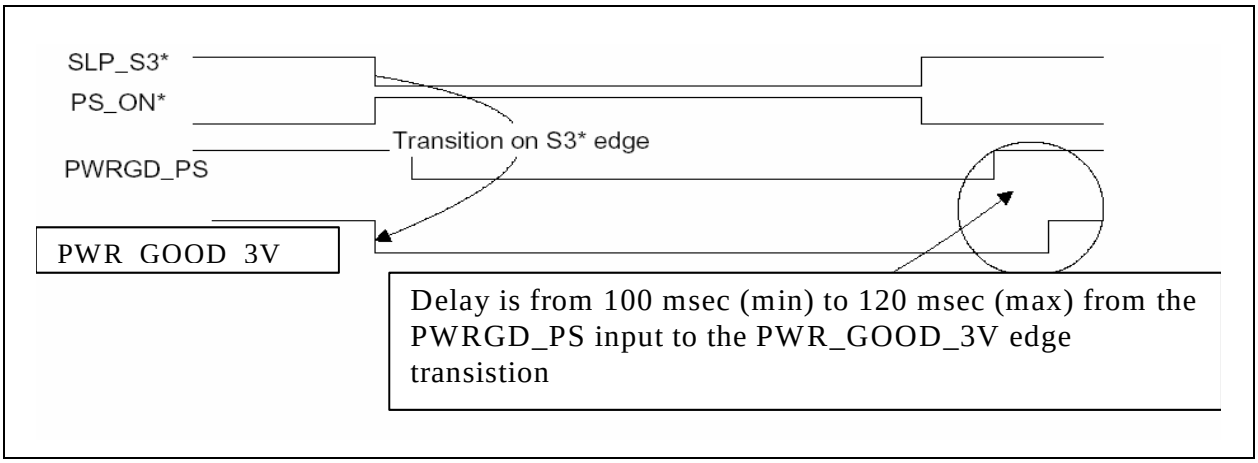
RSMRST#	SLP_S3#	PWRGD_PS	PWR_GOOD_3V Select Bit	Internal Delay Signal	PWR_GOOD_3V
0	X	X	X	X	0
1	1-0 transition or 0 level	X	X	X	0
1	X	1-0 transition or 0 level	X	X	0

TABLE 21-4: PWR_GOOD_3V OUTPUT (CONTINUED)

RSMRST#	SLP_S3#	PWRGD_PS	PWR_GOOD_3V Select Bit	Internal Delay Signal	PWR_GOOD_3V
1	1	0-1 transition	0	X	1 (no delay)
1	1	0-1 transition	1	0	0 (delay time not elapsed)
1	1	0-1 transition	1	1	1 (after 100-120 msec delay)

A timing diagram for this change is shown following:

FIGURE 21-2: PWR_GOOD_3V GENERATION



21.5.1 BITS FOR SELECTING AND LOCKING DELAY

Bits[1:0] in the Power Good Register at offset 27h of the Runtime Registers are used for selecting and locking the delay. The select bit (Bit[0]) selects the delay option as shown in Table 21-4. The LOCK bit (Bit[1]) sets bits[1:0] to be Read Only. These bits remain RO until a VTR POR.

21.6 Resume Reset Signal Generation

The RSMRST# signal is the reset output for the resume well power supply. This signal is used as a power on reset signal for the ICH.

RSMRST# is asserted by hardware and is also asserted and de-asserted by firmware. Hardware will not de-assert RSMRST#. EC Firmware controls the output level of RSMRST#. RSMRST# is forced to 0 and glitch-protected while VTR is rising.

The timing for the resume reset signal is given in Figure 21-3 and Table 21-5. The rising edge of RSMRST# is a delayed 3.3V buffered copy of VTR. This delay, t_{RESET_DELAY}, starts when VTR hits the trip point, V_{TRIP}, and is dependent on internal firmware. The delay is not programmable. Note the RSMRST# will be inactive high after the t_{RESET_DELAY} only if VTR (3.3V) is present. Otherwise, RSMRST# will be active low beyond the t_{RESET_DELAY} – until VTR (3.3V) goes active. On the falling edge there is minimal delay, t_{RESET_FALL}. Note that V_{TRIP} shown in Figure 21-3 has a V_{TRIP_MIN} and a V_{TRIP_MAX}.

FIGURE 21-3: RESUME RESET SEQUENCE

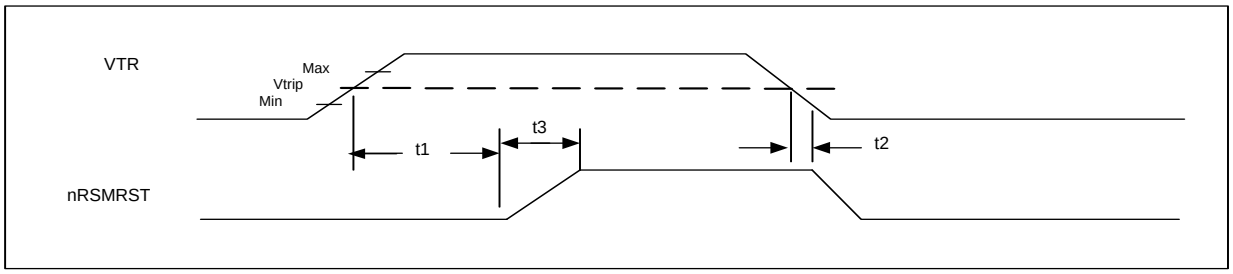


TABLE 21-5: RESUME REST TIMING

Name	Description	MIN	TYP	MAX	Units	Notes
t1	Treset delay. VTR active to RSMRST# inactive	20	30	100	msec	
t2	Treset_fall. VTR below trip voltage to RSMRST# active (Glitch width allowance)			100	nsec	
t3	Treset_rise			100	nsec	
V_{TRIP}	VTR low trip voltage	2.1	2.4	2.7	V	21-1

Note 21-1 The trip point can vary between these limits on a per part basis, but on a given part it should remain relatively stable.

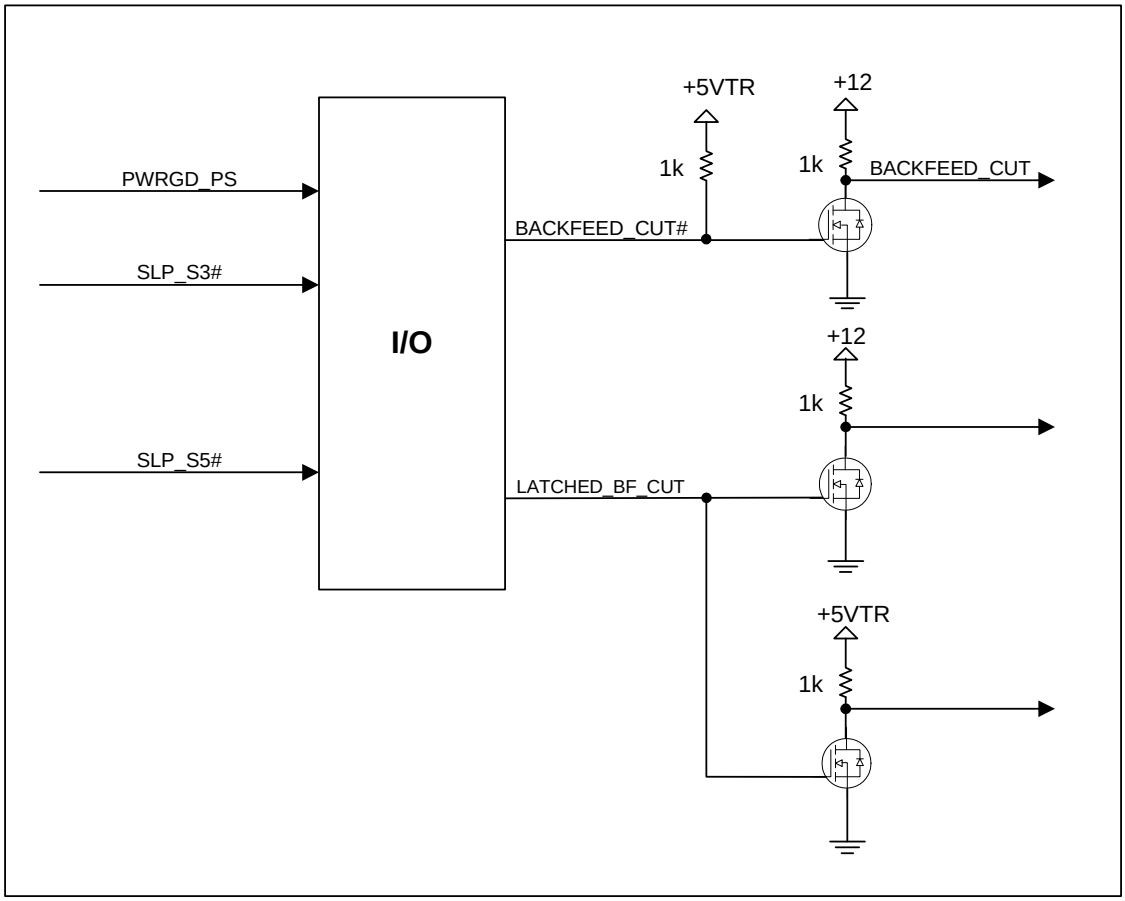
21.7 Backfeed Cut Functionality

BACKFEED_CUT# is a signal required by the S3 power state circuitry and is powered by the VTR supply. It is a function of PWRGD_PS and SLP_S3# according to Table 21-6, "BACKFEED_CUT# Truth Table". BACKFEED_CUT# is used to switch between the main voltage regulator and the suspend voltage regulator for various sub-systems when the system is transitioning into the S3 power state.

TABLE 21-6: BACKFEED_CUT# TRUTH TABLE

Inputs		Output
PWRGD_PS	**SLP_S3#**	**BACKFEED_CUT#**
0	0	Hi-Z
0	1	Hi-Z
1	0	Hi-Z
1	1	0

FIGURE 21-4: BACKFEED CUT AND LATCHED BACKFEED CUT CIRCUIT



The LATCHED_BF_CUT is generated from BACKFEED_CUT# and SLP_S5#. It is powered by VTR. As shown in Table 21-7, "LATCHED_BF_CUT Truth Table", LATCHED_BF_CUT is unconditionally low if either SLP_S5# or BACKFEED_CUT# is low. LATCHED_BF_CUT will only go high on the rising edge of BACKFEED_CUT# if SLP_S5# is already high. Once high, LATCHED_BF_CUT goes low as soon as SLP_S5# or BACKFEED_CUT# goes low.

TABLE 21-7: LATCHED_BF_CUT TRUTH TABLE

Inputs		Output
BACKFEED_CUT# (Internal Signal)	**SLP_S5#**	**LATCHED_BF_CUT**
0	0	0
0	1	0
1	0	0
0 to 1 (rising edge)	1	1
'1' and no rising edge	1	No Change (21-2)

Note 21-2 This is the condition when BACKFEED_CUT# stays high and SLP_S5# goes low and then high again (see Figure 21-4).

APPLICATION NOTE: The figure below shows the power up sequence. The BACKFEED_CUT# signal follows the power rail up to its final value. The LATCHED_BF_CUT signal stays low and never turns on. The SLP_S5# goes to its high value when the power rails have stabilized, approximately 25msec after power on.BACKEED_CUT# is pulled low a period t1 after SLP_S5# goes high. The period t1 can be as short as 1msec. Typical measured values are approximately

200msec. The t1 and t2 values are maintained by the inherent design of the system and are not controlled by the SCH5636.

FIGURE 21-5: LATCHED BACKFEED CUT POWER UP SEQUENCE

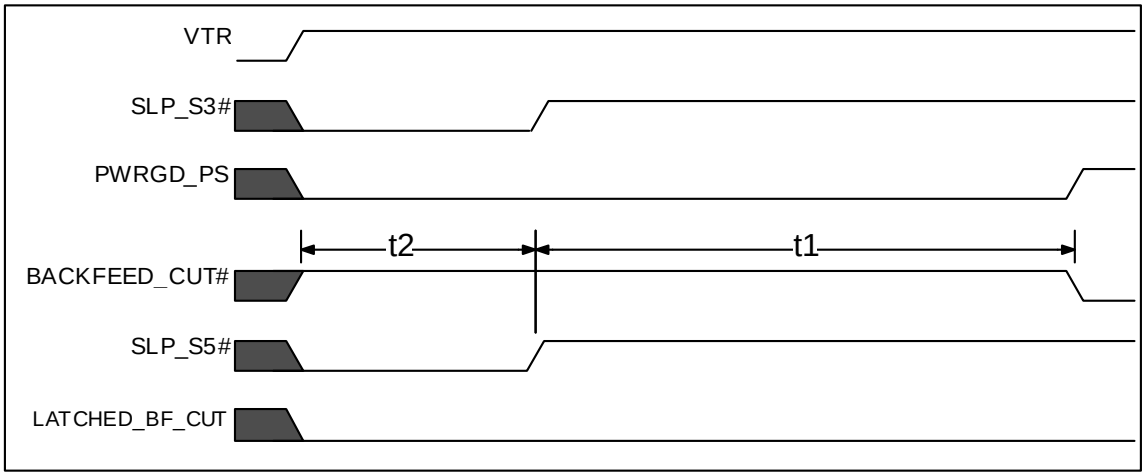
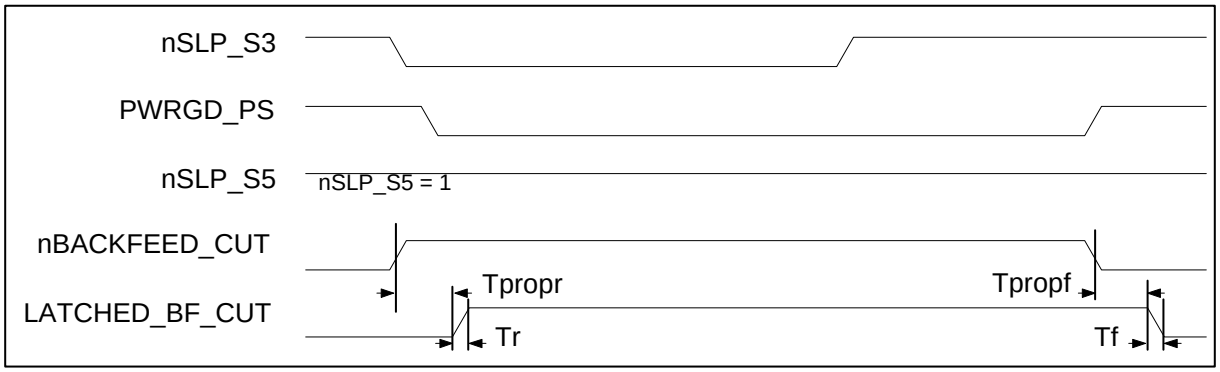


TABLE 21-8: LATCHED BACKFEED CUT POWER UP SEQUENCE TIMING

Name	Description	MIN	TYP	MAX	Units
t1	SLP_S5# inactive to BACKFEED_CUT# active	1	200		msec
t2	SLP_S5# inactive after power rails have stabilized		25		msec

APPLICATION NOTE: There are two possible timing sequences following the power up signal sequencing. The first possible sequence is with SLP_S5# staying high and BACKFEED_CUT# transitioning from low to high, remaining high for an undetermined period and then going back to low. At this point, the system returns to the end of the power-up sequence. During these BACKFEED_CUT# transitions, the propagation delays, rise and fall times for LATCHED_BF_CUT are as described in the figure below. The first sequence can start at the end of the power-up sequence at any time.

FIGURE 21-6: LATCHED BACKFEED CUT SEQUENCE 1



APPLICATION NOTE: The second possible sequence, shown in the figure below, is a normal powerdown sequence. The BACKFEED_CUT# signal goes from low to high when SLP_S3# goes low, and SLP_S5# goes from high to low 30usec to 65usec (t3) later. The LATCHED_BF_CUT signal goes high when BACKFEED_CUT# goes high and then LATCHED_BF_CUT returns

to low when SLP_S5# goes low. The BACKFEED_CUT# stays high and SLP_S5# stays low for an indeterminate time and then SLP_S5# will go high. A minimum of 1msec (t4) later, BACKFEED_CUT# will go low and the system returns to the end of the power-up sequence when SLP_S3# and PWRGD_3V goes high. Typical measured values of t4 are approximately 250msec. During all transitions, the propagation delays, rise and fall times and power regulation times for LATCHED_BF_CUT are as described in Figure 21-7. The first sequence can start at the end of this power-up sequence at any time.

FIGURE 21-7: LATCHED BACKFEED CUT SEQUENCE 2

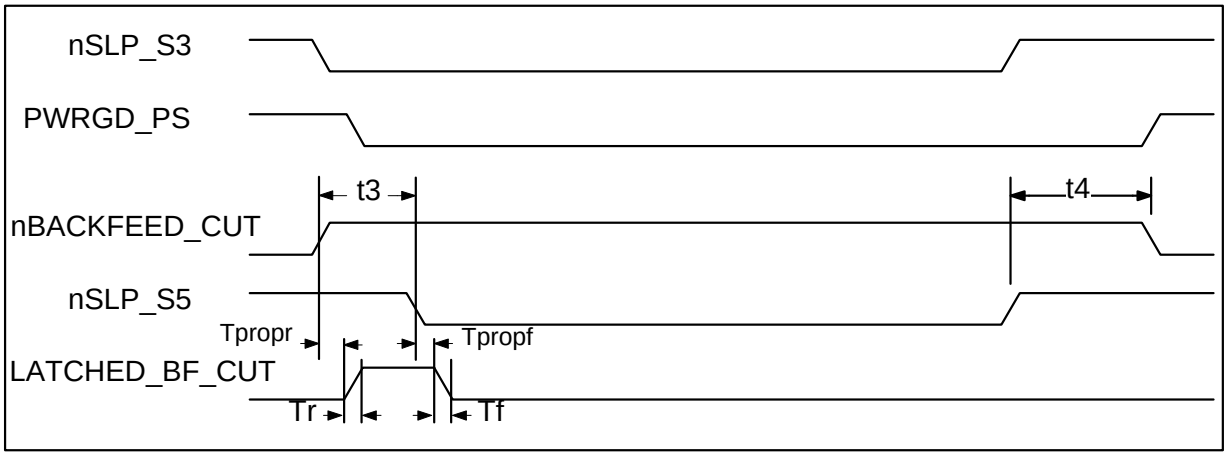
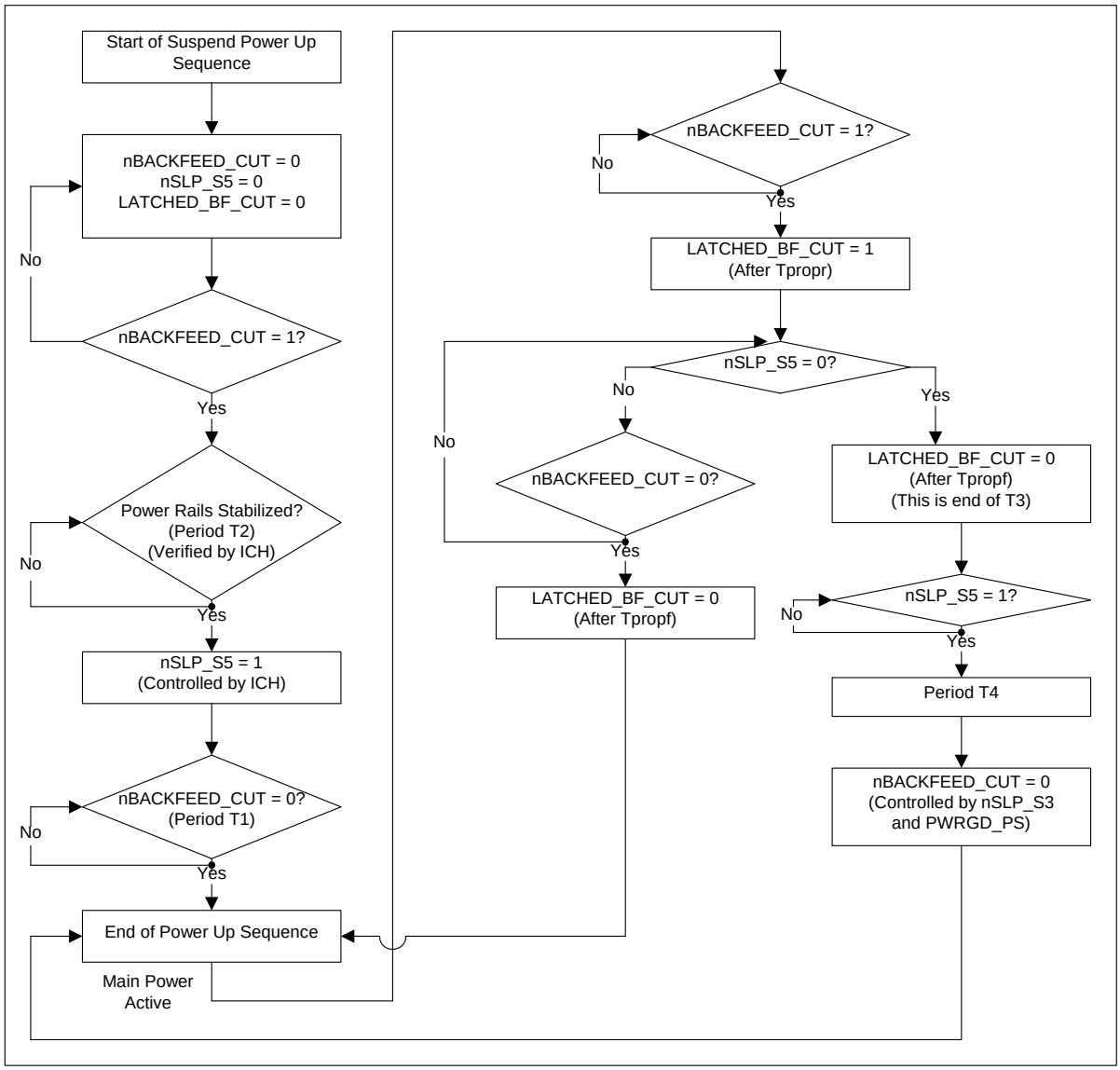


TABLE 21-9: LATCHED BACKFEED CUT SEQUENCE 1 AND 2 TIMING

Name	Description	MIN	TYP	MAX	Units
Tr	LATCHED_BF_CUT rise time. Measured from 10% to 90%.			1	us
Tf	LATCHED_BF_CUT fall time. Measured from 90% to 10%.			1	us
Tpropf	LATCHED_BF_CUT high to low propagation delay. Measured from BACKFEED_CUT#/SLP_S5# threshold to 90% of LATCHED_BF_CUT			50	ns
Tpropr	LATCHED_BF_CUT low to high propagation delay. Measured from BACKFEED_CUT#/SLP_S5# threshold to 10% of LATCHED_BF_CUT			50	ns
CO	Output Capacitance			25	pF
CL	Load Capacitance			50	pF
t3	BACKFEED_CUT# inactive to SLP_S5# active	30		60	us
t4	SLP_S5# inactive to BACKFEED_CUT# active	1	250		ms

APPLICATION NOTE: The following figure shows a flowchart of the logic.

FIGURE 21-8: LATCHED BACKFEED CUT FLOWCHART



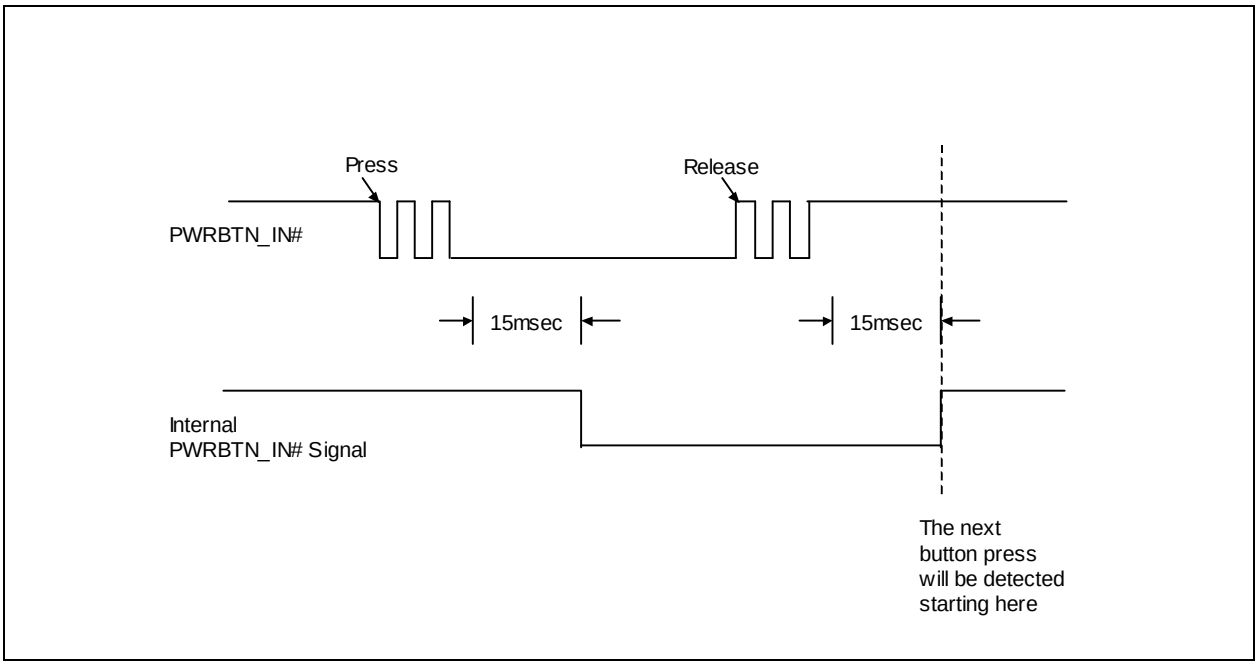
21.8 PWRBTN# Debounce

The power button, multiplexed with GP011, is debounced internally. The debounce will be implemented by sampling the PWRBTN# signal following the first high-to-low edge on the PWRBTN# pin to determine when the PWRBTN# signal has been stable low for 15msec (min), at which point the internal PWRBTN# signal will go active low. The PWRBTN# signal is then sampled to determine when the PWRBTN# signal returns to a logic '1' for 15msec (min), at which point the next high-to-low edge on the PWRBTN# signal can be detected. Any other high-to-low edges are ignored until the PWRBTN# signal has returned to a logic '1' for 15msec (min). This is to prevent false triggering due to noise on the power button and due to oscillations from the button following press and release.

The 15ms period is determined by the 32.768KHz clock sourced from the CLK32 pin. When the external clock is not available, the period is determined by the 32KHz clock derived from the internal ring oscillator.

The power button timing signal is shown below:

FIGURE 21-9: POWER BUTTON TIMING



21.9 LED Pins

The LED1#/GP006 and LED2#/GP007 functions on pin 25 and pin 26 can be used to control two LEDs or a Bi-Color LED. These pins default to LED functions with inverted, open-drain outputs (active low). The GPIO Configuration registers for GP006 and GP007 will have default values of inverted outputs on VTR POR. The LED pins can be configured to control a single bi-color LED or separate LEDs.

When connected to a Bi-Color LED, the two LED pins can configure the LED to be on or blinking in either of two colors. In this section, the colors will be called Color1 and Color2.

The LED pin functions control the LED as follows:

- The LED1# pin is low/blinking, the LED2# pin is high: LED is on/blinking, Color1.
- The LED2# pin is low/blinking, the LED1# pin is high: LED is on/blinking, Color2.
- Both the LED1# and LED2# pins are high: LED is off.

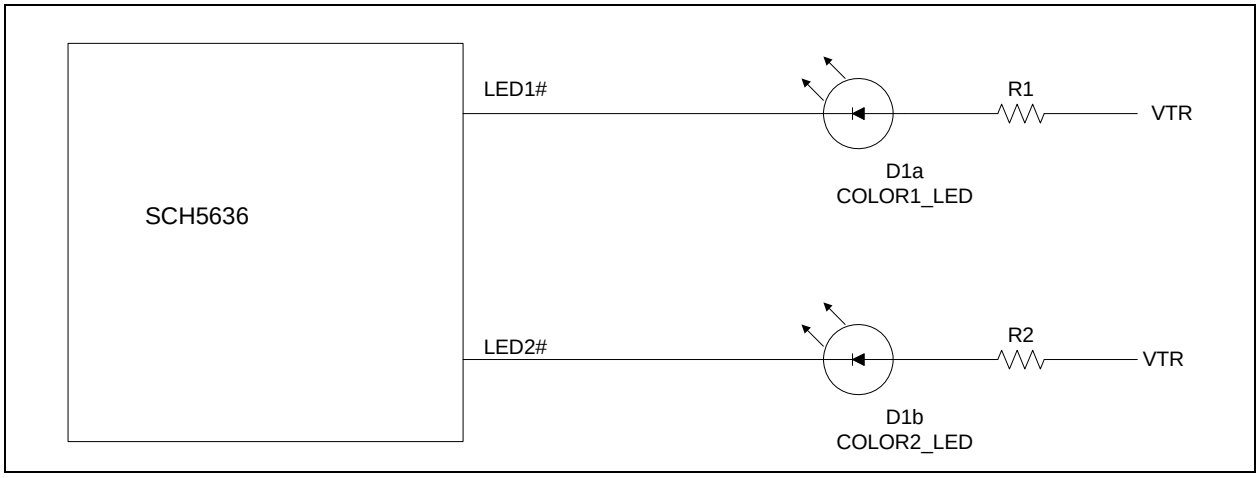
TABLE 21-10: SUMMARY OF BEHAVIOR OF LED PINS

State of LED1# Pin	State of LED2# Pin	State of LED
Low	High	On – Color1
Blinking	High	Blinking - Color1
High	Low	On – Color2
High	Blinking	Blinking – Color2
High	High	Off

Note that the polarity bit in the GP006 and GP007 Configuration registers affect the polarity of the LED1# and LED2# pin functions, respectively. The description below assumes inverted polarity (default).

The following figure shows the LED logic on the motherboard that will be used with the LED1# and LED2# pins. Note: This circuit is subject to change based on the polarity of the LED pins.

FIGURE 21-10: LED1 AND LED2, MOTHERBOARD LOGIC ILLUSTRATION



The SLP_S5#, SLP_S3#, PWRGD_PS pins and the Code-Fetch Bit are inputs to the LED logic circuit. The conditions for different LED1# and LED2# state outputs are described below. Depending on the conditions below, the LED1# and LED2# pins will be either steady ON, OFF or will Blink at 1Hz with 50% duty cycle (500ms on, 500ms off). The LED logic circuit controls the state of the LEDs by controlling bits[3:0] in the LED Register in the Runtime Register Logical Device. In addition, under certain conditions software can write these bits to control the LEDs. The ON, OFF and Blink selection is done via Bits[1:0] in the LED Register. The Color of the LEDs are programmable via Bits[3:2] of the LED Register. See Table 21-12 on page 235.

The LED Register is only writable when the system is the fully on state (the fully on state occurs in the S0 state with the PWRGD_PS signal active high). Writing the LED Register in other states may result in indeterminate results. Table 21-11 summarizes the Bi-Color LED functionality. Bits[2:0] of the LED register are controlled by the LED state machine when in the S3 state, in the S0 state when PWRGD_PS is low, and on transitions of S5 and PWRGD_PS. In other states, Bits[2:0] are set by software writing the LED register.

TABLE 21-11: LED STATES

State	Inputs				LED Register Bits[2:0]		Outputs		
	SLP_S5#	SLP_S3#	PWRGD_PS	LED Reg Bit[3] Code-fetch (21-4)	LED Reg Bit[2]	LED Reg Bit[1:0]	LED1# Pin	LED2# Pin	LED State
S5 (21-5)	0	X	X	X	0	00	1	1	LED1: Off LED2: Off
S3, no PWRGD_PS	1	0	0	X	1	01	1	1Hz, 50% duty cycle[3]	LED1: Off LED2: Blinking
S3, PWRGD_PS (21-3)	1	0	1	X	Prev	Prev	Prev	Prev	Previous State
S0, no PWRGD_PS	1	1	0	X	0	01	1Hz, 50% duty cycle[3]	1	LED1: Blinking LED2: Off

TABLE 21-11: LED STATES (CONTINUED)

State	Inputs				LED Register Bits[2:0]		Outputs		
	SLP_S5#	SLP_S3#	PWRGD_PS	LED Reg Bit[3] Code-fetch (21-4)	LED Reg Bit[2]	LED Reg Bit[1:0]	LED1# Pin	LED2# Pin	LED State
S0, PWRGD_PS, Code-Fetch = 0 (21-5)	1	1	1	0	0	11	0	1	LED1: Steady LED2: Off
S0, PWRGD_PS, Code Fetch = 1 (21-5)	1	1	1	1 (21-6)	1 (21-6)	11 (21-6)	1	0	LED1: Off LED2: Steady

Note 21-3 This entry provides for the possibility of a delay from SLP_S3# active to PWRGD_PS inactive.

Note 21-4 The Code-fetch bit is reset on VCC RESET.

Note 21-5 LED Register writable in this state.

Note 21-6 **The COLOR bit (LED Reg Bit[2]), the Code Fetch bit (LED Reg Bit[3]) and the two Blink bits (LED Reg Bits[1:0]) must all be set to 1 in this state to change the LED to** Color2**.** This is one of the states where the LED register bits is writable.

In the states wherein the LED register is writable, the LED pins are controlled by the LED register bits[3:0]. The Code Fetch bit (bit[3]) and the Color bit (bit[2]) must be set to the same value, either both 1 or both 0. The LED pins are controlled by bits [3:0] of the LED register as defined in Table 21-12.

TABLE 21-12: LED STATES RESULTING FROM WRITING LED REGISTER

LED Register Bits				LED State
Bit[3]	Bit[2]	Bit[1]	Bit[0]	
0	0	0	0	Color1: Off; Color2: Off
0	0	0	1	Color1: Blinking; Color2: Off
0	0	1	0	Color1: Off; Color2: Off
0	0	1	1	Color1: On; Color2: Off
0	1	0	0	Color1: Off; Color2: Off (21-7)
0	1	0	1	Color1: Off; Color2: Blinking (21-7)
0	1	1	0	Color1: Off; Color2: Off (21-7)
0	1	1	1	Color1: On; Color2: Off (21-7)
1	0	0	0	Color1: Off; Color2: Off (21-7)
1	0	0	1	Color1: Blinking; Color2: Off (21-7)
1	0	1	0	Color1: Off; Color2: Off (21-7)
1	0	1	1	Color1: Off; Color2: On (21-7)
1	1	0	0	Color1: Off; Color2: Off
1	1	0	1	Color1: Off; Color2: Blinking
1	1	1	0	Color1: Off; Color2: Off
1	1	1	1	Color1: Off; Color2: On

Note 21-7 Bits[3:2] should be set to the same value. These LED states should never be entered due to these bit settings. They are included for completeness.

21.10 Performance Timer

The SCH5636 contains a 32-bit counter that can be incremented every 1µS, which can be used for firmware timing analysis. The counter can time periods up to 4,295 seconds. The counter is located in the Performance Timer Register.

Software can set the counter to 0 at any time in order to start a timing interval. If the No_Halt bit is set, counting is suppressed when the EC is halted, so accurate timing can be maintained even when the CPU is halted by a debugger.

21.11 Glue Registers

Registers for Glue Logic reside in the EC-only space of the Runtime Register block. The base address for the Glue Logic block in the AHB address space is listed in Table 21-13, "Glue Logic Hardware Base Address Table"

TABLE 21-13: GLUE LOGIC HARDWARE BASE ADDRESS TABLE

Glue Logic Hardware **Instance**	**LDN**	**AHB Base Address**
GLUE LOGIC (IN RUNTIME REGISTERS)	**0Ah**	FF_2800h

Table 21-13, "Glue Logic Hardware Base Address Table", summarizes the registers allocated for the Glue Logic. The offset field in the following table is the offset from the AHB Base Address defined in **Table 21-13**.

TABLE 21-14: GLUE LOGIC HARDWARE REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Resume Reset Register	-	180h	32	R/W	
Signal Monitor Register	-	184h	32	R	
Performance Timer Register	-	188h	32	R/W	
Counter Control Register		18Ch	32	R/W	

21.11.1 RESUME RESET REGISTER

TABLE 21-15: RESUME RESET REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	180h					8-bit	**EC SIZE**	
POWER	VTR					00h	**VTR POR DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Resume_ Reset

Resume_Reset

When this bit is set to 1, the Resume Reset pin (RSMRST#) is set to one, de-asserting resume reset. Software can also set this bit to 0, forcing an assertion of RSMRST#. Hardware automatically asserts RSMRST# on the initial application of VTR, as well as on the falling edge of VTR, as described in Section 21.6, "Resume Reset Signal Generation," on page 227.

21.11.2 SIGNAL MONITOR REGISTER

TABLE 21-16: SIGNAL MONITOR REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	184h					8-bit	**EC SIZE**	
POWER	VTR					00h	**VTR POR DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved						PS_ON_ Internal	PWRBTN#

PWRBTN#

This bit reflects the output of the deglitch filter on the PWRBTN# input.

PS_ON_Internal

This bit reflects the internal PS_ON# circuitry. It is 0 if the internal circuitry is asserting PS_ON#, and 1 if the internal circuitry is not asserting it. Software can read the value of the PS_ON# pin by monitoring its associated GPIO and therefore determine whether the power supply has been turned on by internal or external circuitry.

21.11.3 PERFORMANCE TIMER REGISTER

TABLE 21-17: PERFORMANCE TIMER REGISTER

HOST ADDRESS	N/A			N/A			**HOST SIZE**	
EC OFFSET	188h			32-bit			**EC SIZE**	
POWER	VTR			0000_0000h			**nSYS_RST DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Count[31:24]							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Count[23:16							
BYTE 1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Count[15:8]							
BYTE 0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	–	–	–	–	–	–	–	–
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Count[7:0]							

Count

This register contains the current value of a free-running 32-bit counter. The counter is incremented at a 1MHz rate. When the timer reaches FFFF_FFFFh it rolls over to 0000_0000h. Incrementing can be suppressed based on control bits in the Counter Control Register.

Firmware can update this register at any time. Updates must be performed with a 32-bit store instruction.

21.11.4 COUNTER CONTROL REGISTER

TABLE 21-18: COUNTER CONTROL REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	18Ch					8-bit	**EC SIZE**	
POWER	VTR					00h	**VTR POR DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R/W	R/W
BIT NAME	Reserved						No_ Halt	Enable

Enable

This bit controls incrementing the Performance Timer Register.

0: The Performance Timer Register does not increment. The timer register retains its value.

1: Incrementing of the Performance Timer Register enabled

No_Halt

If this bit is set, incrementing of the Performance Timer Register only takes place if the EC is running (not halted).

0: Incrementing of the Performance Timer Register is independent of the EC

1: Incrementing of the Performance Timer Register only occurs when the EC is running

22.0 GPIO INTERFACE

22.1 General Description

The SCH5636 GPIO Interface provides general purpose input monitoring and output control, as well as managing many aspects of pin functionality; including, multi-function Pin Multiplexing Control, Output Buffer Type control, PU/PD resistors, asynchronous wakeup and synchronous Interrupt Detection, GPIO Direction, and Polarity control.

Features of the GPIO Interface include:

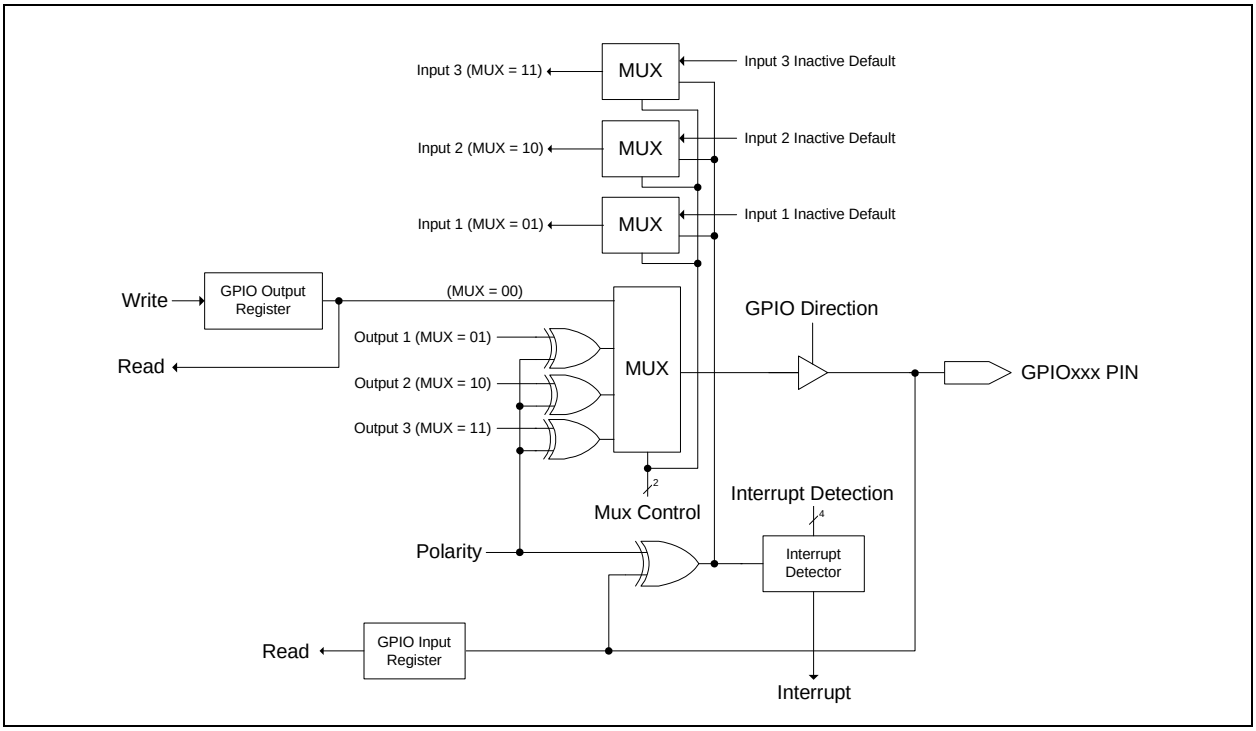
- Inputs:
  - Asynchronous rising and falling edge wakeup detection
  - Interrupt High or Low Level
- On Output:
  - Push Pull or Open Drain output
- Pull up or pull down resistor control
- Interrupt and wake capability available for all GPIOs
- 8 Registers
- Group- or individual control of GPIO data.
- Multiplexing of all multi-function pins are controlled by the GPIO interface

22.2 Block Diagram

The GPIO Interface Block Diagram shown in Figure 22-1 illustrates the functionality of a single SCH5636 GPIO Interface pin. The source for the Pin Multiplexing Control, Interrupt Detection, GPIO Direction, and Polarity controls in Figure 22-1 is a Pin Control Register that is associated with each pin (see Section 22.7.1, "Pin Control Register," on page 246).

The SCH5636 supports up to four independent signal functions per pin including the GPIO signal function itself, which is always positioned at Mux Control = ‘00.’

FIGURE 22-1: GPIO INTERFACE BLOCK DIAGRAM



22.3 GPIO Indexing

Each GPIO signal function name consists of a 2-character prefix (“GP”) followed by a 3-digit octal-encoded index number. In the SCH5636 GPIO indexing is done sequentially starting from ‘GP000.’

22.4 Pin Multiplexing Control

Pin multiplexing depends upon the Mux Control bits in the Pin Control Register. There is a Pin Control Register for each GPIO signal function.

Table 22-1 shows all of the functions on each pin. Default function is assigned on nWDT_RST., except for pins 120 and 121, as described by Note 22-1. The default function is Function 0 unless otherwise noted.

TABLE 22-1: COMPLETE LIST OF PIN FUNCTIONS

Pin #	Pin Name (Default Function First)	Pin Functions			
		Function0	Function1	Function2	Function3
1	SER_IRQ	SER_IRQ	n/a	n/a	n/a
2	CAP1	CAP1	n/a	n/a	n/a
3	VSS	VSS	n/a	n/a	n/a
4	GP000 / PWM4	GP000	PWM4	n/a	n/a
5	GP001 / TACH4	GP001	TACH4	n/a	n/a
6	VCC	VCC	n/a	n/a	n/a
7	GP002 / PCIRST_OUT3#	GP002	PCIRST_OUT3#	n/a	n/a
8	GP003 / PCIRST_OUT4#	GP003	PCIRST_OUT4#	n/a	n/a
9	CLOCKI	CLOCKI	n/a	n/a	n/a
10	LAD0	LAD0	n/a	n/a	n/a
11	LAD1	LAD1	n/a	n/a	n/a
12	LAD2	LAD2	n/a	n/a	n/a
13	LAD3	LAD3	n/a	n/a	n/a
14	LFRAME#	LFRAME#	n/a	n/a	n/a
15	LDRQ#	LDRQ#	n/a	n/a	n/a
16	LRESET# / GP074	GP074	LRESET# **Default**	n/a	n/a
17	SPI_DO / GP004	GP004	SPI_DO **Default**	n/a	n/a
18	PCICLK	PCICLK	n/a	n/a	n/a
19	VSS	VSS	n/a	n/a	n/a
20	AVSS	AVSS	n/a	n/a	n/a
21	CLK32	CLK32	n/a	n/a	n/a
22	GP005 / PECI_REQUEST#	GP005	PECI_ REQUEST#	n/a	n/a
23	VBAT	VBAT	n/a	n/a	n/a
24	VTR	VTR	n/a	n/a	n/a
25	LED1 / GP006	GP006	LED1 **Default**	n/a	n/a
26	LED2 / GP007	GP007	LED2 **Default**	n/a	n/a
27	SMBDAT2 / GP010	GP010	SMBDAT2 **Default**	n/a	n/a

TABLE 22-1: COMPLETE LIST OF PIN FUNCTIONS (CONTINUED)

Pin #	Pin Name (Default Function First)	Pin Functions			
		Function0	Function1	Function2	Function3
28	SMBCLK2 / GP011	GP011	SMBCLK2 **Default**	n/a	n/a
29	SPI_DI / GP012	GP012	SPI_DI **Default**	n/a	n/a
30	SPI_CK / GP013	GP013	SPI_CK **Default**	n/a	n/a
31	PECI VREF	PECI VREF	n/a	n/a	n/a
32	PECI / LVSMBCLK1 / GP072	GP072	PECI **Default**	LVSMBCLK1	n/a
33	PECI READY / LVSMBDAT1 / GP073	GP073	PECI READY **Default**	LVSMBDAT1	n/a
34	VTR	VTR	n/a	n/a	n/a
35	GP014 / INTRUSION# (see Note 22-2)	GP014 / INTRUSION#	n/a	n/a	n/a
36	GP015	GP015	n/a	n/a	n/a
37	VSS	VSS	n/a	n/a	n/a
38	PROCHOT_IN# / PROCHOT_OUT# / GP016	GP016	PROCHOT_IN# **Default**	PROCHOT_OUT#	n/a
39	TACH1 / GP017	GP017	TACH1 **Default**	n/a	n/a
40	TACH2 / GP020	GP020	TACH2 **Default**	n/a	n/a
41	TACH3 / GP021	GP021	TACH3 **Default**	n/a	n/a
42	HVSS	HVSS	n/a	n/a	n/a
43	Remote1+	Remote1+	n/a	n/a	n/a
44	Remote1-	Remote1-	n/a	n/a	n/a
45	Remote2+	Remote2+	n/a	n/a	n/a
46	Remote2-	Remote2-	n/a	n/a	n/a
47	HVTR	HVTR	n/a	n/a	n/a
48	V_IN	V_IN	n/a	n/a	n/a
49	GP022 / PWM1	GP022	PWM1	n/a	n/a
50	GP023 / PWM2	GP023	PWM2	n/a	n/a
51	GP024 / PWM3	GP024	PWM3	n/a	n/a
52	WDT_OUT# / GP025	WDT_OUT# / GP025	n/a	n/a	n/a
53	PCIRST_OUT1 / GP026	GP026	PCIRST_OUT1 **Default**	n/a	n/a
54	PCIRST_OUT2 / GP027	GP027	PCIRST_OUT2 **Default**	n/a	n/a
55	PS_ON#/ / GP030	GP030	PS_ON# **Default/**	n/a	n/a
56	GP031 / BACKFEED_CUT#	GPO31	n/a	BACKFEED_CUT# **Default**	n/a
57	VTR	VTR	n/a	n/a	n/a

TABLE 22-1: COMPLETE LIST OF PIN FUNCTIONS (CONTINUED)

Pin #	Pin Name (Default Function First)	Pin Functions			
		Function0	Function1	Function2	Function3
58	SPI_CS# / GPO32	GP032	SPI_CS# **Default**	n/a	n/a
59	PWR_GOOD_3V / GP033	GP033	PWR_GOOD_3V **Default**	n/a	n/a
60	RSMRST# / GP034	GP034	RSMRST# **Default**	n/a	n/a
61	VSS	VSS	n/a	n/a	n/a
62	DSKCHG#	DSKCHG#	n/a	n/a	n/a
63	HDSEL#	HDSEL#	n/a	n/a	n/a
64	RDATA#	RDATA#	n/a	n/a	n/a
65	WRTPRT#	WRTPRT#	n/a	n/a	n/a
66	TRK0#	TRK0#	n/a	n/a	n/a
67	WGATE#	WGATE#	n/a	n/a	n/a
68	WDATA#	WDATA#	n/a	n/a	n/a
69	LATCHED_BF_CUT / GP035	GP035	n/a	LATCHED_ BF_CUT **Default**	n/a
70	STEP#	STEP#	n/a	n/a	n/a
71	DIR#	DIR#	n/a	n/a	n/a
72	GP036 / SMBCLK1	GP036	SMBCLK1	n/a	n/a
73	DS0#	DS0#	n/a	n/a	n/a
74	GP040 / SMBDAT1	GP040	SMBDAT1	n/a	n/a
75	MTR0#	MTR0#	n/a	n/a	n/a
76	INDEX#	INDEX#	n/a	n/a	n/a
77	GP041 / IO_PME#	GP041	IO_PME#	n/a	n/a
78	GP042 / DRVDEN0	GP042	DRVDEN0	n/a	n/a
79	VTR	VTR	n/a	n/a	n/a
80	SLCT	SLCT	n/a	n/a	n/a
81	PE	PE	n/a	n/a	n/a
82	BUSY	BUSY	n/a	n/a	n/a
83	ACK#	ACK#	n/a	n/a	n/a
84	PD7	PD7	n/a	n/a	n/a
85	PD6	PD6	n/a	n/a	n/a
86	PD5	PD5	n/a	n/a	n/a
87	PD4	PD4	n/a	n/a	n/a
88	PD3 / TMS	PD3	n/a	n/a	n/a
89	PD2 / TDO	PD2	n/a	n/a	n/a
90	PD1 / TDI	PD1	n/a	n/a	n/a
91	PD0 / TCK	PD0	n/a	n/a	n/a
92	VSS	VSS	n/a	n/a	n/a
93	SLCTIN#	SLCTIN#	n/a	n/a	n/a
94	INIT#	INIT#	n/a	n/a	n/a
95	ERROR#	ERROR#	n/a	n/a	n/a
96	ALF#	ALF#	n/a	n/a	n/a
97	STROBE#	STROBE#	n/a	n/a	n/a

TABLE 22-1: COMPLETE LIST OF PIN FUNCTIONS (CONTINUED)

Pin #	Pin Name (Default Function First)	Pin Functions			
		Function0	Function1	Function2	Function3
98	DCD1# / GP043 / MCDAT	GP043	DCD1# **Default**	MCDAT	n/a
99	DSR1# / GP044 / MCCLK	GP044	DSR1# **Default**	MCCLK	n/a
100	RXD1 / GP045	GP045	RXD1 **Default**	n/a	n/a
101	RTS1# / GP046 [SYSOPT]	GP046	RTS1# **Default**	n/a	n/a
102	GP047 / TXD1	GP047	TXD1 **Default**	n/a	n/a
103	CTS1# / GP050	GP050	CTS1# **Default**	n/a	n/a
104	DTR1# [TEST_EN] / GP051	GP051	DTR1# **Default**	n/a	n/a
105	RI1# / GP052	GP052	RI1# **Default**	n/a	n/a
106	VTR	VTR	n/a	n/a	n/a
107	GP053 / DCD2#	GP053	DCD2#	n/a	n/a
108	GP054 / DSR2#	GP054	DSR2#	n/a	n/a
109	GP055 / RXD2	GP055	RXD2	n/a	n/a
110	GP056 / RTS2#	GP056	RTS2#	n/a	n/a
111	GP057 / TXD2	GP057	TXD2	n/a	n/a
112	GP060 / CTS2#	GP 060	CTS2#	n/a	n/a
113	GP 061 / DTR2#	GP061	DTR2#	n/a	n/a
114	GP062 / RI2#	GP062	RI2#	n/a	n/a
115	KCLK	KCLK	n/a	n/a	n/a
116	KDAT	KDAT	n/a	n/a	n/a
117	MCLK	MCLK	n/a	n/a	n/a
118	MDAT	MDAT	n/a	n/a	n/a
119	VSS	VSS	n/a	n/a	n/a
120	GP063 / KBDRST# (Note 22-1)	GP063	KBDRST#	n/a	n/a
121	GP064 / A20M (Note 22-1)	GP064	A20M	n/a	n/a
122	VTR	VTR	n/a	n/a	n/a
123	SLP_S3# / GP065	GP065	SLP_S3# **Default**	n/a	n/a
124	SLP_S4_S5# / GP066	GP066	SLP_S4_S5# **Default**	n/a	n/a
125	PWRGD_PS / GP067	GP067	PWRGD_PS **Default**	n/a	n/a
126	TRST#	TRST#	n/a	n/a	n/a
127	GP070 / SPEAKER	GP070	SPEAKER	n/a	n/a
128	GP071 / IO_SMI#	GP071	IO_SMI#	n/a	n/a

Note 22-1 The pin mux function for this pin is only reset on nSYS_RST. The nWDT_RST signal does not change the current pin configuration.

Note 22-2 The INTRUSION# function is always enabled, in parallel with the GPIO function. The pin function multiplex control is not used.

22.5 Power, Clocks and Reset

22.5.1 RESET

This block is reset on a nSYS_RST. On reset, all Registers are reset to their default values.

22.6 Registers

Table 22-3 is a register summary for the GPIO Interface. Each EC address is indicated as an SPB Offset from its AHB base address.

TABLE 22-2: GPIO INTERFACE BASE ADDRESS TABLE

GPIOs Blocks	LDN	AHB Base Address
GPIOs	31h	F0_C400h

TABLE 22-3: GPIO INTERFACE REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Pin Control Register	-	000h - 200h	32	R/W	

22.7 Pin Control Register table

Default values for the Pin Control registers are shown in Table 22-3, "GPIO Interface Register Summary":

TABLE 22-4: GPIO PIN CONTROL REGISTER DEFAULT TABLE

GPIO	AHB Off	DFLT	GPIO	AHB Off	DFLT	GPIO	AHB Off	DFLT
GP000	000h	0000_0000h	GP025	054h	0000_0000h	GP052	0A8h	0000_1000h
GP001	004h	0000_0000h	GP026	058h	0000_1200h	GP053	0ACh	0000_0000h
GP002	008h	0000_0000h	GP027	05Ch	0000_1200h	GP054	0B0h	0000_0000h
GP003	00Ch	0000_0000h	GP030	060h	0000_1300h	GP055	0B4h	0000_0000h
GP004	010h	0000_1200h	GP031	064h	0000_0000h	GP056	0B8h	0000_0000h
GP005	014h	0000_0000h	GP032	068h	0000_1200h	GP057	0BCh	0000_0000h
GP006	018h	0000_1B00h	GP033	06Ch	0000_1200h	GP060	0C0h	0000_0000h
GP007	01Ch	0000_1B00h	GP034	070h	0000_1200h	GP061	0C4h	0000_0000h
GP010	020h	0000_1300h	GP035	074h	0000_2200h	GP062	0C8h	0000_0000h
GP011	024h	0000_1300h	GP036	078h	0000_0000h	GP063	0CCh	0000_0000h
GP012	028h	0000_1000h	RES	07Ch	0000_0000h	GP064	0D0h	0000_0000h
GP013	02Ch	0000_1200h	GP040	080h	0000_0000h	GP065	0D4h	0000_1000h
GP014	030h	0000_0000h	GP041	084h	0000_0000h	GP066	0D8h	0000_1000h
GP015	034h	0000_0000h	GP042	088h	0000_0000h	GP067	0DCh	0000_1000h
GP016	038h	0000_1000h	GP043	08Ch	0000_1000h	GP070	0E0h	0000_0000h
GP017	03Ch	0000_1000h	GP044	090h	0000_1000h	GP071	0E4h	0000_0000h
GP020	040h	0000_1000h	GP045	094h	0000_1000h	GP072	0E8h	0000_1000h
GP021	044h	0000_1000h	GP046	098h	0000_1200h	GP073	0ECh	0000_1000h
GP022	048h	0000_0000h	GP047	09Ch	0000_0000h	GP074	0F0h	0000_1000h

TABLE 22-4: GPIO PIN CONTROL REGISTER DEFAULT TABLE (CONTINUED)

GPIO	AHB Off	DFLT	GPIO	AHB Off	DFLT	GPIO	AHB Off	DFLT
GP023	04Ch	0000_0000h	GP050	0A0h	0000_1000h	RES	0F4h	0000_0000h
GP024	050h	0000_0000h	GP051	0A4h	0000_1200h	RES	0F8h	0000_0000h

Note: The only field that can be modified in the pin control register for GP067, the register located at offset 0DCh, is the Interrupt Detection field. Other fields, such as the Mux Control and GPIO Direction, are fixed at their default values.

Note: The Pin Control registers for GP063 and GP064 are only reset on nSYS_RST. All other GPIO Pin Control registers are reset on nWDT_RST.

The Pin Control Register format is illustrated in Table 22-5 below and described in the subsections that follow. Pin Control Register address offsets and defaults are defined in Section 22.3, "GPIO Indexing," on page 241.

22.7.1 PIN CONTROL REGISTER

TABLE 22-5: PIN CONTROL REGISTER

HOST ADDRESS	N/A					N/A	**HOST SIZE**	
EC OFFSET	See Table 22-4 on page 245			32-bit			**EC SIZE**	
POWER	VTR			See Table 22-3 on page 245			nWDT_RST **DEFAULT**	
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							GPIO input from pad
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
TYPE	R	R	R	R	R	R	R/W	R/W
BIT NAME	Reserved						Alternative GPIO Write Enable	Alternative GPIO data
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
TYPE	R	R	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved		Mux Control		Polarity	Reserved	GPIO Direction	Output Buffer Type
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
TYPE	R/W	R/W	R/W	R/W	R	R	R/W	
BIT NAME	Edge Enable	Interrupt Detection			Reserved		PU/PD	

PU/PD

TABLE 22-6: PU/PD BITS DEFINITION

Bit 1	Bit 0	Selected Function
0	0	None
0	1	Pull Up Enabled
1	0	Pull Down Enabled
1	1	None

Interrupt Detection

TABLE 22-7: INTERRUPT DETECTION BITS DEFINITION

D6	D5	D4	Selected Function
0	0	0	Low Level Sensitive
0	0	1	High Level Sensitive
0	1	0	Reserved
0	1	1	Reserved
1	0	0	Reserved
1	0	1	Rising Edge Triggered
1	1	0	Falling Edge Triggered
1	1	1	Either edge triggered

Note: Only edge triggered interrupts can wake up the main ring oscillator. The GPIO must be configured for edge-triggered interrupts (Interrupt Detection set to 101b - 111b), edge-triggered interrupts must be enabled (Edge Enable set to 1b) and the GPIO interrupt must be enabled in the interrupt aggregator in order to wake up the ring when the ring is shut down.

Edge Enable

When this bit is ‘1’, the GPIO has edge detection enabled. When this bit is ‘0’, edge detection is disabled.

When this bit is ‘0’, the ring oscillator wakeup function is disabled. In order to put the pin in its lowest power state, the Edge Enable bit should be set to ‘0’, and the Interrupt Detection field set to one of the edge triggered values (101b - 111b). This combination provides that no interrupt will be generated and that no wakeup function will be enabled.

TABLE 22-8: EDGE ENABLE BIT DEFINITION

D7	Description
0	Edge detection disabled
1	Edge detection enabled

Output Buffer Type

TABLE 22-9: OUTPUT BUFFER TYPE BIT DEFINITION

D8	Selected Function
0	Push-Pull
1	Open Drain

Note 22-3 Unless explicitly stated otherwise, pins with (I/O/OD) or (O/OD) in their buffer type column in the tables in Section TABLE 3-1:, "Signal Descriptions," on page 11 are compliant with the following Programmable OD/PP Multiplexing Design Rule: Each compliant pin has a programmable open drain/push-pull buffer controlled by the Output Buffer Type bit in the associated Pin Control

Register. The state of this bit controls the mode of the interface buffer for all selected functions, including the GPIO function.

GPIO Direction

The GPIO Direction bit controls the buffer direction only when the Mux Control field is ‘00’ selecting the pin signal function to be GPIO. When the Mux Control field is greater than ‘00’ (i.e., a non-GPIO signal function is selected) the GPIO Direction bit has no affect and the selected signal function logic directly controls the pin direction.

TABLE 22-10: GPIO DIRECTION BIT DEFINITION

D9	Selected Function
0	Input
1	Output

Polarity

When the Polarity bit is set to ‘1’ and the Mux Control bits are greater than ‘00,’ the selected signal function outputs are inverted and Interrupt Detection sense defined in Table 22-7, "Interrupt Detection Bits Definition" is inverted. When the Mux Control field selects the GPIO signal function (Mux = ‘00’), the Polarity bit does not effect the output. Regardless of the state of the Mux Control field and the Polarity bit, the state of the pin is always reported without inversion in the GPIO input register. See FIGURE 22-1: GPIO Interface Block Diagram on page 240.

TABLE 22-11: POLARITY BIT DEFINITION

D11	Description
0	Non-inverted
1	Inverted

Mux Control

TABLE 22-12: MUX CONTROL BIT DEFINITION

D13	D12	Description
0	0	GPIO Function Selected
0	1	Signal Function 1 Selected
1	0	Signal Function 2 Selected
1	1	Signal Function 3 Selected

The Mux Control field determines the active signal function for a pin as defined in Table 22-12.

Alternative GPIO Write Enable

If this bit is 1, then a write to bit D16, Alternative GPIO Data, updates the GPIO output latch. If this bit is 0, then writes to bit D16 have no effect on the GPIO output.

TABLE 22-13: ALTERNATIVE GPIO BITS DEFINITION ON WRITES

D17	D16	Description
0	x	GPIO alternative write disabled
1	0	GPIO[x] out = ‘0’
1	1	GPIO[x] out = ‘1’

Alternative GPIO Data

On writes, see Table 22-13. On reads, Bit [16] returns the programmed value.

GPIO Input from Pad

On reads, Bit [24] reflects the state of GPIO input from the pad regardless of setting of Bit [17].

23.0 WATCHDOG TIMER

23.1 General Description

The function of the Watchdog Timer is to provide a mechanism to detect if the embedded controller has failed.

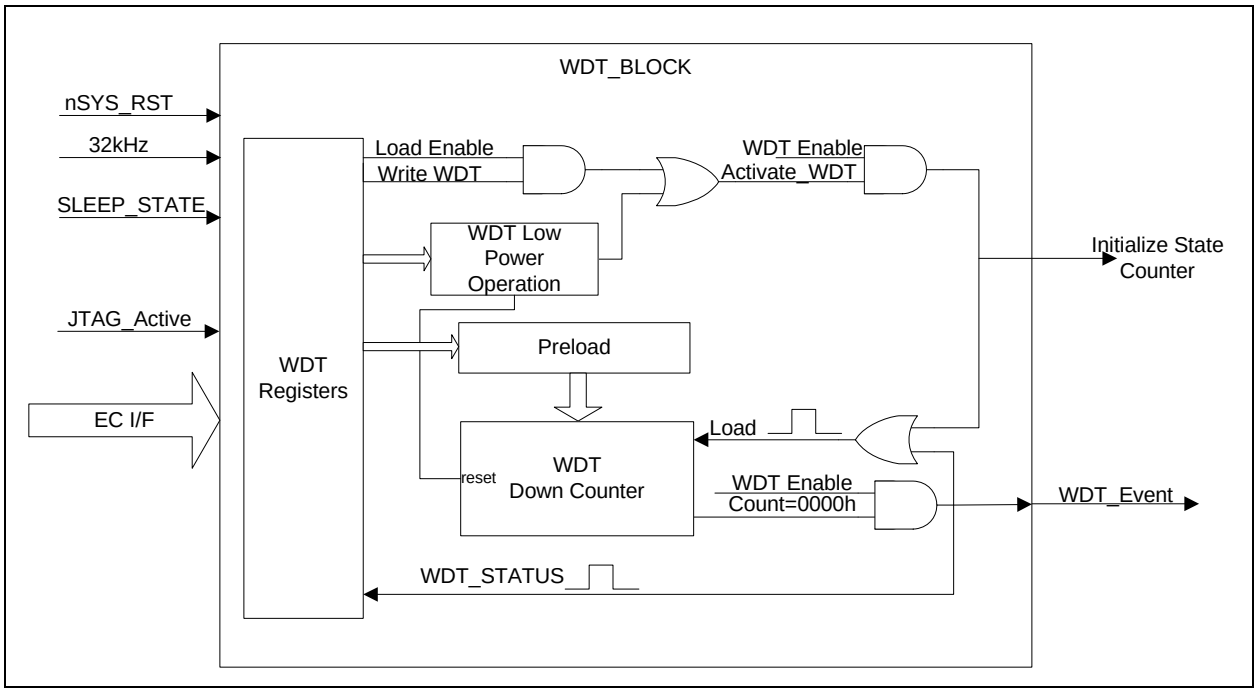
When enabled, the Watchdog Timer (WDT) circuit will generate a WDT Event if the user program fails to reload the WDT within a specified length of time known as the WDT Interval.

This timer can be held inactive via the WDT Stall feature if the JTAG interface is enabled and active. This featured, if enabled, can be used to avoid unintended system resets.

Some operations can be carried out without any delay, e.g., registers can be read at any time and disabling the WDT takes effect immediately. On the other hand, 'kicking' the WDT may have a latency of up to 1 32-kHz cycle (~ 30 us). Similarly, when the load register is altered, the WDT cannot be enabled for up to 1 32-kHz cycle.

23.2 Block Diagram

FIGURE 23-1: WATCHDOG TIMER BLOCK DIAGRAM



23.3 Watchdog Timer Signal List

TABLE 23-1: WATCHDOG TIMER SIGNAL LIST

Signal Name	Direction	Description
nSYS_RST	INPUT	Power on Reset to the block
32kHz	INPUT	32.768 KHz Clock, Clock source for WDT logic
JTAG_Active	INPUT	Signal indicating the JTAG interface is active. See Section 33.0, "JTAG and XNOR," on page 312
SPB Interface	I/O Bus	Bus used by microprocessor to access the registers in this block.
WDT Event	OUTPUT	Pulse generated when WDT expires

Note: In the SCH5636, the assertion of the WDT Event output from this block causes a Watchdog Timer reset. See Section 5.7.6, "nWDT_RST," on page 37.

23.4 Power, Clocks and Reset

23.4.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

23.4.2 CLOCKS

This block has two clock inputs, the EC Bus Clock and the 32.768 KHz Clock. The EC Bus Clock is used in the interface to the embedded controller accessible registers. The 32.768 KHz Clock is the clock source for the Watchdog Timer functional logic, including the counter.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

23.4.3 RESET

This block is reset on a nSYS_RST.

See Section 5.9, "Registers," on page 43 for details on reset.

23.5 WDT Event output routing

The WDT Event (output) causes a nWDT_RST reset.

The WDT Event state is also retained in the WDT bit of the Power-Fail and Reset Status Register on page 55. The Power-Fail and Reset Status Register can generate a interrupt via the PFR interrupt located in the GIRQ22 Source register.

The WDT Event output is not directly connected to an EC interrupt.

23.6 WDT Operation

23.6.1 WDT ACTIVATION MECHANISM

The WDT is activated by the following sequence of operations during normal operation:

1. Load the WDT Load Register with the count value.
2. Wait at least one period of the 32.768KHz clock (30.5µs).
3. Set the WDT Enable bit in the WDT Control Register.

The WDT Activation Mechanism starts the WDT decrementing counter.

23.6.2 WDT DEACTIVATION MECHANISM

The WDT is deactivated by the clearing the WDT Enable bit in the WDT Control Register. The WDT Deactivation Mechanism places the WDT in a low power state in which clock are gated and the counter stops decrementing.

23.6.3 WDT RELOAD MECHANISM

The WDT must be reloaded within periods that are shorter than the programmed watchdog interval; otherwise, the WDT will underflow and a WDT Event will be generated and the WDT_STAT bit will be set in the WDT Control Register. It is the responsibility of the user program to continually execute code which reloads the watchdog timer, causing the counter to be reloaded.

There are two methods of reloading the WDT: a write to the WDT Kick Register or the WDT Activation Mechanism.

23.6.4 WDT INTERVAL

The WDT Interval is the time it takes for the WDT to decrements from the WDT Load Register value to 0000h. The WDT Count Register value takes 1.007ms to decrement by 1 count.

23.6.5 WDT STALL OPERATION

When a WDT STALL event is asserted, the WDT stops decrementing, and the WDT enters a low power state. When a WDT STALL event is de-asserted, the counter continues decrementing from the.value it had hen the STALL was asserted.

The WDT STALL feature has been implemented for convenience. The WDT defaults with the WDT STALL feature disabled.

There is one Stall input to the WDT, the JTAG_Active, corresponding to the JTAG interface being active. The Stall input has an enable bit: the JTAG STALL_EN bit in the WDT Control Register.

TABLE 23-2: WDT STALL EVENT BEHAVIOR

WDT STALL Input (Activity Indicator)	WDT Control Register on page 253		WDT Behavior	WDT Event Output
	STALL_EN Bit	WDT Enable Bit		
X	X	0	Counter is reset and not active. Clock source to counter is gated to save power.	0
X	0	1	Count is active. If counter > 0000h	1
0	1	1		
X	X	1	Count is decremented to 0000h	1
1	1	1	Counter is not active. Clock source to counter is gated to save power.	0

When the WDT recovers from a STALL event, the counter is reloaded with the programmed preload value.

When the counter reaches 0000h it wraps to the preload value and starts counting down again. This creates a pulse on the WDT Event output.

23.7 Instance Description

There is one instance of the Watchdog Timer block implemented in the SCH5636.

The Watchdog Timer is instantiated on the EC AHB and has its own Logical Device Number, and Base Address as indicated in Table 23-3.

TABLE 23-3: WATCHDOG TIMER BASE ADDRESS TABLE

Watchdog Timer Instance	LDN	AHB Base Address
Watchdog Timer	1h	F0_0400h

The Table 23-4 is a register summary for the Watchdog Timer. Each EC address is indicated as an SPB Offset from its AHB address.

TABLE 23-4: WATCHDOG TIMER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
WDT Load Register	-	00h	32	R/W	
WDT Control Register	-	04h	32	R/W	
WDT Kick Register	-	08h	32	R/W	
WDT Count Register	-	0Ch	32	R	

23.8 Detailed Register Descriptions

23.8.1 WDT LOAD REGISTER

TABLE 23-5: WDT LOAD REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	00h					32-bit	**EC SIZE**	
POWER	VTR					FFFFh	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	▪	▪	**D2**	**D1**	**D0**
HOST TYPE	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	WDT Load							

WDT Load

Writing this field reloads the Watch Dog Timer counter.

23.8.2 WDT CONTROL REGISTER

TABLE 23-6: WDT CONTROL REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	04h					32-bit	**EC SIZE**	
POWER	VTR					00h	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EC TYPE	R	R	R	R/W	R	R	R/WC	R/W
BIT NAME	Reserved			JTAG STALL_EN	Reserved		WDT Status	WDT Enable

WDT Enable

Note:	The default of the WDT is inactive.

In WDT Operation, the WDT is activated by the sequence of operations defined in Section 23.6.1, "WDT Activation Mechanism" and deactivated by the sequence of operations defined in Section 23.6.2, "WDT Deactivation Mechanism". In WDT STALL Operation, hardware may be enabled to automatically activate and deactivate the WDT.

WDT Status

WDT_RST is set by hardware if the last reset of SCH5636 was caused by an underflow of the WDT. See Section 23.6.3, "WDT Reload Mechanism," on page 251 for more information.

This bit must be cleared by the EC firmware writing a ‘1’ to this bit. Writing a ‘0’ to this bit has no effect.

JTAG STALL_EN

This bit is used to enable the JTAG_Active (JTAG_RST# pin not asserted) See WDT STALL Operation on page 251.

0= JTAG_Active WDT STALL Operation not enabled

1= JTAG_Active WDT STALL Operation enabled

23.8.3 WDT KICK REGISTER

TABLE 23-7: WDT KICK REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	08h					32-bit	**EC SIZE**	
POWER	VTR					n/a	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EC TYPE	W	W	W	W	W	W	W	W
BIT NAME	Kick							

Kick

The WDT Kick Register is a strobe. Reads of the WDT Kick Register return 0.

Writes to the WDT Kick Register cause the WDT to reload the WDT Load Register value and start decrementing when the WDT Enable bit in the WDT Control Register is set to ‘1’. When the WDT Enable bit in the WDT Control Register is cleared to ‘0’, writes to the WDT Kick Register have no effect.

23.8.4 WDT COUNT REGISTER

TABLE 23-8: WDT COUNT REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	00h					32-bit	**EC SIZE**	
POWER	VTR					FFFFh	nSYS_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	▪	▪	**D2**	**D1**	**D0**
HOST TYPE	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	WDT COUNT							

WDT COUNT

This read-only register provide the current WDT count.

24.0 16-BIT TIMER

24.1 General Description

The SCH5636 16-Bit Timer block implements a 16-bit auto-reloading timer.

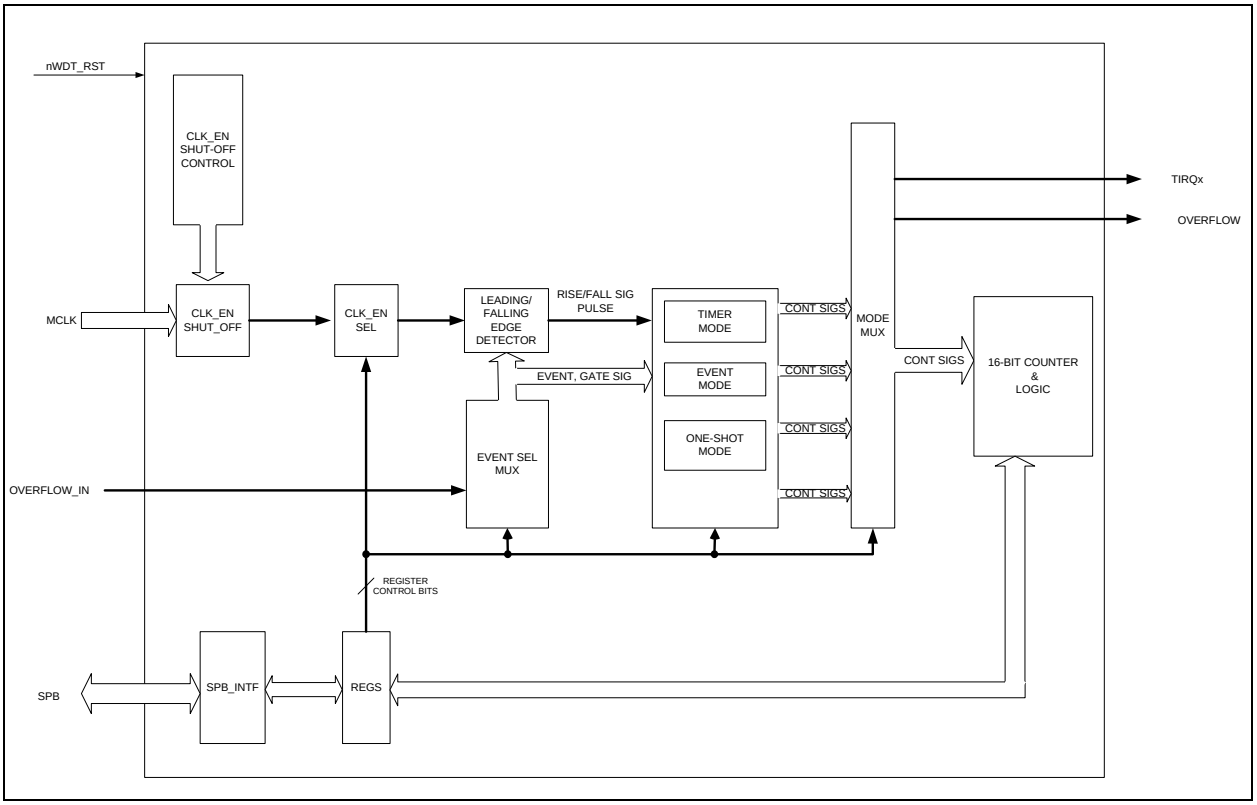
24.2 Terms

The following defines terms used in this chapter.

Term	Definition
Overflow	When the timer counter transitions from FFFFh to 0000h
Underflow	When the timer counter transitions from 0000h to FFFFh.
timer tick rate	This is the rate at which the timer is incremented or decremented.

24.3 Block Diagram

FIGURE 24-1: BLOCK DIAGRAM FOR TIMER X



24.4 Signal List for Block Diagram

TABLE 24-1: BLOCK DIAGRAM SIGNAL LIST DESCRIPTION

Signal Name	Direction	Description
nWDT_RST	INPUT	Watchdog timer reset
MCLK	INPUT	Clock source to block.
DIV1,2,4,8,16,32,64,128_EN	INPUT	Clock Enables for supporting Filter and Timer frequencies.

TABLE 24-1: BLOCK DIAGRAM SIGNAL LIST DESCRIPTION (CONTINUED)

Signal Name	Direction	Description
OVERFLOW_IN	INPUT	Input signal for timer
TIRQx	OUTPUT	Timer x Interrupt Request
OVERFLOW	OUTPUT	Output signal from timer
SPB	I/O Bus	Bus used by microprocessor to access the registers in this block.

24.5 Timers

TABLE 24-2: TIMER CASCADING DESCRIPTION

Timer Name	Timer Type	Overflow/Underflow Input Connection
Timer 1	General Purpose	none
Timer 2	General Purpose	from Timer 1
Timer 3	General Purpose	from Timer 2

Note: All timers cascaded together must be on the same clock source.

24.6 Power, Clocks and Reset

24.6.1 POWER DOMAIN

This block is powered by the VTR power supply.

24.6.2 CLOCKS

There is a clock enable input for each of the supported frequencies listed in Table 24-11, “Timer Clock Frequencies,” on page 263. Any of these enables may be selected for the Timer Clock Frequency.

24.6.3 RESET

This block is reset on a nWDT_RST. On nWDT_RST all timers are reset to their default values. The timers are also reset by the RESET bit in each Timer x Control Register.

See Section 5.9, "Registers," on page 43 for details on reset.

24.7 Interrupts

The timers in the SCH5636 can be used to generate interrupts when the timer overflows or underflows. The timer interrupts are routed to the TIME3, TIMER2 and TIMER1 bits in the GIRQ23 Source Register.

Note: No interrupts are generated while the ENABLE bit is cleared.

24.8 Low Power Modes

This block is designed to conserve power when it is either sleeping or a clock source is not required.

During normal operation, if the timer is disabled via the PD bit the TIMERx_CLK_REQ signal is de-asserted. This indicates to the clock generator logic that this timer does not require the 64.52MHz clock source.

During Sleep modes the clock input is gated, the TIMERx_CLK_REQ signal is asserted, and the interrupt output goes to the inactive state. When the block returns from sleep, if enabled, it will be restarted from the preload value.

Note: The timer is terminated one TCLK after the SLEEP ENABLE is asserted.

The following table illustrates the low power mode options.

TABLE 24-3: BLOCK CLOCK GATING IN LOW POWER MODES

Power Down (PD) Bit	SLEEP ENABLE	Block Idle Status	TIMERX_CLK_REQ	State	Description
1	X	NOT IDLE	1	PREPARING to SLEEP	The core clock is still required for up to one Timer Clock period.
		IDLE	0	SLEEPING	The block is idle and the core clock can be stopped.
0	0	X	1	NORMAL OPERATION	The block in neither disabled by firmware nor commanded to SLEEP
	1	NOT IDLE	1	PREPARING to SLEEP	The core clock is still required for up to one Timer Clock period.
		IDLE	0	SLEEPING	The block is idle and the core clock can be stopped.

24.9 Operating Modes

24.9.1 STARTING AND STOPPING

The SCH5636 timers can be started and stopped by setting and clearing the Timer Enable bit in the Timer Control Register in all modes, except one-shot.

24.9.2 TIMER MODE

The Timer mode of the SCH5636 is used to generate periodic interrupts to the EC. When operating in this mode the timer always counts down based on one of the internally generated clock sources. The Timer mode is selected by setting the Timer Mode Select bits in the Timer Control Register. See Section 24.11.1, "Timer x Control Register," on page 261

The period between timer interrupts and the width of the output pulse is determined by the speed of the clock source and the value programmed into the Timer Reload Register. The timer clock source and clock rate are selected using the Clock Source Select bits (TCLK) in the Timer x Clock and Event Control Register. See Section 24.11.2, "Timer x Clock and Event Control Register," on page 263.

TABLE 24-4: TIMER MODE OPERATIONAL SUMMARY

Item	Description
Timer Clock Frequencies	This mode supports all the programmable frequencies listed in Table 24-11, “Timer Clock Frequencies,” on page 263
Count Operation	Down Counter
Reload Operation	When the timer underflows: RLOAD = 1, timer reloads from Timer Reload Reg RLOAD = 0, timer rolls over to FFFFh.
Count Start Condition	ENABLE = 1
Count Stop Condition	ENABLE = 0
Interrupt Request Generation Timing	When timer underflows from 0000h to reload value (as determined by RLOAD) an interrupt is generated.
Read From Timer	Current count value can be read by reading the Timer Count Register
Write to Preload Register	When the timer is stopped, values written to the Timer Reload Register are copied into the timer counter. When the timer is running, values written to the Timer Reload Register are written to the timer counter when the timer underflows. The assertion of Reset also copies the Timer Reload Register into the timer counter.
Selectable Functions	▪ Reload timer on underflow with programmed Preload value (Basic Timer) ▪ Reload timer with FFFFh in Free Running Mode (Free-running Timer)

24.9.2.1 Timer Mode Underflow

The SCH5636 timers operating in Timer mode can underflow in two different ways. One method, the Reload mode shown in Figure 24-2, is to reload the value programmed into the Reload register and continue counting from this value. The second method, the Free Running mode shown in Figure 24-3, is to set the timer to FFFFh and continue counting from this value. The underflow behavior is controlled by the RLOAD bit in the Timer Control Register.

FIGURE 24-2: RELOAD MODE BEHAVIOR

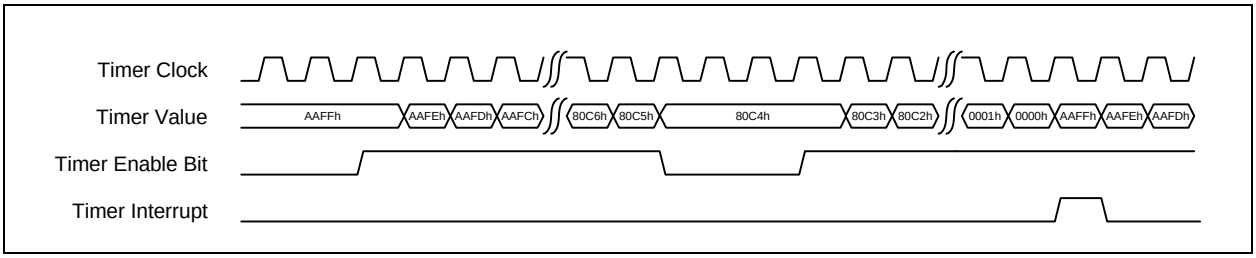
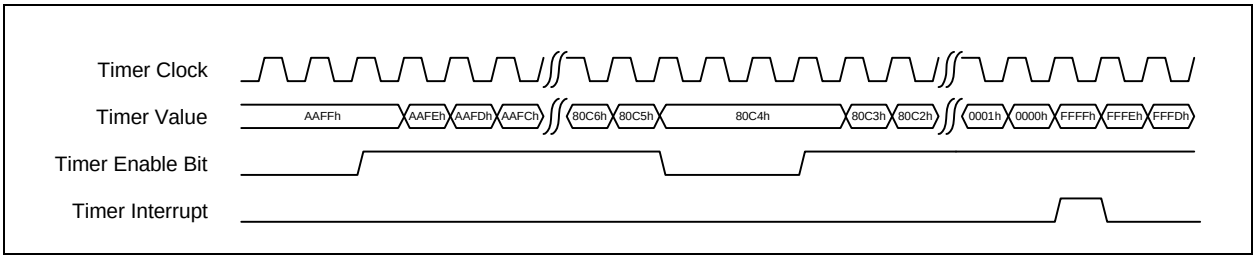


FIGURE 24-3: FREE RUNNING MODE BEHAVIOR



24.9.3 EVENT MODE

Event mode is used to count overflow events from the previous timer. The direction the timer counts in Event mode is controlled by the UPDN bit in the Timer Control Register.

The timer can be programmed using the Clock and Event Control register to respond to the rising edge of overflow input, falling edge of the overflow input, and the rising and falling edges of the overflow input.

TABLE 24-5: EVENT MODE OPERATIONAL SUMMARY

Item	Description
Timer Clock Frequencies	This mode supports all the programmable frequencies listed in Table 24-11, “Timer Clock Frequencies,” on page 263
Count Operation	Up/Down Counter
Reload Operation	• When the timer underflows: RLOAD = 1, timer reloads from Timer Reload Reg RLOAD = 0, timer rolls over to FFFFh. • When the timer overflows: RLOAD = 1, timer reloads from Timer Reload Reg RLOAD = 0, timer rolls over to 0000h.
Count Start Condition	Timer Enable is set (ENABLE = 1)
Count Stop Condition	Timer Enable is cleared (ENABLE = 0)
Interrupt Request Generation Timing	When timer overflows or underflows
Read From Timer	Current count value can be read by reading the Timer Count Register

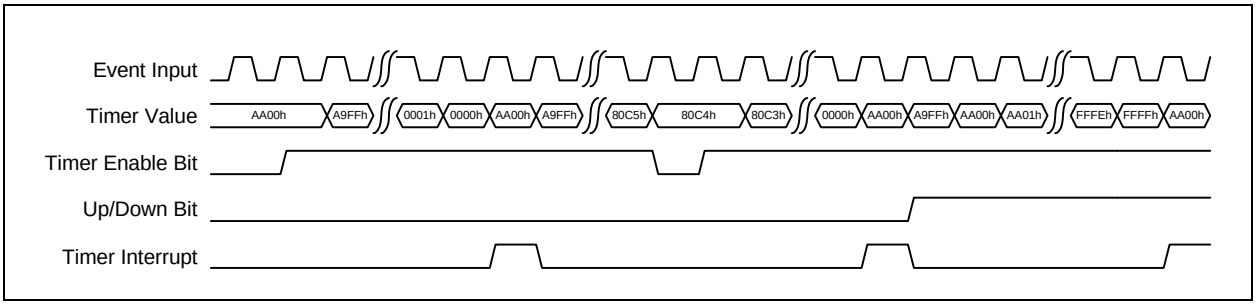
TABLE 24-5: EVENT MODE OPERATIONAL SUMMARY (CONTINUED)

Item	Description
Write to Preload Register	When the timer is stopped, values written to the Timer Reload Register are copied into the timer counter. When the timer is running, values written to the Timer Reload Register are written to the timer counter when the timer underflows. The assertion of Reset also copies the Timer Reload Register into the timer counter.
Selectable Functions	• The direction of the counter is selectable via the UPDN bit. • Reload timer on underflow/overflow with programmed Preload value (Basic Timer) • Reload timer with FFFFh in Free Running Mode (Free-running Timer)

24.9.3.1 Event Mode Operation

The timer starts counting events when the ENABLE bit in the Timer Control Register is set and continues to count until the ENABLE bit is cleared. When the ENABLE bit is set, the timer continues counting from the current value in the timer except after a reset event. After a reset event, the timer always starts counting from the value programmed in the Reload Register if counting down or from 0000h if counting up. Figure 24-4 shows an example of timer operation in Event mode. The RLOAD bit controls the behavior of the timer when it underflows or overflows.

FIGURE 24-4: EVENT MODE OPERATION



24.9.4 ONE-SHOT MODE

The One-Shot mode of the timer is used to generate a single interrupt to the EC after a specified amount of time. The timer can be configured to start using the EN bit (Figure 24-5) or on a timer overflow event from the previous timer. See Section 24.11.2, "Timer x Clock and Event Control Register," on page 263 for configuration details. The ENABLE bit must be set for an event to start the timer. The ENABLE bit is cleared one clock after the timer starts. The timer always starts from the value in the Reload Register and counts down in One-Shot mode.

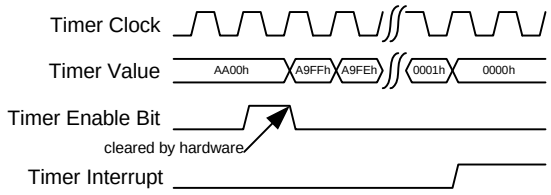
TABLE 24-6: ONE SHOT MODE OPERATIONAL SUMMARY

Item	Description
Timer Clock Frequencies	This mode supports all the programmable frequencies listed in Table 24-11, “Timer Clock Frequencies,” on page 263
Count Operation	Down Counter
Reload Operation	When the timer underflows the timer will stop. When the timer is enabled timer starts counting from value programmed in Timer Reload Register. (RLOAD has no effect in this mode)
Count Start Condition	Setting the ENABLE bit to 1 starts One-Shot mode. The timer clock automatically clears the enable bit one timer tick later. **Note:** One-Shot mode may be enabled in Event Mode. In Event mode an overflow from the previous timer is used for timer tick rate.
Count Stop Condition	• Timer is reset (RESET = 1) • Timer underflows
Interrupt Request Generation Timing	When an underflow occurs.
Read From Timer	Current count value can be read by reading the Timer Count Register

TABLE 24-6: ONE SHOT MODE OPERATIONAL SUMMARY (CONTINUED)

Item	Description
Write to Preload Register	When the timer is stopped, values written to the Timer Reload Register are copied into the timer counter. When the timer is running, values written to the Timer Reload Register are written to the timer counter when the timer underflows. The assertion of Reset also copies the Timer Reload Register into the timer counter.
Selectable Functions	Pulse Output Function

FIGURE 24-5: TIMER START BASED ON EN BIT



24.10 16-Bit Counter/Timer Interface Register Summary

There are three instances of the 16-Bit Timer block implemented in the SCH5636 enumerated as [1:3] with an overflow/underflow interface. Each instance of the 16-Bit Timer is instantiated on the EC AHB and has its Base Address as indicated in Table 24-7.

TABLE 24-7: 16-BIT TIMER INTERFACE BASE ADDRESS TABLE

16-Bit Timer Instance	LDN	AHB Base Address
TIMER1	3h	F0_0C00h
TIMER2		F0_0C80h = F0_0C00h + 80h
TIMER3		F0_0D00h = F0_0C00h + 100h

Table 24-8 is a register summary for one instance of the 16-Bit Timer block.

TABLE 24-8: 16-BIT TIMER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Timer x Control Register	-	00h	32	R/W	
Timer x Clock and Event Control Register	-	04h	32	R/W	
Timer x Reload Register	-	08h	32	R/W	
Timer x Count Register	-	0Ch	32	R	

24.11 Detailed Register Descriptions

24.11.1 TIMER X CONTROL REGISTER

TABLE 24-9: TIMER X CONTROL REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	00h			32-bit			**EC SIZE**	
POWER	VTR			0200h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R/W	R/W	R/W
BIT NAME	Reserved			TIMERx_ CLK_ REQ	SLEEP_E NABLE	Reserved	PD	Reserved
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	RLOAD	Reserved	UPDN	Reserved	MODE		RESET	ENABLE

ENABLE

Timer Enable - This bit is used to start and stop the timer. This bit does not reset the timer count but does reset the timer pulse output. This bit will be cleared when the timer starts counting in One-Shot mode.

0= Timer is disabled

1= Timer is enabled

Note:	The ENABLE bit is cleared after the RESET cycle is done. Firmware must poll the RESET bit to determine when the RESET cycle is done.

RESET

Timer Reset - This bit stops the timer and resets the internal counter to the value in the Timer Reload Register. This bit also clears the Timer Enable bit if it is set. This bit is self clearing after the timer is reset. Firmware must poll this RESET bit.

0= Normal timer operation

1= Timer reset

APPLICATION NOTE: When the RESET takes effect interrupts are blocked. Interrupts are not blocked until RESET takes effect and the ENABLE bit is cleared. If interrupts are not desired, firmware must mask interrupt in the interrupt block.

MODE

Timer Mode Select - These bits control the timer mode.

00= Timer Mode

01= Event Mode

10= One Shot Mode

11= Reserved

UPDN

Up/Down. In Event mode this bit selects the timer count direction.

Event Mode:

0= The timer counts down

1= The timer counts up

RLOAD

Reload Control. This bit controls how the timer is reloaded on overflow or underflow in Event and Timer modes, it has no effect in One Shot mode.

0=Roll timer over to FFFFh and continue counting when counting down and rolls over to 0000h and continues counting when counting up.

1=Reload timer from Timer Reload Register and continue counting.

PD

Power Down.

0= The timer is in a running state (default).

1= The timer is powered down and all clocks are gated.

SLEEP ENABLE

This bit is a read-only bit that reflects the state of the SLEEP ENABLE signal. This signal stops the timer and resets the internal counter to the value in the Timer Reload Register. Once the timer is disabled, the TIMERx_CLK_ bits will be deasserted. This signal does not clear the Timer Enable bit if it is set. If the timer is enabled, the counter will resume operation when the SLEEP ENABLE signal is deasserted. The timer is held in reset as long as the input signal is asserted.

0=Normal timer operation. In Normal Mode, the timer operates as configured. When returning from a sleep mode, if enabled, the counter will be restarted from the preload value.

1=Sleep Mode Requested. In Sleep Mode, the timer is reset, the counter is disabled, and the TIMERX_CLK_REQ outputs are deasserted.

TIMERX_CLK_REQ

The TIMERx_CLK_ bit is a read-only bit that reflects the state of the TIMERX_CLK_REQ output signal.

0= Indicates the 64.52MHz clock domain can be turned 'off' when appropriate

1= Indicates the 64.52MHz clock domain is required to be 'on.'

24.11.2 TIMER X CLOCK AND EVENT CONTROL REGISTER

TABLE 24-10: TIMER X CLOCK AND EVENT CONTROL REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	04h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				Reserved			
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
BIT NAME	EVENT	EDGE		Reserved	TCLK			

TCLK

This field is the Timer Clock Select, used to determine the clock source to the 16-bit timer. Available frequencies are shown in Table 24-11:

TABLE 24-11: TIMER CLOCK FREQUENCIES

Timer Clock Select	Frequency Selected
0000	64.52MHz
0001	32.26MHz
0010	16.13MHz
0011	8.06MHz
0100	4.03MHz
0101	2.01MHz
0110	1MHz
0111	500KHz
1xxx	Reserved

EDGE

Edge Type Select. In Event Mode, these bits are used to select the edge type that the timer counts. In One-Shot mode these bits select which edge starts the timer.

Event Mode:

00= Counts falling edges

01= Counts rising edges

10= Counts rising and falling edges

11= No event selected

One-Shot Mode:

00= Starts counting on a falling edge

01= Starts counting on a rising edge

10= Starts counting on a rising or falling edge

11= Start counting when the Enable bit is set

EVENT

Event Select - This bit is used to select the count source when the timer is operating in event mode.

0= Timer x-1 overflow is count source

1= Reserved

24.11.3 TIMER X RELOAD REGISTER

TABLE 24-12: TIMER X RELOAD REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	08h			32-bit			**EC SIZE**	
POWER	VTR			FFFFh			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	**…**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Timer Reload [15:0]							

Timer Reload

The Timer Reload register is used in Timer and One-Shot modes to set the lower limit of the timer. In Event mode the Timer Reload register sets either the upper or lower limit of the timer depending on if the timer is counting up or down. Valid Timer Reload values are 0001h - FFFFh. If the timer is running, the reload value will not be updated until the timer overflows or underflows.

Note:	Programming a 0000h as a preload value is not a valid count value.

24.11.4 TIMER X COUNT REGISTER

TABLE 24-13: TIMER X COUNT REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	0Ch			32-bit			**EC SIZE**	
POWER	VTR			FFFFh			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D14**	**D13**	**...**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Timer Count [15:0]							

Timer Count

The Timer Count register returns the current value of the timer in all modes.

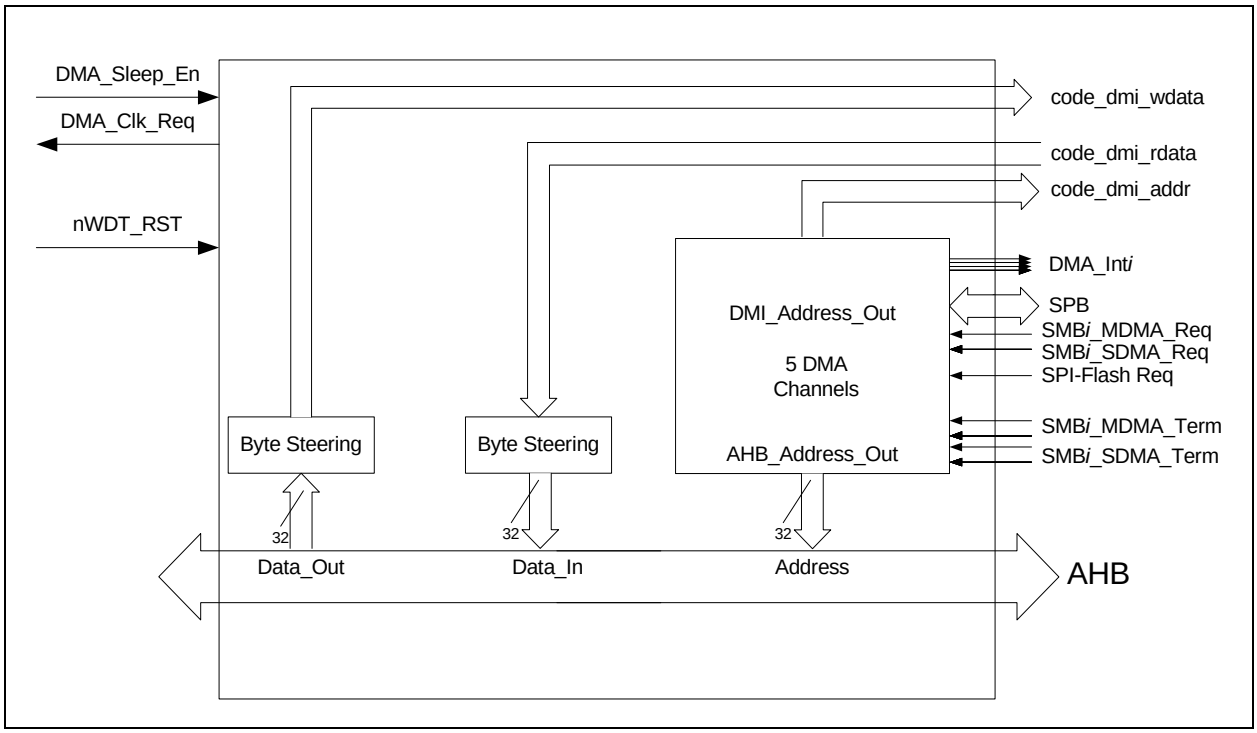
25.0 DMA CONTROLLER

25.1 General Description

This block describes the SCH5636 DMA controller. The DMA controller is designed to move data between the SMBus and SPI Flash controllers and the EC closely-coupled SRAM memory. There are five independent channels that each move byte-wide data between the SMBus and SPI Flash controllers and the SRAM, in either direction.

25.2 DMA Block Diagram

FIGURE 25-1: DMA BLOCK DIAGRAM



25.3 Block Diagram Signal List

TABLE 25-1: DMA CONTROLLER SIGNAL LIST

Signal Name	Direction	Description
AHB	I/O Bus	SCH5636 system bus
SPB	I/O Bus	SCH5636 peripheral bus
code_dmi_wdata, code_dmi_rdata, code_dmi_addr	I/O Bus	Direct Memory Interface (DMI) to ARC ICCM memory
MCLK	INPUT	Master SCH5636 clock
nWDT_RST	INPUT	Block reset signal
DMA_Int[1:5]	OUTPUT	DMA Interrupt signals
SMB[2:1]_MDMA_Req	INPUT	DMA request control from SMBus Master channel.
SMB[2:1]_SDMA_Req	INPUT	DMA request control from SMBus Slave channel.
SMB[2:1]_MDMA_Term	INPUT	DMA termination control from SMBus Master channel.
SMB[2:1]_SDMA_Term	INPUT	DMA termination control from SMBus Slave channel.

TABLE 25-1: DMA CONTROLLER SIGNAL LIST (CONTINUED)

Signal Name	Direction	Description
SPIFLASH_Read_Req	INPUT	DMA request control from SPI Flash receive Buffer register
SPIFLASH_Write_Req	INPUT	DMA request control from SPI Flash transmit Buffer register
DMA_SLEEP_EN	INPUT	External enable/disable signal used to put the block in the lowest power consumption state. 0=No Sleep Requested. The block should operate as configured. 1=Sleep Requested. The block enters sleep mode. See Low Power Mode on page 267.
DMA_CLK_REQ	OUTPUT	This output indicates when this block requires this clock input. 0= 64MHz can be turned 'off' when appropriate 1= 64MHz is required to be 'on.'

25.4 Power, Clocks and Reset

25.4.1 POWER DOMAIN

This block is powered by the VTR power supply.

25.4.2 CLOCKS

This block has three clock inputs: MCLK, the EC AHB bus clock enable, used for AHB transfers, and the EC clock enable, used for DMI transfers.

25.4.3 RESET

This block is reset on a nSYS_RST.

25.5 DMA Interrupts

Each channel of the DMA controller generates an interrupt event to the EC which indicate a DMA transfer is complete.

25.6 Low Power Mode

This block is designed to conserve power when it is either sleeping or disabled. There are two ways to put the DMA Controller into a low power mode: Disable all DMA channels via the Activate Bits or assert the DMA_SLEEP_EN signal to the DMA Controller. The following table summarizes the DMA Controller behavior for each of these low power modes.

TABLE 25-2: BLOCK CLOCK GATING IN LOW POWER MODES

Activate	DMA_SLEEP_EN	Busy	DMA_CLK_REQ	State	Description
All 0	X	X	0	INACTIVE	All channels are disabled
Any 1	0	X	1	NORMAL OPERATION	The block in neither disabled by firmware nor commanded to SLEEP
Any 1	1	1	1	PREPARING to SLEEP	The core clock is required until the current transaction is completed and the Block is IDLE.
Any 1	1	0	0	SLEEPING	The block is idle and the core clock can be stopped.

25.7 Operation

The SCH5636 features a five channel DMA controller. The DMA controller can autonomously move data from I/O devices to and from EC local memory without EC intervention.

The DMA has the following characteristics:

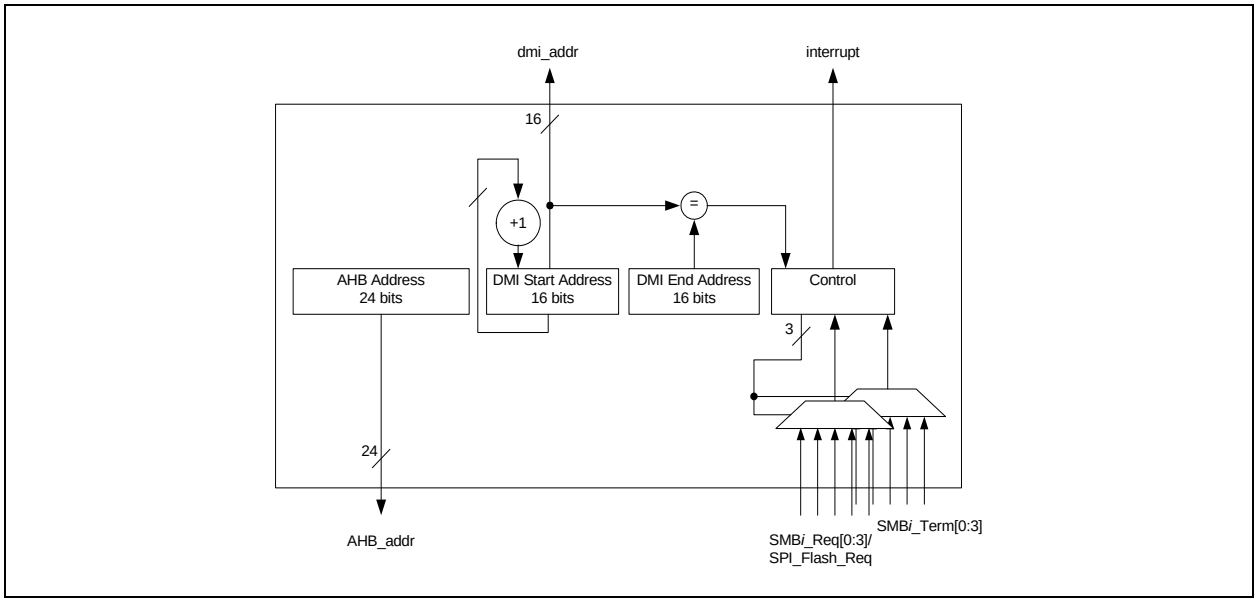
- Data is only moved 1 byte (8 bits) at a time
- Data only moves between devices on the AHB bus, or devices connected to an AHB bus bridge, and the EC SRAM. Since the SRAM is dual-ported to be both ICCM and DCCM, the DMA can be interpreted as moving data into and out of the ICCM as well as the DCCM.
- The number of DMA channels can be less than the number of I/O devices that can use the DMA for data transfers, so the DMA channels are shareable and can be assigned to any device.

The DMA controller is not designed to communicate with I/O devices with more than an 8-bit interface. The controller will accesses SRAM buffers only with incrementing addresses (that is, it cannot start at the top of a buffer, nor does it handle circular buffers automatically). The controller does not handle chaining (that is, automatically starting a new DMA transfer when one finishes).

25.7.1 DMA CHANNELS

Each DMA channel is capable of bi-directional data movement between an logical device and the EC closely-coupled memory. A single DMA channel is illustrated in Figure 25-2, "DMA Channel":

FIGURE 25-2: DMA CHANNEL



There are 5 possible logical devices that may connect to a DMA Controller channel. There are two SMBus controllers, each of which has a separate Master request and Slave request. In addition, the SPI Flash controller has a request. The SMBus controllers also provide a termination signals for each device. The SPI Flash device does not provide a termination signal, so termination inputs corresponding to a SP Flash request are always held to 0.

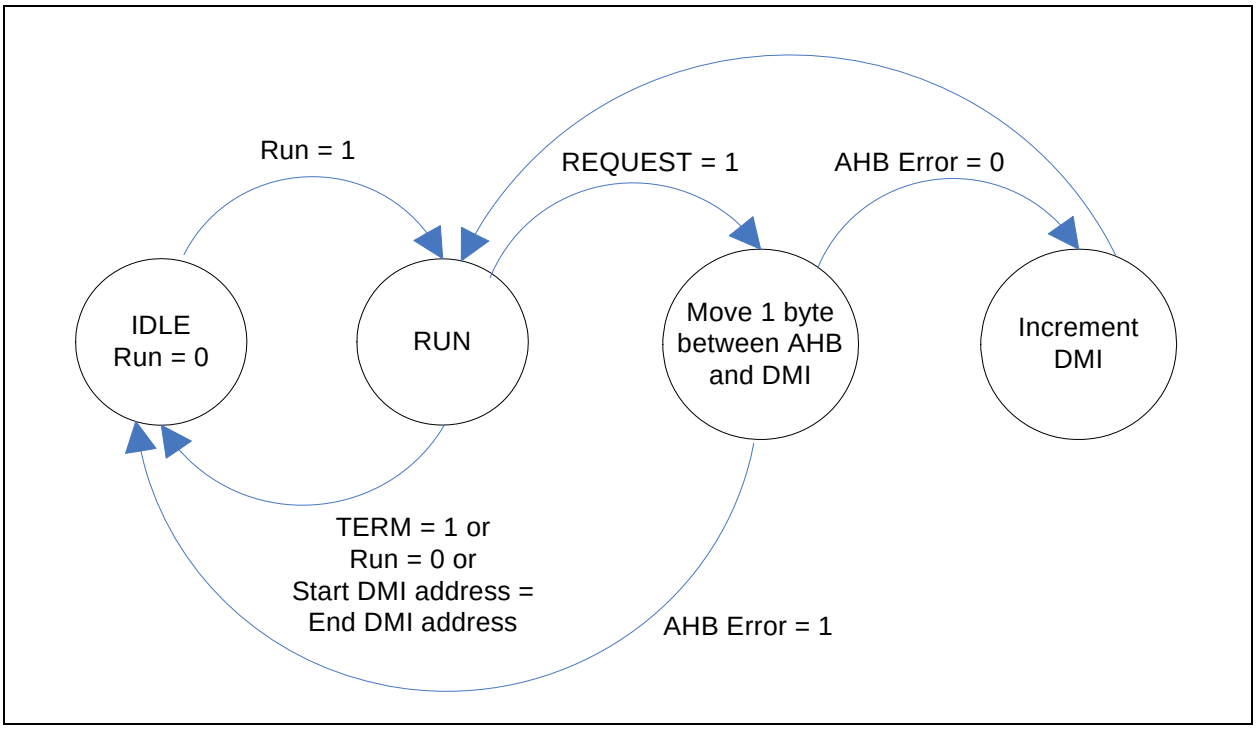
Based on the DIR in the channel's DMA Control Register, bytes are copied from a device's Receive Buffer to the ICCM/DCCM, or from the ICCM/DCCM to a device's Transmit Buffer. The AHB Address Register is programmed to be the address of the buffer required for the transfer. Software is responsible for insuring that this address is associated with the correct SRAM buffer defined by the DMI Start Address Register and by the correct SMBus as selected by the Device field in the Control register.

The DMI Start Address is an offset from the base of the SRAM. It is an offset from both the base of the DCCM and the base of the ICCM, since the two memories are different aliases of a single dual-ported SRAM. This register is loaded into a 16-bit counter, which increments by 1 under state machine control in order to generate the current DMI address.

The End Address register contains the address one greater than the last byte to transfer. The DMA transfer terminates when the current DMI address equals the End Address. If the DMA channel is configured with the Start Address and End Address registers set to the same address, no bytes will be transferred. If the End Address is configured with an address that is less than the Start Address, the DMI Start Address register will wrap around from FFFFh to 0000h. The DMA Controller always sends 16 address bits to the SRAM. If fewer than 16 bits are required to address the ICCM/DCCM, then the SRAM will ignore the upper bits of the Start Address and End Address registers. For example, if the ICCM/DCCM is 16KB, bits 14 to 15 in the Start and End Address registers will be ignored by the SRAM.

The state machine that runs each DMA channel is illustrated in Figure 25-3, "Channel State Machine". When software sets the bit to 1, the state machine enters the RUN state and waits for an assertion of the selected REQUEST input. As long as REQUEST is asserted when the state machine is in the RUN state the DMA controller starts a 1-byte wide AHB bus transaction, in the direction defined by DIR. After the AHB transaction completes, the current DMI address counter is incremented by 1 and compared to the End Address register. If the current address is not equal to the End Address, the state machine returns to the RUN state. If the current address is equal to the End Address, or if the TERM input is asserted, the DMA transaction is terminated and the state machine returns to the IDLE state. If there is an AHB bus error on the transfer between the AHB and the DMI, the DMI address counter increment is inhibited. The state machine returns to the IDLE state and the Status is set to AHB Bus Error.

FIGURE 25-3: CHANNEL STATE MACHINE



25.7.2 I/O DEVICES

The DMA Controller is configured to work with either of the two SMBus controllers and the SPI Flash controller on the SCH5636. Table 25-3, "DMA Device Selection" shows the mapping between the Device and DIR fields for each channel and each device.

TABLE 25-3: DMA DEVICE SELECTION

Device	DIR	SMBus Controller	Data AHB Address	Request Signal
0	0	SMBus Controller 1	F0_184Ch	SMB1_SDMA_Req
0	1	SMBus Controller 1	F0_1848h	SMB1_SDMA_Req
1	0	SMBus Controller 1	F0_1854h	SMB1_MDMA_Req
1	1	SMBus Controller 1	F0_1850h	SMB1_MDMA_Req

TABLE 25-3: DMA DEVICE SELECTION (CONTINUED)

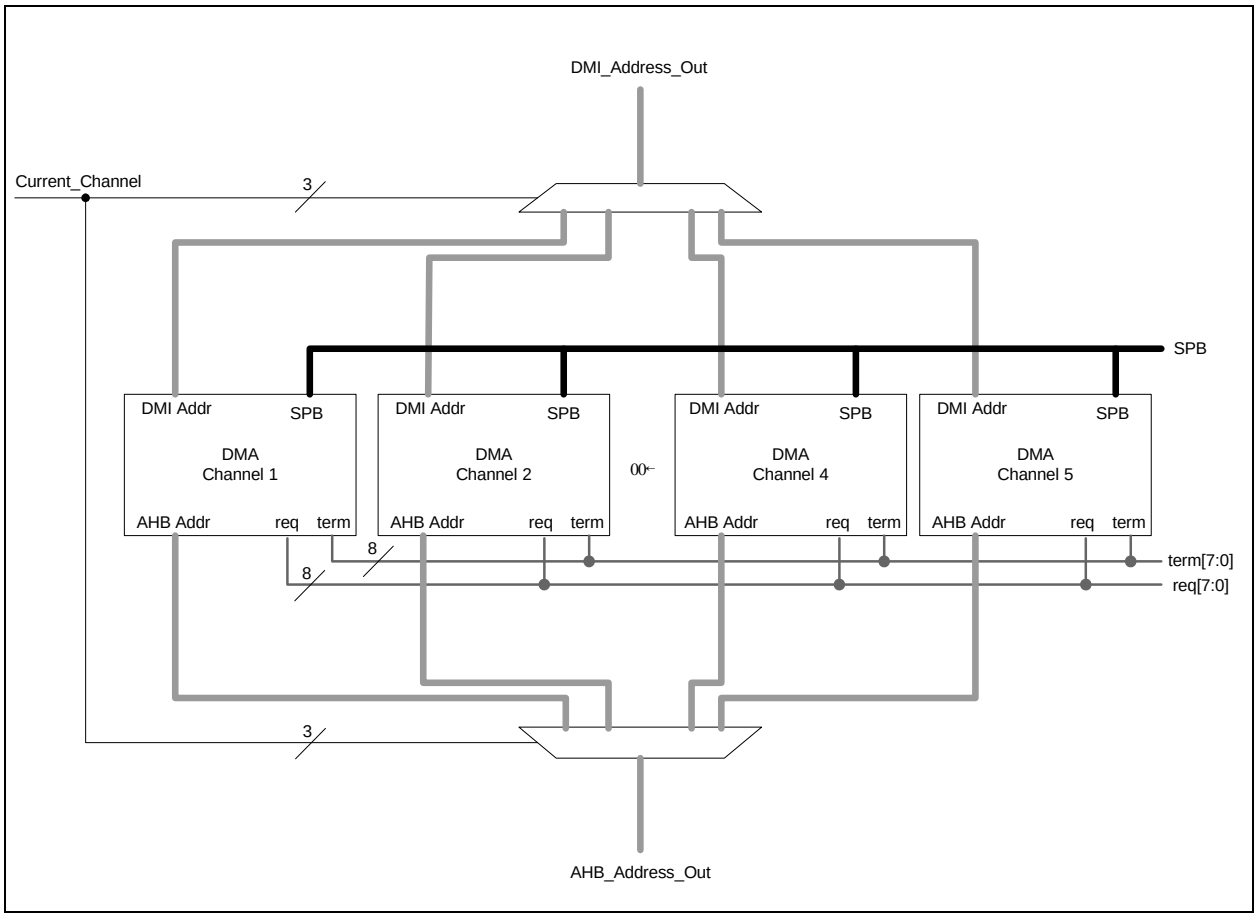
Device	DIR	SMBus Controller	Data AHB Address	Request Signal
2	0	SMBus Controller 2	F0_18CCh	SMB2_SDMA_Req
2	1	SMBus Controller 2	F0_18C8h	SMB2_SDMA_Req
3	0	SMBus Controller 2	F0_18D4h	SMB2_MDMA_Req
3	1	SMBus Controller 2	F0_18D0h	SMB2_MDMA_Req
4	0	SPI Flash Controller	FF_3C10h	SPIFLASH_Req
4	1	SPI Flash Controller	FF_3C0Ch	SPIFLASH_Req

The Request signals from the devices are the Data register status signals. In the read (0, from the device to SRAM) direction, a DMA request is asserted as long as the Receive Data register is not empty. In the transmit (1, from the SRAM to the device) direction, a DMA request is asserted as long as the Transmit Data register is not full. For the SMBus controllers, the Terminate signal is asserted if the SMBus controller detects an error condition during an SMBus transaction, or if software shuts down the SMBus controller. There is no Terminate signal for the SPI Flash controller.

25.7.3 DMA CHANNEL ARBITRATION AND MULTIPLEXING

The five DMA channels share the DMI interface and the AHB bus. Figure 25-4, "DMA Channel Multiplexing" illustrates the multiplexing of the channels onto the busses.

FIGURE 25-4: DMA CHANNEL MULTIPLEXING



A DMA Channel is ready to run as long as it is in the Run state and its selected Request input is asserted. The DMA controller services DMA Channels on a first-come first-served basis. If two channels become ready to run simultaneously, they are served in numerical order (channel 0 before channel 1, etc.). If multiple channels are continuously ready (that is, their respective Request inputs are always asserted), then they will be served in round-robin order.

25.8 DMA Registers

The base address for the DMA Controller block in the AHB address space is listed in Table 25-4, "DMA Controller Base Address Table".

TABLE 25-4: DMA CONTROLLER BASE ADDRESS TABLE

DMA Controller Instance	LDN from (Table 4-3 on page 27)	AHB Base Address
DMA Channel1	9h	F0_2400h + 000h
DMA Channel2		F0_2400h + 020h
DMA Channel3		F0_2400h + 040h
DMA Channel4		F0_2400h + 060h
DMA Channel5		F0_2400h + 080h

The following table summarizes the registers allocated for the DMA Controller. The offset field in the following table is the offset from the AHB Base Address defined in Table 25-4 on page 271.

TABLE 25-5: DMA CONTROLLER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
DMA Control Register	-	00h	32	R/W	
DMI End Address Register	-	04h	32	R/W	
DMI Start Address Register	-	08h	32	R/W	
AHB Address Register		0Ch	32	R/W	
DMA Activate Register		10h	32	R/W	

25.8.1 DMA CONTROL REGISTER

The DMA Control Register is used to control the behavior of the DMA controller.

TABLE 25-6: DMA CONTROL REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	00h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R/W	R/W	R/W	R/W
BIT NAME	Reserved				Device			DIR
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved		Busy	Status		Done	Request	Run

Run

When this bit is 1, the state machine is active and the channel continually tries to move a byte between the I/O device at AHB Address and SRAM at DMA Address. When this bit is 1, software cannot modify any of the other registers in the channel, or any other bits in this register besides, in order to insure that no AHB transaction is modified while it is in progress.

Setting this bit to 0 will halt the DMA function. If there is an AHB transfer in progress when this bit is set to 0 the transfer will complete before the DMA state machine returns to the IDLE state. Firmware should query the Busy bit after setting Run to 0 in order to determine when the DMA transaction has terminated.

The DMA_Int signal is only asserted when is 1.

Request

Read-only. This bit is always 0 when Run is 0, and is set when the DMA request input is 1. The DMA request input is selected by Device and DIR.

Done

Read-only. This bit is always 0 when is Run is 0, and is 1 when the DMA Controller state machine returns to the IDLE state. The DMA Controller state machine will transition back to IDLE when DMI Start Address Register equals the DMI End Address Register, when the DMA Termination input is 1 or if the AHB transaction is terminated with a bus error. The DMA Termination input is selected by Device and DIR. This bit is routed to the interrupt controller.

Status

Read-only. This field is updated whenever Done goes from 0 to 1 or when Run goes from 1 to 0, and indicates why a DMA transfer completed. Status values are:

00:Run is set to 0. This field is always 0 when Run is 0.

01:Start Address matched End Address

10:DMA_Term input asserted

11:An AHB bus error occurred on the transfer

Status values are shown in order of priority. If more than one condition caused a return to the IDLE state, the condition with the lowest Status value is reported.

Busy

Read-only. This bit is 1 when the DMA State Machine is not in the IDLE state and 0 when the DMA State Machine is in the IDLE state.

DIR

DMA transfer direction. 0 for reads from the AHB device to DCCM memory, 1 for writes from DCCM memory to the AHB device. When combined with Device, determines which DMA_request input is used to start a DMA transfer

Device

This field selects which I/O device is assigned to this DMA channel. Choices are:

000:SMBus Controller 1 Slave

001:SMBus Controller 1 Master

010:SMBus Controller 2 Slave

011:SMBus Controller 2 Master

100:SPI Flash Controller

25.8.2 DMI END ADDRESS REGISTER

This address defines the DMA stops transferring. When the incrementer that was loaded from the DMI Start Address Register is equal to this register, the DMA completes.

TABLE 25-7: DMI END ADDRESS REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC OFFSET	04h					32-bit	**EC SIZE**	
POWER	VTR					0000h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	DMI_End_Address[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	DMI_End_Address[7:0]							

DMI_End_Address

This field contains the address one past the last byte to be transferred for the DMA channel. The DMA transfer stops when the current DMI address is equal to this register. If the End Address register is equal to the Start Address register when is set to 1, no data are transferred and the DMA terminates immediately.

25.8.3 DMI START ADDRESS REGISTER

Note: This register is 16-bit only. It does not support 8-bit accesses.

TABLE 25-8: DMI START ADDRESS REGISTER

HOST OFFSET	N/A					N/A	**HOST SIZE**	
EC ADDRESS	08h					32-bit	**EC SIZE**	
POWER	VTR					0000h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	DMI_Start_Address[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	DMI_Start_Address[7:0]							

DMI_Start_Address

This field defines an offset from the base of the SRAM, which is an offset from both the base of the DCCM and the base of the ICCM, since the two memories are different aliases of a single dual-ported SRAM. This register is loaded into a 16-bit counter, which increments by 1 under state machine control. This register defines the initial byte address for bytes to be transferred on the associated DMA channel.

When first written by software, this register contains the start address in the DCCM for the DMA transfer. While a DMA transfer is in progress, this register contains the address of the next byte to be transferred. When the DMA transfer completes, this register is one greater than the address of the last byte transferred. Software can determine how many bytes were transferred overall by subtracting the value it used to configure this register initially from the value of this register when the transfer completes.

25.8.4 AHB ADDRESS REGISTER

The AHB Address Register is the address of the I/O device that is the source or sink of the DMA transfer.

TABLE 25-9: AHB ADDRESS REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC ADDRESS	0Ch			32-bit			**EC SIZE**	
POWER	VTR			0000_0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	AHB_Address[23:16]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	AHB_Address[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	AHB_Address[7:0]							

AHB_Address[23:0]

This is the address of the I/O port in the AHB address space. Software is responsible for insuring that this address is the correct address for the I/O device assigned to the channel.

25.8.5 DMA ACTIVATE REGISTER

The DMA Activate Register is used to gate clocks to a DMA channel, in order to conserve power. Software must set the Activate bit to '1b' in order for a channel to operate.

TABLE 25-10: DMA ACTIVATE REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	10h			32-bit			**EC SIZE**	
POWER	VTR			00h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R/W
BIT NAME	Reserved							Activate

Activate

When this bit is 0, the MCLK is gated to this channel, so the channel will not operate. When this bit is 1, the channel is provided with the system clock and the channel can operate.

26.0 SMB DEVICE INTERFACE

26.1 General Description

The SCH5636 SMB Device Interface includes two instances of an SMBus controller core: SMBus 1 and SMBus 2. This chapter describes aspects of the SMB Device Interface that are unique to the SCH5636 instantiations of this core; including, Power Domain, Resets, Clocks, Interrupts, Registers and the Physical Interface. For a *General Description*, *Features*, *Block Diagram*, *Functional Description, Registers Interface* and other core-specific details, see Ref [1] (note: in this chapter, *italicized text* typically refers to SMBus controller core interface elements as described in Ref [1]).

26.1.1 REFERENCES

1. SMBus Controller Core Interface, Revision 2.0, Core-Level Architecture Specification, SMSC, 6/18/08

26.2 Power, Clocks and Reset

26.2.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains. For more detail about the SMBus controller core Power Domain, see *Section 3.1, "Power Configuration"* in Ref [1].

26.2.2 CLOCKS

SMB Device Interface Clocking is described below inTable 26-1. Use this table when programming the SMBus controller core bus clock and timing values as specified in Ref [1].

TABLE 26-1: SMBUS INTERFACE CLOCKING

Clock Source	SMBus Controller Core Clock (Note 26-1)	Frequency	Description
MCLK	CORE_CLK	64.52 MHz	–
MCLK_DIV8_EN	BAUD_CLK_EN	8.06 MHz	Use this frequency when programming the *Bus Clock Register*, *Data Timing Register* and the *Time-Out Scaling Register* described in Ref [1].
EC Bus Clock	SPB_CLK_EN	Programmable	EC Bus Clock.

Note 26-1 For more detail about SMBus controller core Clocking see *Chapter 2, "Hardware Interface"* and *Section 3.3,* "*Clocking*" in Ref [1].

26.2.3 RESETS

Each of the SMBus controller core instances in the SCH5636 SMB Device Interface are reset by nWDT_RST. See Section 5.9, "Registers," on page 43 for details on resets in the SCH5636. For more detail about SMBus controller core Resets, see *Section 3.2,* "*Reset Interface*" in Ref [1].

26.3 Interrupts

Each EC SMB Controller has an activity interrupt event and each SMB port has a Wake-up interrupt event. The SMB activity interrupt events are routed to the SMB1 and SMB2 bits in the GIRQ12 Source register. The port Wake-up events are triggered by transitions on the SMB Data signal and are routed to the SMB1 WK, LVSMB1 WK and SMB2 WK bits in the GIRQ12 Source register. The edge detection of the wake events are controlled by their associated pin control registers described in Section 22.0, "GPIO Interface," on page 240

APPLICATION NOTE: The GPIO pin control register for the GPIO associated with the SMBus Data signal for an SMB wakeup event should be programmed to Input, Falling Edge Triggered, non-inverted polarity detection.

26.4 Registers

Each SMBus controller core instance in the SCH5636 SMB Device Interface has unique Register Interface Addressing, defined by a base address as indicated in Table 26-2. For more detail about SMBus controller core registers, see *Chapter 5, "Registers Interface"* in Ref [1].

TABLE 26-2: SMBUS INTERFACE BASE ADDRESS

SMB Device Interface **Instance**	**LDN from (Table 4-3 on page 27)**	**AHB Base Address**
SMB1	6h	F0_1800h
SMB2		F0_1880h = F0_1800h + 80h

26.5 Physical Interface

26.5.1 SMBUS 1

SMBus Controller 1 has two physical ports, selected by the *PORT SEL [3:0]* bits in the *Configuration Register* as described in Ref [1]. Port 0 is the SMBCLK1/SMBDAT1 pair, which are 3.3V pins. Port 1 is the LVSMBCLK1/LVSMBDAT1 pair, which are low voltage pins referenced to the Vtt processor reference voltage.

The buffer type for these pins must be configured as open-drain outputs in the GPIO Configuration registers associated with the GPIO signals that share the ports.

26.5.2 SMBUS 2

SMBus Controller 2 has one port, SMBCLK2/SMBDAT2. The buffer type for these pins must be configured as open-drain outputs in the GPIO Configuration registers associated with the GPIO signals that share the port.

27.0 TACH MONITOR

27.1 General Description

This block is designed to monitor tach output signals or locked rotor signals from various types of fans to determine their speed. One mode returns the value in number of CLOCK_LOW pulses. Another mode returns the value in pulses per programmed amount of time. This second mode can use the raw tach input. Each Tach is associated with a pair of limit registers that define maximum and minimum acceptable Tach counter values. If the readings on a Tach is outside these limits an interrupt to the EC can be generated.

In typical systems the fans are powered by the main power supply. Firmware may disable this block when it detects the main power rail has been turned off.

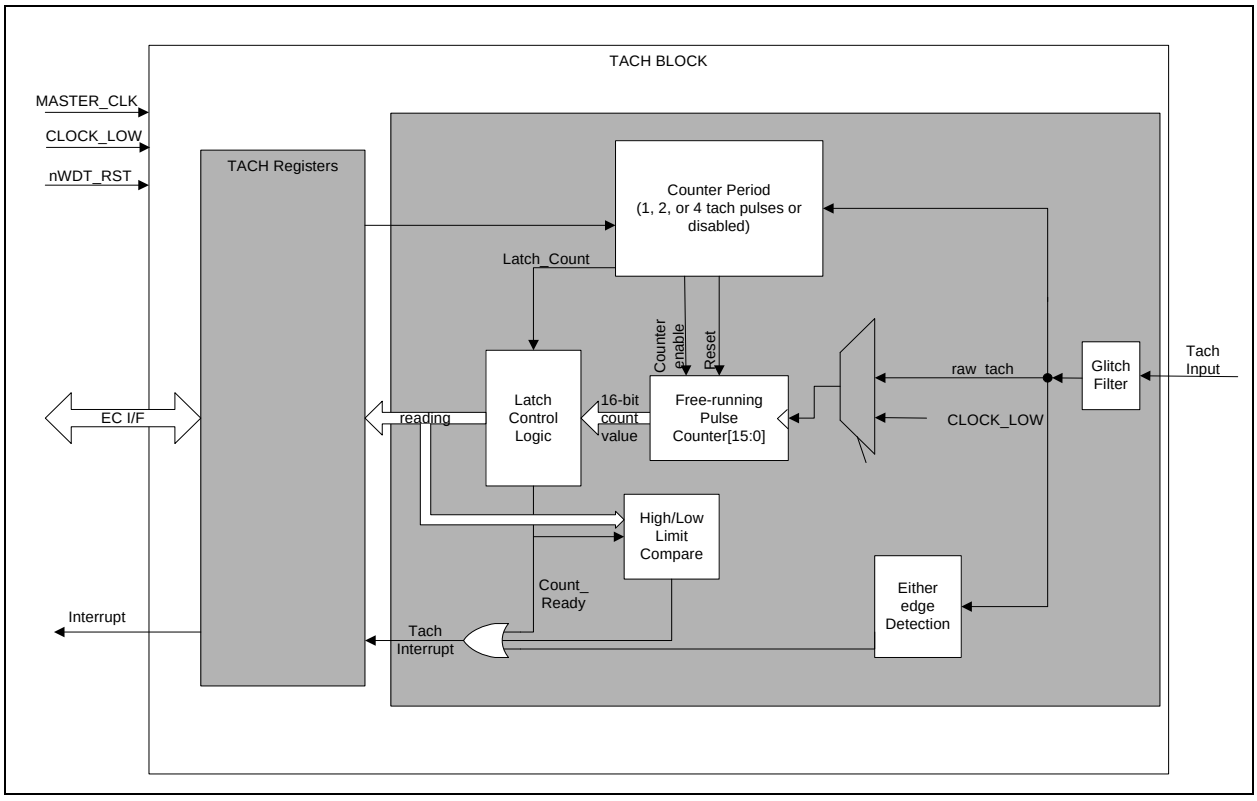
This block can be utilized with fans running at speeds ranging from 100 RPM to 30K RPM.

The TACH Monitor performs the following functions:

- Count the number of pulses detected on the raw tach input.
- Count the number of clocks for a programmed number of pulses.
- Generate an interrupt when the count value is latched into the reading register.
- Generate a programmable either-edge triggered interrupt for detecting when the tach input changes state. This may be used for Locked Rotor detection.
- Generate an interrupt when the count value latched into the reading register is greater than the high limit or less than the low limit.

27.2 TACH Monitor Block Diagram

FIGURE 27-1: BLOCK DIAGRAM OF TACH MONITOR



Note:	Once the counter is enabled it is a 16-bit free-running counter. The count value is latched on a read or when the programmed number of tach pulses is detected (if enabled), for 1, 2, or 4 pulses. The counter is reset to 0000h if the count value is latched by a programmed number of tach pulses as well as on a VCC POR. The counter enable is a software controlled signal.

27.3 Block Diagram Signal List

TABLE 27-1: TACH PORT LIST

Signal Name	Direction	Description
nWDT_RST	INPUT	Watchdog timer reset
Master_Clock	INPUT	64.52MHz MCLK
CLOCK_LOW	INPUT	90KHz
EC I/F	I/O Bus	EC-side SPB bus
Tach Input	INPUT	Tachometer signal from TACHx Pin
Interrupts	OUTPUT	Interrupt used to indicate that either the Tach Input has changed state or the TACH reading has been updated. One per TACH

27.4 Power, Clocks and Reset

27.4.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

27.4.2 CLOCKS

This block uses the EC Bus Clock and CLOCK_LOW, the 90KHz clock. EC Bus Clock is used when reading and writing the TACH Monitor control registers. The individual TACH counters are driven by CLOCK_LOW.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

27.4.2.1 Clock Idle

When the internal ring oscillator is disabled or when the TACH block is disabled, the internal TACH counters are reset. The reading register is not affected. This insures that inaccurate readings are not generated if the master clock halts in the middle of a TACH reading or when the TACH starts up.

Note:	Each Tach pin should be pulled up via an external resistor to the main power supply.

27.4.3 RESET

This block is reset on a nWDT_RST.

See Section 5.9, "Registers," on page 43 for details on reset.

27.5 TACH Interrupts

Each TACH Monitor in the SCH5636 can be used to generate one interrupt event. Each TACH Monitor interrupt source is a level, active high signal. The three TACH Monitor interrupts are routed to the TACH4, TACH3, TACH2, & TACH1 bits in the GIRQ17 Source register.

27.5.1 TACH INTERRUPT SOURCES

bit There are three interrupt source events: when there is an update of the TACH Counter register, when the TACH input toggles, or when the TACH Counter register exceeds a programmed limit. The corresponding interrupt status bits are Count Ready Status, Toggle Status and Tach Out-of-Limit Status (Bits[3,1,0] in TACHx Status Register), respectively.

27.5.1.1 Count Reading Ready Status

This status bit is asserted when the counter value is latched. The bit is located in Bit D3 of the TACHx Status Register.

27.5.1.2 Tach Input Toggle Status

This status bit is asserted when the Tach input changes state. The bit is located in Bit D2 of the TACHx Status Register.

27.5.1.3 TACH Out-of_Limit STATUS

To generate a TACH out-of-limit status event, the high and low limits may be programmed in the TACHx High Limit Register and TACHx Low Limit Register. An out-of-limit event is triggered when the reading register (Tachx Counter in the TACHx Control Register) is set to a value less than the TACHx Low Limit Register or to a value greater than the TACHx High Limit Register. If the value in the Tachx Counter violates the programmed limits the TACH limit registers a status event will be generated, indicating the out-of-limit event. This status bit is implemented in Bit D0 of the TACHx Status Register. This signal may be used to interrupt the Embedded Controller, if enabled via TACH Out-of-Limit Enable (Bit D0 of the TACHx Control Register).

Note:	If the TACHx Low Limit Register is set to 0000h, no out-of-limit event will be triggered by a Tachx Counter value that is below the limit. If the TACHx High Limit Register is set to FFFFh, no out-of-limit event will be triggered by a Tachx Counter value that is above the limit.

APPLICATION NOTE: Out-of-Limit checks are typically only used when the tach counter is incremented in Mode 1 (in which the counter counts the number of CLOCK_LOW until a programmed number of pulses occur on the raw tach input).

27.6 TACH Circuitry

The TACH Circuitry is implemented with an internal pulse counter. There are two toggling signals that can be used to increment the counter: the raw tach input or CLOCK_LOW. See FIGURE 27-1: Block Diagram of TACH Monitor on page 278. The two modes for incrementing the counter are controlled by Tach Reading Mode Select in the TACHx Control Register.

If the raw tach input is used to increment the counter, the circuitry can be configured as a free-running counter that increments when a pulse from the tach is detected (i.e., input signal transitions from low-to-high).The counter is latched into the reading register (Tachx Counter in the TACHx Control Register) every time it is incremented. If this mode is selected, firmware will monitor the number of pulses detected over a period of time to determine the speed of the attached fan.

If CLOCK_LOW is used to increment the internal counter, the raw tach input will be used to determine when to latch the current count value into the reading register and reset the internal counter to 0000h. The counter is latched after a programmed number of tach pulses is detected. The programmed period can be configured to be 1, 2, or 4 tach pulses in duration.

Each Tach counter has comparison logic to compare the counter value with the high limit and low limit registers.

27.7 Registers

There are four block instances defined in this chapter: TACH[1,2,3, 4].

Each instance of the TACH Monitor is instantiated on the EC AHB and has its own Logical Device Number, and Base Address as indicated in Table 27-2.

TABLE 27-2: TACH MONITOR BASE ADDRESS

TACH Monitor **Instances**	**LDN**	**AHB Base Address**
TACH1	18h	F0_6000h
TACH2		F0_6080h
TACH3		F0_6100h
TACH4		F0_6180h

Table 27-3 is a register summary for one instance of the TACH Monitor.

TABLE 27-3: TACH MONITOR REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
TACHx Control Register	-	00h	32	R/W	
TACHx Status Register	-	04h	32	R/W	
TACHx High Limit Register	-	08h	32	R/W	
TACHx Low Limit Register		0Ch	32	R/W	

27.7.1 TACHX CONTROL REGISTER

TABLE 27-4: TACHX CONTROL REGISTER

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	00h				32-bit		**EC SIZE**	
POWER	VTR				0000_0000h		nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	TACHx Counter [15:8] Register							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	TACHx Counter [7:0] Register							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Tach Input INT_EN	Count Ready INT_EN	Reserved	Tach Edges		Tach Reading Mode Select	Reserved	Filter Enable
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R/W	R/W
BIT NAME	Reserved						TACH Enable	Tach Out-of-Limit Enable

TACH Out-of-Limit Enable

The TACH Out-of_Limit Enable is used to enable the TACH Out-of_Limit Status bit in the TACHx Status Register to generate an interrupt event.

0= disable interrupt output from tach block (default)

1= enable interrupt output from tach block

TACH Enable

This bit enables the TACH logic.

0= TACH Idle, clocks gated (default)

1= TACH Monitoring enabled, clocks enabled.

APPLICATION NOTE: This bit gates the clocks into the block. When clocks are gated, the TACHx pin is tristated. When re-enabled, the internal counters will continue from the last known state and stale status events may still be pending. Firmware should discard any status or reading values until the reading value has been updated at least one time after the enable bit is set.

Filter Enable

This filter is used to remove high frequency glitches from Tach Input. When this filter is enabled, tach input pulses less than three CLOCK_LOW periods wide get filtered.

0= Filter disabled (default)

1= Filter enabled

Tach Reading Mode Select

0= Counter is incremented when Tach Input transitions from low-to-high state (default)

1= Counter is incremented on the rising edge of the CLOCK_LOW input. The counter is latched into Tachx Counter and reset when the programmed number of edges is detected.

Tach Edges

A tach signal is a square wave with a 50% duty cycle. Typically, two tach periods represents one revolution of the fan. A tach period consists of three tach edges.

This programmed value represents the number of tach edges that will be used to determine the interval for which the number of CLOCK_LOW pulses will be counted

00= 2 Tach edges (1/2 tach period)

01= 3 Tach edges (1 tach period)

10= 5 Tach edges (2 tach periods)

01= 9 Tach edges (4 tach periods)

Count Ready INT_EN

0= disable Count Ready interrupt from tach block (default)

1= enable Count Ready interrupt from tach block

Tach Input INT_EN

0= disable Tach Input toggle interrupt from tach block (default)

1= enable Tach Input toggle interrupt from tach block

Tachx Counter

This 16-bit field contains the latched value of the internal tach pulse counter, which may be configured by the Tach Reading Mode Select field to operate as a free-running counter or to be gated by the tach input signal.

If the counter is free-running (Mode 0), the internal tach counter increments (if enabled) on transitions of the raw tach input signal and is latched into this field every time it is incremented. The act of reading this field will not reset the counter, which rolls over to 0000h after FFFFh. The firmware will compute the delta between the current count reading and the previous count reading, to determine the number of pulses detected over a programmed period.

If the counter is gated by the tach input and clocked by CLOCK_LOW (Mode 1), the internal counter will be latched into the reading register when the programmed number of edges is detected or when the counter reaches FFFFh. The internal counter is reset to zero after it is copied into this register.

APPLICATION NOTE: In Mode 1, a counter value of FFFFh means that the tach did not detect the programmed number of edges in 728ms. A stuck fan can be detected by setting the TACHx High Limit Register to a number less than FFFFh. If the internal counter then reaches FFFFh, the reading register will be set to FFFFh and an out-of-limit interrupt can be sent to the EC.

27.7.2 TACHX STATUS REGISTER

TABLE 27-5: TACHX STATUS REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	04h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3-1 BIT	**D31**	**D32**	**D31**	**…**		**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R/WC	R/WC	R	R/WC
BIT NAME	Reserved				Count Ready Status	Toggle Status	TACH Pin Status	Tach Out-of-Limit Status

TACH Out-of-Limit STATUS

This bit is set when the Tach Count value is greater than the high limit or less than the low limit. It is cleared when written with a '1'. To disable this status event set the limits to their extreme values. If TACH Out-of-Limit Enable in the TACHx Control Register is set to '1', this status bit will assert the Tach Interrupt signal.

0= Tach is within limits

1= Tach is outside of limits

Tach Pin Status

This bit reflects the state of Tach Input. This bit is a read only bit that may be polled by the embedded controller.

0= Tach Input is low

1= Tach Input is high

Toggle Status

This bit is set when Tach Input changes state. It is cleared when written with a '1'. If Tach Input INT_EN in the TACHx Control Register is set to '1', this status bit will assert the Tach Interrupt signal.

0=Tach stable (default)

1=Tach Input changed state (this bit is set on a low-to-high or high-to-low transition)

APPLICATION NOTE: Some fans offer a Locked Rotor output pin that generates a level event if a locked rotor is detected. This bit may be used in combination with the tach pin status bit to detect a locked rotor signal event from a fan.

APPLICATION NOTE: Tach Input may come up as active for Locked Rotor events. This would not cause an interrupt event because the pin would not toggle. Firmware must read the status events as part of the initialization process, if polling is not implemented.

Count Ready Status

This bit remains cleared to '0' when the Tach Reading Mode Select bit in the TACHx Control Register is '0'.

When the Tach Reading Mode Select bit in the TACHx Control Register is set to '1', the Count Ready Status bit is set when the counter value is latched by the hardware. It is cleared when written with a '1'. If Count Ready INT_EN in the TACHx Control Register is set to 1, this status bit will assert the Tach Interrupt signal.

0= Reading not ready

1= Reading ready

27.7.3 TACHX HIGH LIMIT REGISTER

TABLE 27-6: TACHX HIGH LIMIT REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSETS	08h			32-bit			**EC SIZE**	
POWER	VTR			FFFFh			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE[3:2] BIT	**D31**	**D30**	**D29**	**...**		**D18**	**D17**	**D16**
HOST TYPE	R	R	R	R	R	R	R	R
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE1-0 BIT	**D15**	**D30**	**D29**	**...**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	TACHx High Limit [15:0]							

TACHx High Limit

The TACHx High Limit value is compared with the value in the TACHx Control Register. If the value in the TACHx Control Register is greater than the value programmed in the TACHx High Limit Register the TACH Out-of-Limit STATUS bit will be set. The TACH Out-of-Limit status event may be enabled to generate an interrupt to the embedded controller via the TACH Out-of-Limit Enable bit in the TACHx Control Register.

Note:	To disable the Tach out-of-limit high event, program FFFFh into this register.

27.7.4 TACHX LOW LIMIT REGISTER

TABLE 27-7: TACHX LOW LIMIT REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSETS	0Ch			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3-2 BIT	**D31**	**D32**	**D31**	**...**		**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BIT	**D15**	**D30**	**D29**	**...**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	TACHx Low Limit [15:0							

TACHx Low Limit

The TACHx Low Limit value is compared with the value in the Tachx Counter Field of the TACHx Control Register. If the value in the Tachx Counter Field is less than the value programmed in the TACHx Low Limit Register the TACH Out-of-Limit STATUS bit will be set. The TACH Out-of-Limit status event may be enabled to generate an interrupt to the embedded controller via the TACH Out-of-Limit Enable bit in the TACHx Control Register.

To disable the Tach out-of-limit low event, program 0000h into this register.

28.0 PWM CONTROLLER

28.1 General Description

The function of this block is to generate a PWM output that may be used to control 4-wire fans, blink LEDs, and so on. Each PWM can generate an arbitrary duty cycle output at frequencies from less than 0.1Hz to 32MHz. The PWM controllers are also used to generate the PROCHOT output and the Speaker output.

The PWMx Counter ON Time registers and PWMx Counter OFF Time registers determine the operation of the PWM_OUTPUT signals. See Section 28.3.1, "PWMx Counter ON/OFF Time Registers," on page 289 for a description of the PWM_OUTPUT signals.

28.1.1 PWM_OUTPUT

The PWM_OUTPUT signal is used to generate a duty cycle at a frequency. This block has been designed such that the PWM signal may be programmed to hold PWM_OUTPUT high, to hold PWM_OUTPUT low, or to toggle PWM_OUTPUT. If the PWM is configured to toggle, then PWM_OUTPUT will alternate high and low for the programmed duration in the PWMx Counter ON/OFF Time Registers registers as defined in the register description. The PWM equations are described in Figure 28-1.

28.1.2 PWM FEATURES

APPLICATION NOTE: Each PWM pin signal functions is muxed with a GPIO pin signal function. The pin's default signal function is GPIO input as controlled by the associated Pin Control Register. (See Section 22.0, "GPIO Interface," on page 240). At VTR POR or when a WDT event occurs (see Section 23.0, "Watchdog Timer," on page 250,) the pin will tristate. For fan applications, an external resistor termination can provide the pin state to force the external fans to the full on state, thereby protecting the system from overheating.

28.1.3 PWM CONTROLLER BLOCK DIAGRAM

FIGURE 28-1: PWM DUTY CYCLE EQUATION

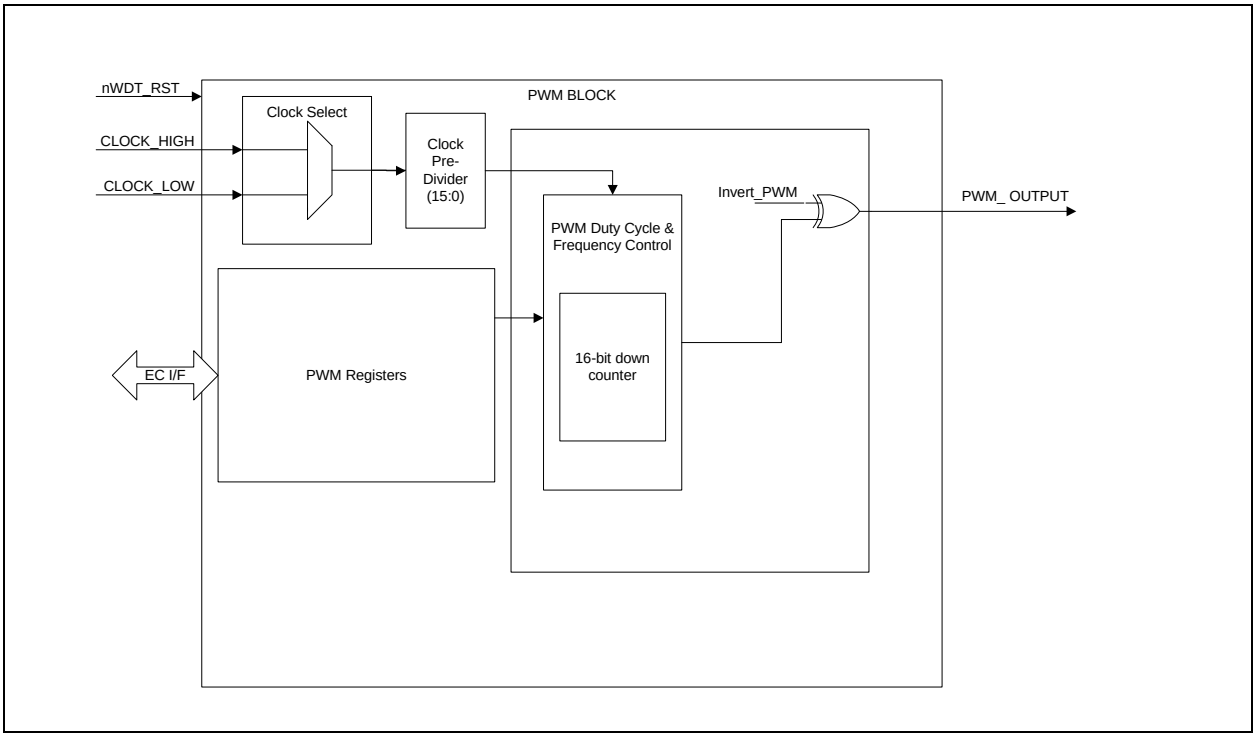
$$\text{PWM Duty Cycle} = \frac{PWMCounterONTime}{(PWMCounterONTime \div PWMCounterOFFTime)}$$

FIGURE 28-2: PWM FREQUENCY EQUATION

$$\text{PWM Frequency} = \frac{1}{(PreDivisor + 1)} \times \frac{((\overline{ClockSelect} \wedge 64.52MHz) \vee (ClockSelect \wedge 90KHz))}{(PWMCounterOffTime + PWMCounterOnTime)}$$

In Figure 28-2, the symbol $\wedge$ is the Boolean AND operator and the symbol $\vee$ is the Boolean OR operator. The variables PreDivisor, PWMCounterONTime, PWMCounterOFFTime and ClockSelect are registers that are defined in Section 28.3, "Registers".

FIGURE 28-3: BLOCK DIAGRAM OF PWM CONTROLLER



28.1.3.1 Block Diagram Signal List

TABLE 28-1: BLOCK DIAGRAM SIGNAL LIST DESCRIPTION

Signal Name	Direction	Description
nWDT_RST	INPUT	Watchdog timer reset
CLOCK_HIGH	INPUT	64.52MHz MCLK
CLOCK_LOW	INPUT	90KHz
PWM_OUTPUT	OUTPUT	Pulse Width Modulated signal to PWMx pin.
E/C IF	I/O Bus	EC-side SPB bus

28.2 Power, Clocks and Reset

28.2.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

28.2.2 CLOCKS

This block uses the EC Bus Clock, the 64.52MHz MCLK and the 90KHz clock. The EC Bus Clock is used when reading and writing the PWM Controller control registers. The individual PWM counters can be driven either by MCLK or the 90KHz clock.

See Section 5.9, "Registers," on page 43 for details on clocks.

28.2.2.1 Pre-Divider

The clock source to the PWM Down Counter used to generate a duty cycle and frequency on the PWM_OUTPUT may be pre-divided via bits D6:D3 in the PWMx Configuration Register. This results in a wide range of frequencies for the PWM output. Table 28-2 shows examples of frequencies supported.

TABLE 28-2: EXAMPLE OF PWM FREQUENCIES DERIVED USING PRE-DIVIDER

Example	Clock Select	Clock Pre-Divider	High Count	Low Count	PWM_OUTPUT FREQ	PWM_OUTPUT Duty Cycle
1.	64.52MHz	0	32768	32768	984Hz	50%
2.	64.52MHz	0	192	192	168kHz	50%
3.	64.52MHz	0	382	2	168kHz	99.5%
4.	64.52MHz	0	960	960	33.6kHz	50%
5.	90KHz	0	30000	30000	1.5Hz	50%
6.	90KHz	1	30000	30000	0.75Hz	50%
7.	90KHz	11	30000	30000	0.125Hz	50%

28.2.2.2 Sleep Enable

The Embedded Controller can put each PWM into a sleep state. When a PWM is in the sleep state the internal counters are reset to 0 and the internal state of the PWM and thus the PWM_OUTPUT signal is set to the OFF state.

28.2.3 RESET

This block is reset by nWDT_RST. After the assertion of nWDT_RST, The PWM_OUTPUT is held in the OFF state and the hardware resets the PWM counter registers to their default value.

See Section 5.9, "Registers," on page 43 for details on reset.

28.3 Registers

There are six instances of the PWM Controller block implemented in the SCH5636. Each instance of the PWM Controller is instantiated on the EC AHB and has its Base Address as indicated in Table 28-3, "PWMx Controller Base Address Table":

TABLE 28-3: PWMX CONTROLLER BASE ADDRESS TABLE

PWM Controller Instance	LDN	AHB Base Address
PWM1	16h	F0_5800h
PWM2		F0_5880h
PWM3		F0_5900h
SPEAKER		F0_5980h
PROCHOT		F0_5A00h
PWM4		F0_5A80h

Table 28-4 is a summary of one instance of the PWM controller:

TABLE 28-4: PWMX REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
PWMx Counter ON Time Register	-	00h	32	R/W	
PWMx Counter OFF Time Register	-	04h	32	R/W	
PWMx Configuration Register	-	08h	32	R/W	

TABLE 28-5: PWM CONTROLLER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
PWMx Counter ON Time Register	-	00h	32	R/W	
PWMx Counter OFF Time Register	-	04h	32	R/W	
PWMx Configuration Register	-	08h	32	R/W	

TABLE 28-6: PWMX EC ACCESSIBLE REGISTERS

Offset	Register Name	VTR POR (Suspend)	Full Fan Reset
0h	PWMx Counter ON Time Register	0000h	0000h
4h	PWMx Counter OFF Time Register	FFFFh	0000h
8h	PWMx Configuration Register	0000h	n/a

28.3.1 PWMX COUNTER ON/OFF TIME REGISTERS

TABLE 28-7: PWMX COUNTER ON TIME REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	00h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D30**	**D29**	**…**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	PWMx Counter ON Time[15:0]							

TABLE 28-8: PWMX COUNTER OFF TIME REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	04h			32-bit			**EC SIZE**	
POWER	VTR			FFFFh			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BIT	**D15**	**D30**	**D29**	**…**		**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	PWMx Counter OFF Time[15:0]							

The PWMx Counter ON/OFF Time registers determine both the duty cycle and frequency of the signal generated on PWM_OUTPUT. See FIGURE 28-1: PWM Duty Cycle equation on page 286.

If the PWMx Counter OFF Time[15:0] is set to zero, PWM_OUTPUT is held high (Full On). If the PWMx Counter ON Time is set to zero and the PWMx Counter OFF Time[15:0] is not set to zero, PWM_OUTPUT is held low (Full Off). Note that the default case is full off. Otherwise, both the high and low count registers will contain a value that will be used to determine the length of time PWM_OUTPUT will be held high and low. See Table 28-9, "PWM_OUTPUT State".

TABLE 28-9: PWM_OUTPUT STATE

PWM Count ON Time	PWM Count OFF Time	State Of PWM_OUTPUT
Don't Care	0000h	Full On
0000h	Non-Zero Value	Full Off
Non-Zero Value	Non-Zero Value	Toggling On and Off

The counter values preload a 16-bit down-counter that is clocked by either the high frequency clock source or the low frequency clock source (see bit[1] CLK_Select of PWMx Configuration Register). The firmware will program the on and off count values that correspond to the PWM Current Duty Cycle and PWM Frequency. When PWM_OUTPUT is OFF and the internal counter is zero, the PWMx Counter ON Time is loaded into the counter. The PWM_OUTPUT signal will transition to the ON state and the internal counter will count down to zero at the programmed frequency for the duration of the programmed on time. Similarly, when the PWM_OUTPUT is in the ON state and the internal counter is zero, the PWMx Counter OFF Time is loaded into the counter. The PWM_OUTPUT signal will transition OFF and the internal counter will count down to zero at the programmed frequency for the duration of the programmed off time.

The PWMx Counter ON/OFF Time Registers may be updated at any time. Values written into the two registers are kept in holding registers. The holding registers are transferred into the PWMx Counter ON/OFF Time Registers when all four bytes have been written with new values and the internal counter completes the OFF time count. If the PWM is in the Full On state then the PWMx Counter ON/OFF Time Registers are updated from the holding registers as soon as all four bytes have been written. Once the two registers have been updated the holding registers are marked empty. and all four bytes must again be written before the holding registers will be reloaded into the PWMx Counter ON/OFF Time Registers. Reads of the PWMx Counter ON/OFF Time Registers return the current contents of the registers that are used to load the counter and not the holding registers.

28.3.2 PWMX CONFIGURATION REGISTER

TABLE 28-10: PWMX CONFIGURATION REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	08h			32-bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R							
BIT NAME	Reserved							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved	Clock Pre-Divider				Invert	Clock Select	PWM Enable

PWM Enable

0= disabled (gates clocks to save power)

1= enabled (default)

Note:	When the PWM enable bit is set to 0 the internal counters are reset and the internal state machine is set to the OFF state. In addition, the PWM_OUTPUT signal is set to the inactive state as determined by the Invert bit. The PWMx Counter ON/OFF Time Registers are not affected by the PWM enable bit and may be read and written while the PWM enable bit is 0.

Clock Select

The Clk_Select bit determines the clock source used by the PWM duty cycle and frequency control logic.

0= 64.52MHz MCLK (default)

1= 90KH

Invert

0= PWM_OUTPUT ON State is active high

1= PWM_OUTPUT ON State is active low

Clock Pre-Divider

The Clock source for the 16-bit down counter (see PWMx Counter ON/OFF Time Registers) is determined by bit D1 of this register. The Clock source is then divided by the value of Pre-Divider+1 and the resulting signal determines the rate at which the down counter will be decremented. For example, a Pre-Divider value of 1 divides the input clock by 2 and a value of 2 divides the input clock by 3. A Pre-Divider of 0 will disable the Pre-Divider option.

29.0 PECI INTERFACE

29.1 Overview

The SCH5636 includes a PECI Interface to allow the EC to retrieve temperature readings from PECI-compliant devices. The PECI Interface implements the PHY and Link Layer of a PECI host controller as defined in References[1] and includes hardware support for PECI_REQUEST# functionality and the PECI 2.0 command set.

This chapter focuses on SCH5636 specific PECI Interface configuration information such as For register details see References [1]., Power Domain, Reset, Physical Interface, Interrupts and Clocks. For a functional description of the SCH5636 PECI Interface refer to References [1].

29.2 References

1. PECI Interface Core, Rev. 1.2, Core-Level Architecture Specification, SMSC Confidential, 8/1/08.

29.3 Block Interface Parameters

29.3.1 SIGNAL LIST

TABLE 29-1: PECI INTERFACE SIGNAL LIST

Signal Name	Type	Description
VREF_PECI	INPUT	PECI Voltage Reference pin
PECI_READY	INPUT	PECI Ready input pin (VREF_PECI)
PECI_DAT	INPUT/OUTPUT	PECI Data signal pin (VREF_PECI)
PECI_REQUEST#	OUTPUT	PECI Request output pin (VTR)
EC SPB	I/O Bus	EC SCH5636 peripheral bus
MCLK	INPUT	Master Clock
MCLK_DIV2_EN	INPUT	SCH5636 clock enable signal for PECI baud clock (32 MHz)
SPB_CLK_EN	INPUT	SCH5636 clock enable signal for Host interface clock
nWDT_RST	INPUT	Synchronous block reset signal
PECI_INT	OUTPUT	Interrupt signal from PECI controller to EC
SLEEP_EN	INPUT	External sleep enable control
CLOCK_REQ	OUTPUT	Clock required status
VTR	POWER	Digital logic voltage supply
GND		Ground

29.4 Power, Clocks and Reset

29.4.1 POWER DOMAIN

The PECI Interface core logic is powered by VTR; the Physical Interface Power Domain is VREF_PECI.

29.4.2 CLOCKS

The PECI Interface clock requirement is defined below in Table 29-2:

TABLE 29-2: PECI CLOCKING

Domain	Clock	Type	Frequency
CORE	MCLK	Fixed Clock	64.5 MHz
	MCLK_DIV2_EN	Fixed Clock Enable	32.25 MHz
HOST	MCLK	Fixed Clock	64.5 MHz
	SPB_CLK_EN	Variable Clock Enable	32.25 MHz and slower.

29.4.3 RESET

The PECI Interface is reset on a nWDT_RST. The PECI Interface core also includes soft reset capabilities which reset control logic and part of registers. See References [1] for details.

29.5 Interrupts

The interrupt from the PECI Interface module is routed to the PECI bit of the GIRQ16 Source register.

29.6 Physical Interface

PECI pins for the SCH5636 are listed in Table 3-1, “Signal Descriptions,” on page 11.

Timing for PECI pins is described in Reference 1.

29.7 Instance Description

There is one instance of the PECI Interface block implemented in the SCH5636.

The PECI Interface is instatiated on the EC AHB and has its own Logical Device Number, and Base Address as indicated in Table 29-3.

TABLE 29-3: PECI INTERFACE BASE ADDRESS

Watchdog Timer Instance	LDN	AHB Base Address
PECI	18h	F0_6400h

The Table 29-4 is a register summary for the PECI Interface. Each EC address is indicated as an SPB Offset from its AHB address.

TABLE 29-4: PECI INTERFACE REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Write Data Register	-	00h	32	RW	
Read Data Register	-	04h	32	RW	
Control Register	-	08h	32	RW	
Status Register 1	-	0Ch	32	RWC	
Status Register 2	-	10h	32	RWC	
Error Register	-	14h	32	RWC	
Interrupt Enable 1 Register	-	18h	32	RW	
Interrupt Enable 2 Register	-	1Ch	32	RW	
Optimal Bit Time Register (Low Byte)	-	20h	32	RW	
Optimal Bit Time Register (High Byte)	-	24h	32	RW	
Request Timer Register (Low Byte)	-	28h	32	RW	
Request Timer Register (High Byte)	-	2Ch	32	RW	
Reserved	-	30h-3Ch	32	R	
Block ID Register	-	40h	32	R	
Revision Register	-	44h	32	R	
MCHP Reserved	-	48h - 7Ch	32	RW	Note 29-1

Note 29-1 MCHP Reserved registers are reserved for Microchip use only. Reading and writing MCHP Reserved registers may cause undesirable results.

For register details see References [1].

30.0 PROCHOT# MONITOR

30.1 General Description

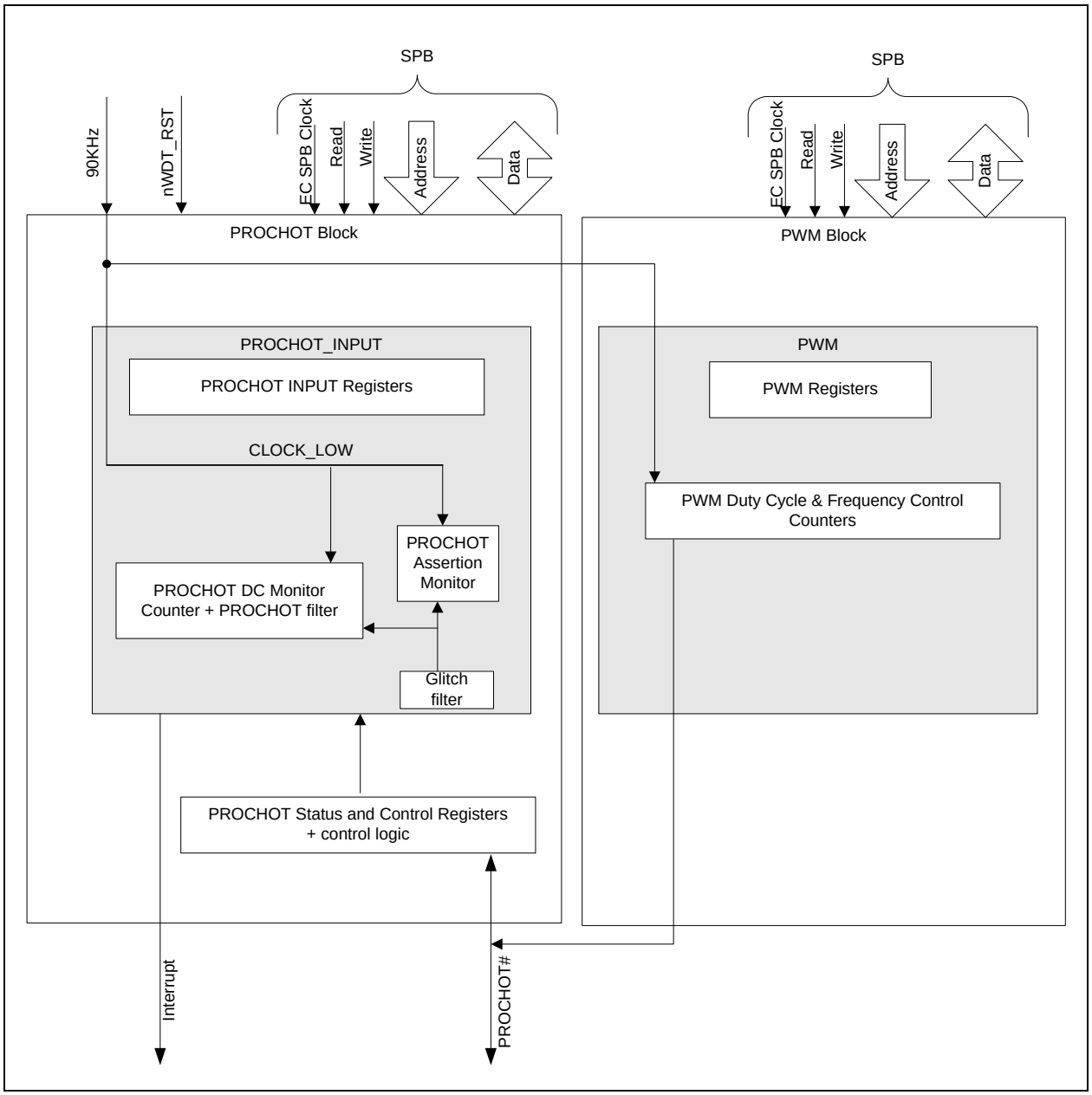
This block monitors the PROCHOT signal from the host processor. Features include:

- monitor for a single assertion
- monitor the cumulative PROCHOT active time

The FORCE_PR output function is implemented with a dedicated PWM unit for the PROCHOT# pin.

30.2 PROCHOT Monitor Block Diagram

FIGURE 30-1: PROCHOT# MONITOR BLOCK DIAGRAM



30.3 Port List

TABLE 30-1: PROCHOT PORT LIST

Signal Name	Direction	Description
nWDT_RST	INPUT	Watchdog Timer Reset
SPB	BUS	EC SPB
90KHZ_En	INPUT	Low Frequency Clock Source
PROCHOT#	INPUT	PROCHOT input signal from PROCHOT_IN# Pin
Interrupt	OUTPUT	Output used to indicate that PROCHOT monitor needs to be serviced.

30.4 Power, Clocks and Reset

30.4.1 POWER DOMAIN

This block is powered by the VTR power supply.

30.4.2 CLOCKS

This block uses the EC AHB clock, as well as the system clock MCLK and the 90KHz clock enable.

30.4.3 RESET

This block is reset when nWDT_RST is asserted.

30.5 Interrupts

The PROCHOT# Monitor can generate an interrupt event to the EC, routed to bit PHOT in the GIRQ16 Source register. The interrupt is asserted whenever either PHOT_Assert or PHOT_Period in the PROCHOT Status/Control Register is 1.

30.6 Operation

This PROCHOT Monitor has been designed to be used in a microcontroller implementation. The registers used to configure and control the PROCHOT logic are accessible via the EC SPB.

This block may be used by the EC to support the following features:

- Monitor and measure a single PROCHOT assertion
- Monitor and measure cumulative PROCHOT assertions

The PROCHOT logic defaults to disabled (i.e., not monitoring). To enable the PROCHOT function set the PROCHOT Enable bit in the PROCHOT Status/Control Register to one.

A FORCE_PROCHOT# is enabled by changing the output mux function for the PROCHOT# pin to the alternate function PWM5. The PROCHOT Monitor will continue to monitor the pin even if the pin mux selects the PWM output function.

30.7 PROCHOT Monitor Functions

30.7.1 MONITORING PROCHOT ASSERTIONS

The PROCHOT Assertion monitor measures a single PROCHOT assertion and, if the PROCHOT# input signal is asserted for a period of time greater than or equal to the programmed limit, a status bit will be set. If enabled, this status bit may generate an interrupt event.

30.7.1.1 PROCHOT Assertion Counter

The PROCHOT Assertion Counter is an internal 16-bit up-counter clocked by the 90KHZ clock. It is reset to 00_0000h on a VCC RESET and when the PROCHOT# input signal is high. When the PROCHOT# input signal is low, the counter is incrementing. If the count value meets or exceeds the limit value programmed, a PROCHOT Assertion status bit will be set immediately (i.e., the count value is evaluated every time the counter is incremented), and the Assertion Counter will be copied into the PROCHOT Assertion Counter Register.

30.7.2 MONITORING PROCHOT CUMULATIVE ACTIVE TIME

The internal PROCHOT Active Counter is a 24-bit counter that is clocked by the 90KHZ clock and gated by the PROCHOT# input signal. When the PROCHOT# input signal is active (low), the counter is incrementing. When the PROCHOT# input signal is inactive (high), the counter is idle. The counter is reset to 00_0000h on a VTR POR, when the PROCHOT Monitor is disabled, and when the PROCHOT Duty Cycle Period counter expires. The value in the PROCHOT Active Counter is readable via the PROCHOT Cumulative Count Register. If the PROCHOT# input signal has been asserted for a period greater than or equal to the PROCHOT filter time, the PROCHOT Cumulative Count Register reflects the current value in the internal Active Counter, otherwise it reads back the last valid count value.

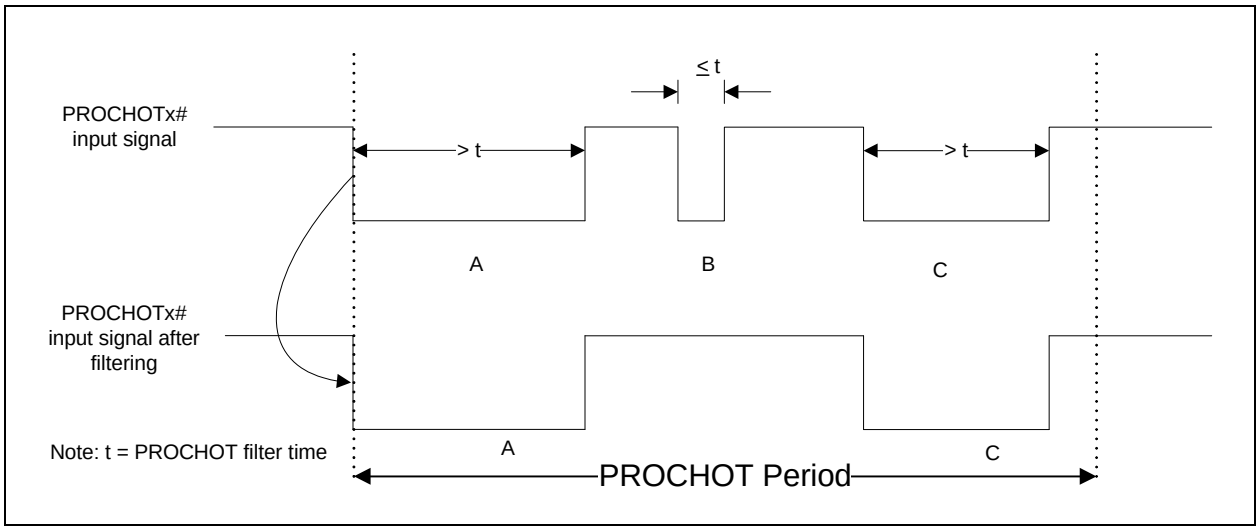
30.7.2.1 PROCHOT Glitch Filter

The incoming PROCHOT# signal can optionally be filtered to eliminate transitions on the signal that last less than ~31ns and potentially up to ~46ns. This filter is enabled by the Filt_Enable bit in the PROCHOT Status/Control Register.

30.7.2.2 PROCHOT Filters

The processor generating the PROCHOT# signal is required to assert that signal low for a period greater than or equal to 500usec. A filter has been implemented via the PROCHOT Active Counter to remove low frequency pulses less than 450usec. This is done by discarding counter incremental values less than or equal to 40 90KHz clock pulses. Figure 30-2, "Effects of PROCHOT Filtering" shows the effect of PROCHOT# input filtering:

FIGURE 30-2: EFFECTS OF PROCHOT FILTERING

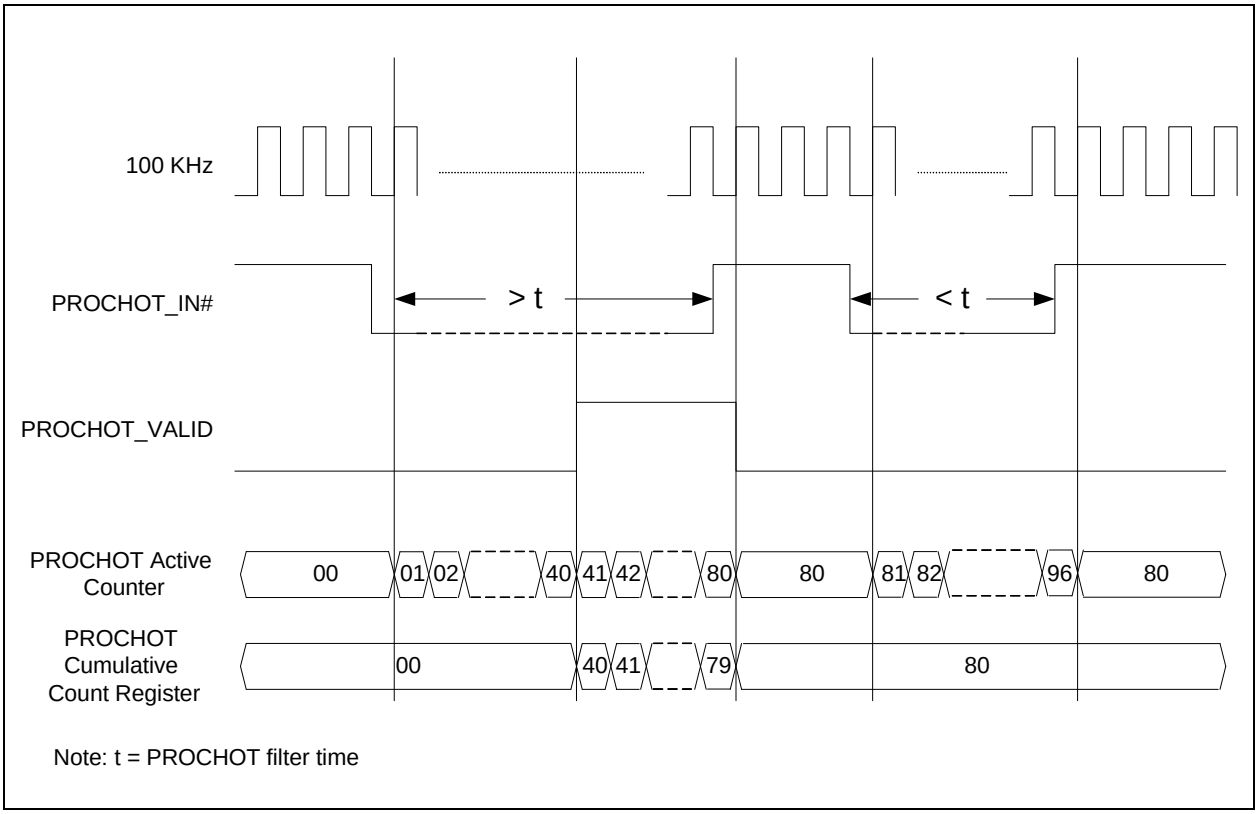


Note 1: The PROCHOT Cumulative Count Register will contain the number of clocks that occurred during low periods A and C. Since time period B is less than the filter time, it will not be counted.

2: The PROCHOT filter removes only low pulses detected on the PROCHOT# input that are less than or equal to the filter time.

Figure 30-3, "Example of PROCHOT Active Counter" provides an example of the interaction between the internal PROCHOT Active Counter and the PROCHOT Cumulative Count Register:

FIGURE 30-3: EXAMPLE OF PROCHOT ACTIVE COUNTER



Note:	When the PROCHOT Active Counter is read during the time PROCHOT_VALID is asserted, its value reflects the current PROCHOT Active Counter value. A read at any other time will return the last known valid count value.

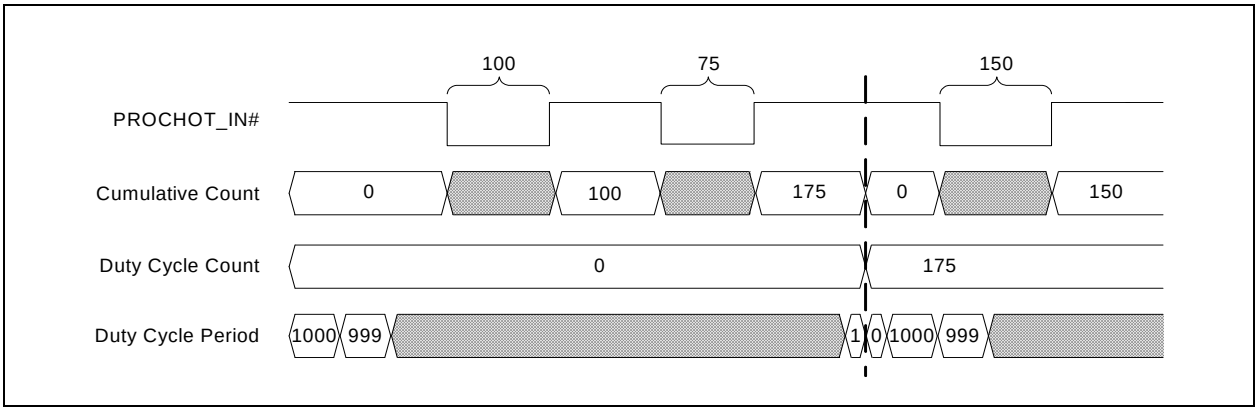
30.7.2.3 PROCHOT Duty Cycle

The PROCHOT duty cycle (the percentage of time that PROCHOT was asserted for a period longer than the filter time) is derived from the PROCHOT Duty Cycle Count Register and the PROCHOT Duty Cycle Period Register. The PROCHOT Duty Cycle Period Register enables a repeating interval up to 186 seconds. It provides a reload value for an internal Duty Cycle Counter that is clocked on the 90KHz clock. When the period expires, the current value of the PROCHOT Cumulative Count Register is copied into the PROCHOT Duty Cycle Count Register and an interrupt is optionally signaled. The internal PROCHOT Active Counter and the internal Duty Cycle Counter are reset, along with the PROCHOT Cumulative Count Register. The duty cycle can be calculated using the relation:

$$DutyCycle = \frac{PROCHOT\ Duty\ Cycle\ Count}{PROCHOT\ Duty\ Cycle\ Period}$$

Figure 30-4, "PROCHOT Duty Cycle Example" shows an example of how the Cumulative Count, Duty Cycle Count and Duty Cycle Period registers relate. The numbers are arbitrary and are for illustration purposes only:

FIGURE 30-4: PROCHOT DUTY CYCLE EXAMPLE



30.8 FORCE_PROCHOT Function

Each PROCHOT pin can be used either as an input-only pin, in which only the PROCHOT monitor function operates, or as a bi-directional pin, where the SCH5636 can also be used to generate a PROCHOT event to the system CPU by generating a PWM output at a particular duty cycle and frequency.

To configure a PROCHOT# pin for bi-directional use, use the GPIO pin multiplexing control for the PROCHOT# pin. Configure the pin as an open-drain output, and set the pin multiplexing function for PWM. Set the Invert bit of the associated PWM to ‘1b’. Configure the associated PWM for the desired duty cycle and frequency. By adjusting the PWM On time and PWM Off time, the PROCHOT# output can be set to full on, full off, or toggling.

The PROCHOT# monitor function will continue to operate when the pin mux is set for PWM operation.

30.9 Registers

The PROCHOT# Monitor block is instantiated on the EC AHB and has its own Logical Device Number, and Base Address as indicated in Table 30-2.

TABLE 30-2: PROCHOT# MONITOR BASE ADDRESS

PROCHOT# Monitor **Instance**	**LDN from (Table 4-3 on page 27)**	**AHB Base Address**
PROCHOT	1Bh	F0_6C00h

Table 30-3 is a register summary for the PROCHOT# Monitor.

TABLE 30-3: PROCHOT# MONITOR REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
PROCHOT Cumulative Count Register	-	00h	32	R	
PROCHOT Duty Cycle Count Register	-	04h	32	R	
PROCHOT Duty Cycle Period Register	-	08h	32	R/W	
PROCHOT Status/Control Register	-	0Ch	32	R/W	
PROCHOT Assertion Counter Register	-	10h	32	R	
PROCHOT Assertion Counter Limit Register	-	14h	32	R/W	

30.9.1 PROCHOT CUMULATIVE COUNT REGISTER

Whenever byte 0 of this register is read, bytes 1, 2 and 3 are held in a holding register. Subsequent reads of 8-bits or 16-bits to bytes other than byte 0 return data from the holding register, although the actual register may continue to be updated by hardware. A 32-bit read always returns the current register value.

TABLE 30-4: PROCHOT CUMULATIVE COUNT REGISTER

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	00h				32-Bit		**EC SIZE**	
POWER	VTR				0000_0000h		nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Cumulative_PROCHOT_Active[23:16]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Cumulative_PROCHOT_Active[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Cumulative_PROCHOT_Active[7:0]							

Cumulative_PROCHOT_Active

This register contains the current filtered PROCHOT cumulative count value. This register returns the value of the internal PROCHOT Active Counter as long as the filtered value of PROCHOT# is low (active). When PROCHOT# transitions from low to high (from active to inactive) this register retains its most recent value. This register, as well as the internal PROCHOT Active counter, are cleared to 0 when this register is copied into the PROCHOT Duty Cycle Count Register on the 1 to 0 transition of the internal Duty Cycle Counter.

30.9.2 PROCHOT DUTY CYCLE COUNT REGISTER

Whenever byte 0 of this register is read, bytes 1, 2 and 3 are held in a holding register. Subsequent reads of 8-bits or 16-bits to bytes other than byte 0 return data from the holding register, although the actual register may continue to be updated by hardware. A 32-bit read always returns the current register value.

TABLE 30-5: PROCHOT DUTY CYCLE COUNT REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	04h			32-Bit			**EC SIZE**	
POWER	VTR			0000_0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Duty_Cycle_Count[23:16]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Duty_Cycle_Count[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Duty_Cycle_Count[7:0]							

Duty_Cycle_Count

The contents of the PROCHOT Cumulative Count Register is copied into this register when the PROCHOT Duty Cycle Period Register transitions from 1 to 0.

30.9.3 PROCHOT DUTY CYCLE PERIOD REGISTER

TABLE 30-6: PROCHOT DUTY CYCLE PERIOD REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	08h			32-Bit			**EC SIZE**	
POWER	VTR			0000_0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D31**	**D30**	**D29**	**D28**	**D27**	**D26**	**D25**	**D24**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Reserved							
BYTE2 BIT	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Duty_Cycle_Period[23:16]							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Duty_Cycle_Period[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Duty Cycle_Period[7:0]							

Duty_Cycle_Period

This register defines the number of 11.1µs periods required for a duty cycle measurement. It can be programmed for periods from 11µs to 186s. As long as the PROCHOT device is enabled, an internal counter repeatedly counts down from this value to 0. When the counter transitions from 1 to 0, the contents of the PROCHOT Cumulative Count Register is copied into the PROCHOT Duty Cycle Count Register, the status bit in the register is set and the counter is reloaded from this register.

Setting this register to 0 disables duty cycle measurement.

When this register is written, both the internal PROCHOT Active Counter and the PROCHOT Cumulative Count Register are reset to 0.

30.9.4 PROCHOT STATUS/CONTROL REGISTER

TABLE 30-7: PROCHOT STATUS/CONTROL REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	0Ch			32-Bit			**EC SIZE**	
POWER	VTR			0000h if PHOT_Pin is 0 0002h if PHOT_Pin is 1			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R/WC	R/WC	R	R
BIT NAME	Reserved				PHOT_ Period	PHOT_As sert	Reserved	
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R/W	R/W	R/W	R/W	R	R/W
BIT NAME	Reserved		Filt_ Enable	PHOT_ Reset	Period_ Enable	Assert_ Enable	PHOT_ Pin	PHOT_ Enable

PHOT_Enable

This bit enables the PROCHOT Monitor logic. When Enable is 0, no status bits in this register or any of the counters in this block will be updated, although the registers can still be read by the EC.

0=PROCHOT Idle (default). This mode gates the clocks to the PROCHOT I/O block Contents of the registers are not affected.

1=PROCHOT Monitoring

PHOT_Pin

This bit reflects the state of the PROCHOT# Pin input.

0=PROCHOT Pin is low

1=PROCHOT Pin is high

Note: When PHOT_Enable is 0, this bit is not updated and does not reflect the state of the PROCHOT# pin input.

Assert_Enable

This bit determines whether or not an interrupt will be generated when a PROCHOT Assertion Event occurs, as indicated by the PHOT_Assert bit.

0=PROCHOT Assertion Event interrupt blocked

1=PROCHOT Assertion Event interrupt enabled

Period_Enable

This bit determines whether or not an interrupt will be generated when a PROCHOT Duty Cycle Period Event occurs, as indicated by the PHOT_Period bit.

0=PROCHOT Duty Cycle Period Event interrupt blocked

1=PROCHOT Duty Cycle Period Event interrupt enabled

PHOT_Reset

When this bit is set to 1, all the registers in the PROCHOT logic, including the PROCHOT Status/Control Register, are set to the initial default value. This occurs independently of the value of When the PROCHOT Status/Control Register is set to its default value, this bit is cleared to 0.

When this bit is 0, it has no effect.

PHOT_Assert

This bit is set when the Assertion Counter value is greater than or equal to the Assertion Counter Limit value. It is cleared when written with a '1b' if the counter value is no longer violating the limit. Writes of '0b' or when the counter is violating the limit have no affect.

PHOT_Period

This bit is set to '1b' when the PROCHOT Duty Cycle Count Register transitions from '1b' to '0b'. It is cleared when written with a '1b'. Writes of '0b' have no affect.

Filt_Enable

This bit determines whether a digital filter eliminates pulses less than ~31ns and potentially up to ~46ns on the PROCHOT# signal before PROCHOT# is sampled by the Assertion counter or the Active counter.

0=PROCHOT# input not filtered

1=PROCHOT# input filtered

30.9.5 PROCHOT ASSERTION COUNTER REGISTER

Whenever byte 0 of this register is read, bytes 1, 2 and 3 are held in a holding register. Subsequent reads of 8-bits or 16-bits to bytes other than byte 0 return data from the holding register, although the actual register may continue to be updated by hardware. A 32-bit read always returns the current register value.

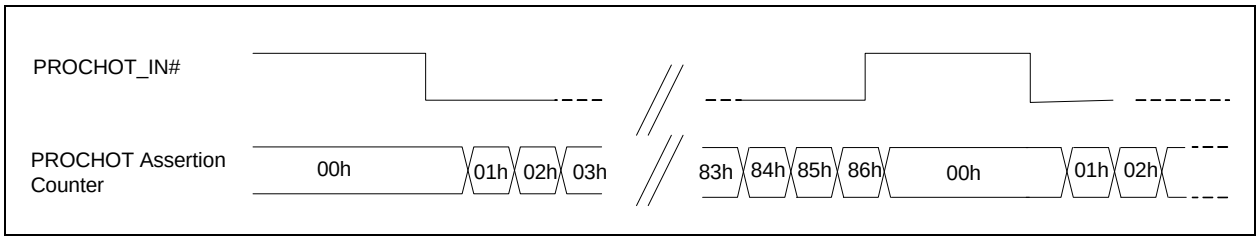
TABLE 30-8: PROCHOT ASSERTION COUNTER REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	10h			32-Bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Assertion_Counter[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R	R	R	R	R	R	R	R
BIT NAME	Assertion_Counter[7:0]							

Assertion_Counter

The PROCHOT Assertion Counter is a 16-bit up-counter that is clocked by the 90KHz clock and is gated and reset by the PROCHOT# input signal. If enabled, this counter increments when the PROCHOT# input signal is active (low) and is reset to 0000h when the pin is inactive (high). This counter is used to measure a single PROCHOT assertion. The following example shows how the counter is reset on the falling edge of the PROCHOT# input signal. This register allows the firmware to read the current count value.

FIGURE 30-5: PROCHOT ASSERTION COUNTER EXAMPLE



This counter is a saturating counter: when it reaches FFFFh, it stops counting, rather than rolling over to 0000h.

30.9.6 PROCHOT ASSERTION COUNTER LIMIT REGISTER

TABLE 30-9: PROCHOT ASSERTION COUNTER LIMIT REGISTER

HOST OFFSET	N/A			N/A			**HOST SIZE**	
EC OFFSET	14h			32-Bit			**EC SIZE**	
POWER	VTR			0000h			nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE1 BIT	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Assertion_Count_Limit[15:8]							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Assertion_Count_Limit[7:0]							

Assertion_Count_Limit

The PROCHOT Assertion Counter Limit register is compared to the 16-bit PROCHOT Assertion Counter. If the value in the PROCHOT Assertion counter is greater than or equal to the value in the limit register, then the PHOT_Assert bit contained in the PROCHOT Status/Control Register is set. In addition, an interrupt will be generated if the Assert_Enable bit in the PROCHOT Status/Control Register is set.

Note: A value of 0000h disables the comparison process.

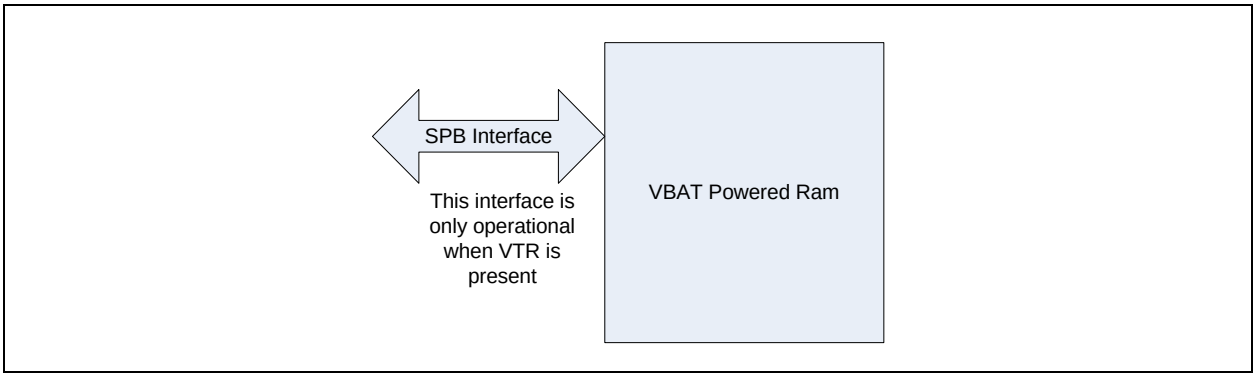
31.0 VBAT POWERED RAM

31.1 General Description

The VBAT Powered RAM provides a 64 Byte Random Accessed Memory that is operational while VTR is powered, and will retain its values while VTR is unpowered. The RAM is organized as a 16 x 32-bit memory, for a total of 64 bytes.

31.2 Block Diagram

FIGURE 31-1: BLOCK DIAGRAM OF VBAT POWERED RAM



31.3 Power, Clocks and Reset

31.3.1 POWER DOMAIN

This block is in the VTR power domain for EC interaction and uses the VBAT power domain for memory retention

See Section 5.9, "Registers," on page 43 for details on power domains.

31.3.2 CLOCKS

The VBAT Powered RAM has one clock input., the EC Bus Clock.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

31.3.3 POWER ON RESET

The VBAT Powered RAM is reset on a VBAT_POR.

See Section 5.9, "Registers," on page 43 for details on reset.

31.4 Interrupts

The VBAT Powered RAM has no interrupts

31.5 Registers

The VBAT Powered RAM is instantiated on the EC AHB and has its own Logical Device Number and Base Address as indicated in Table 31-1. See Note 1: on page 26 for information on how VBAT-powered logical devices are organized on the AHB bus.

TABLE 31-1: VBAT POWERED RAM BASE ADDRESS TABLE

VBAT Powered RAM **Blocks**	**LDN from (**Table 4-3 on page 27**)**	**AHB Base Address**
VBAT Backed Memory	33h	F0_CD00h

Each 32-bit RAM location is an SPB Offset from the AHB base address.

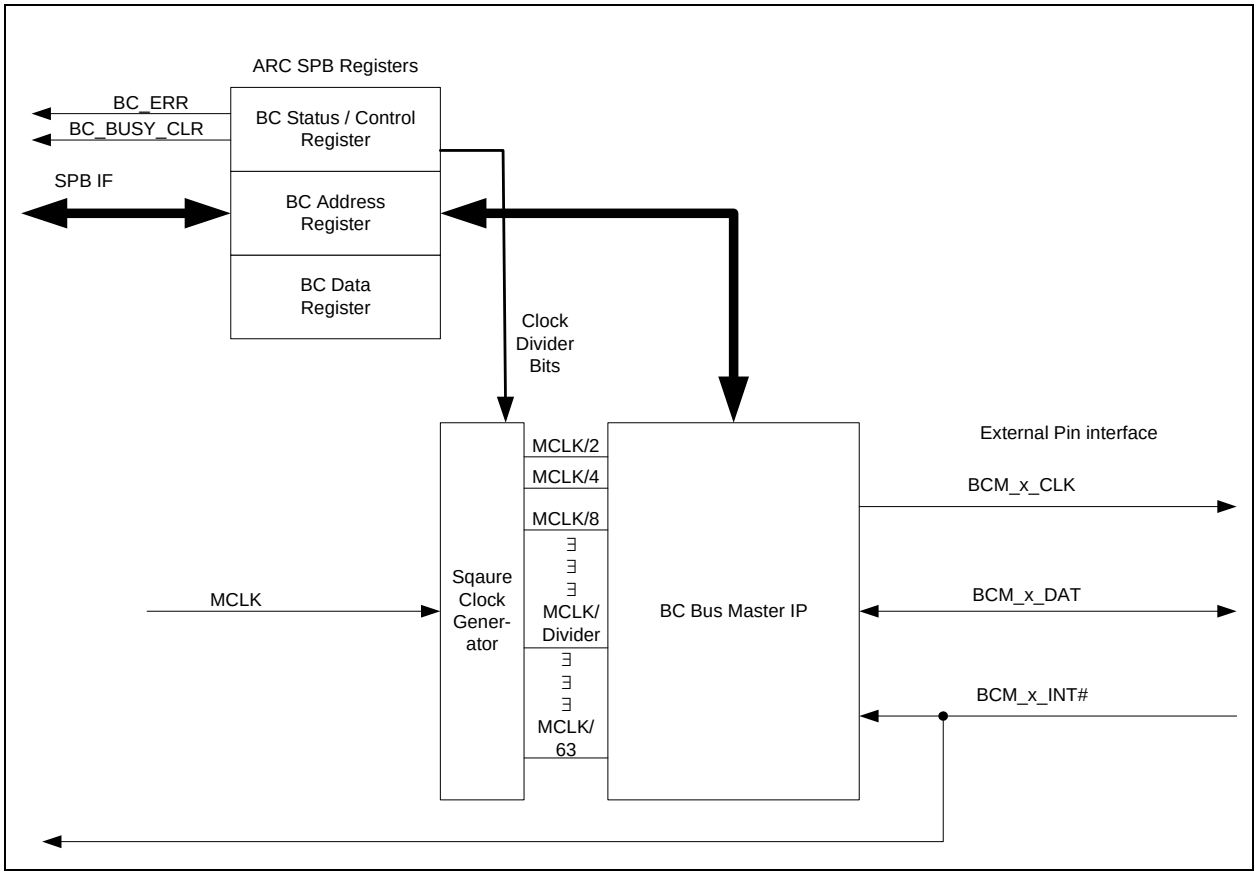
32.0 MICROCHIP BC-LINK™ MASTER

32.1 General Description

The function of this block is to.provide Microchip's BC-Link™ to a slave device. The Microchip BC-Link protocol includes a start bit to signal the beginning of a message, and a turnaround (TAR) period for bus transfer between the Master and Companion devices.

32.2 Block Diagram

FIGURE 32-1: BC-LINK™ MASTER BLOCK DIAGRAM



32.3 Signal List

TABLE 32-1: BC-LINK™ SIGNAL LIST

Signal Name	Direction	Description
BCM_x_CLK	OUTPUT	64MHz - 252KHz output clock, where x is A, B,C or D.
BCM_x_DAT	INPUT/OUTPUT	Bidirectional data line, where x is A, B,C or D.
BMM_x_INT#	INPUT	Input from the companion device, where x is A, B,C or D.
BC_ERR	OUTPUT	BC-Link master error interrupt
BC_BUSY_CLR	OUTPUT	BC-Link master Busy Clear interrupt
MCLK	INPUT	64 MHz Master clock
SPBIF	I/O Bus	Bus used for register access

32.3.1 BC-LINK™ PINS

The BC-Link master in the SCH5636 uses 8 ma buffers and consists of signals BC_INT#, BC_DAT and BC_CLK.

The BC-Link Master maximum clock frequency is 3Mhz. The Clock frequency is set with the BBC Clock Select Register.

The BC_DAT signal requires a weak pull up (100K).

32.4 Power, Clocks and Reset

32.4.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

32.4.2 CLOCKS

This block uses the EC Bus Clock and the 64.52MHz MCLK. The EC Bus Clock is used to access the Registers described in this block. MCLK is divided down to generate the external bus clock.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

32.4.3 RESET

This block is reset on a nWDT_RST. On reset, the BC-Link state machine transitions to the Idle state and waits for the address and data registers to be written.

See Section 5.9, "Registers," on page 43 for details on reset.

32.5 Interrupts

Each Microchip BC-Link™ Master instance has three interrupt events: BC_BUSY_CLR, BC_ERR, and BC_INT#. The Microchip BC-Link™ Master BC_BUSY_CLR and BC_ERR interrupt are generated by changes to the BC-Link™ Status Register. The BC_INT# is an active low level interrupt generated by input pin signal function. The edge detection of the interrupt and wake events are controlled by their associated pin control registers in the Section 22.0, "GPIO Interface," on page 240.

The Microchip BC-Link™ Master block Instance is routed to bits BCM_BUSY_CLR, BCM_ERR and BCM_INT# in the GIRQ18 Source register.

32.6 Operation

Descriptions of the BC-Link read and write operations follows:

32.6.1 READ

The BC-Link Read protocol requires two reads of the BC-Link™ Data Register. The two reads drive a two state-state machine: the two states are Read#1 and Read#2. In both states, a read of the BC-Link™ Data Register transfers data to the EC. In state Read#1 only, a read of the BC-Link™ Data Register also causes a BC-Link transaction to start on the BC-Link pins. In the Read#1 state, a read of the BC-Link™ Data Register starts the read protocol on the BC-Link pins, sets the Busy bit in the BC-Link™ Status Register and transfers to state Read#2. The contents of the data read during Read#1 by the EC is stale and is not to be used. After the Busy bit in the BC-Link™ Status Register autonomously clears to 0, a read of the BC-Link™ Data Register transfers the data read from the peripheral/BC-Link companion chip to the EC and transfers the state machine to state Read#1.

1. Software starts by checking the status of the BUSY bit in the BC-Link™ Status Register. If the Busy bit is 0, proceed; if Busy is 1, wait.
2. Software writes the address of the register to be read into the BC-Link™ Address Register.
3. Software then reads the BC-Link™ Data Register. This read returns random data. The read activates the Microchip BC-Link™ Master to transmit the read request packet to the BC-Link companion. When the transfer initiates, the hardware sets the BUSY bit to a 1.
4. The BC-Link companion reads the selected register and transmits the read response packet to the Microchip BC-Link™ Master.

Note:	The companion will ignore the read request if there is a CRC error; this will cause the base to time-out and issue a BC_ERR Interrupt.

5. The Microchip BC-Link™ Master loads the BC-Link™ Data Register, clears the BUSY bit to 0 and asserts the BC_BUSY_CLR interrupt.
6. In response to the BC_BUSY_CLR interrupt, or after a poll of the BUSY bit returns 0, software checks the BC_ERR bit in the BC-Link™ Status Register.

a) If there was no BC Link error, software reads the BC-Link™ Data Register, which contains valid data.

b) If a Bus Error occurred, software issues a soft reset by setting the Reset bit in the BC-Link™ Status Register.

7. The read can re-tried once BUSY is cleared.

Software ***must*** read the BC-Link™ Data Register exactly twice for each BC-Link READ transaction (the reads in step 3 and step 6a). If not, the BC-Link interface could be presenting incorrect data when software thinks it is accessing a valid register.

32.6.2 WRITE

1. Software starts by checking the status of the BUSY bit in the BC-Link™ Status Register. If the BUSY bit is 0, proceed; if BUSY is 1, wait.
2. Software writes the address in the companion of the register to be written into the BC-Link™ Address Register.
3. Software writes the data to be written into the addressed companion register in to the BC-Link™ Data Register. The write starts the BC_Link write operation. The Microchip BC-Link™ Master sets the BUSY bit in the BC-Link™ Status Register.
4. The Microchip BC-Link™ Master Interface transmits the write request packet.
5. When the write request packet is received by the BC-Link companion, the CRC is checked and data is written to the addressed companion register.
6. The companion sends an ACK if the write is completed.

Note:	A time-out will occur approximately 16 BC-Link clocks after the packet is sent by the Microchip BC-Link™ Master. Approximately 48 clocks later, the Microchip BC-Link™ Master will set the BC_ERR bit in the BC-Link™ Status Register to 1 and clear the BUSY bit to 0.

7. The Microchip BC-Link™ Master issues the BC_BUSY_CLR interrupt and clears the BUSY bit after receiving the ACK from the companion
8. In response to the BC_BUSY_CLR interrupt, or after a poll of the BUSY bit returns 0, software checks the BC_ERR bit in the BC-Link™ Status Register.

a) If there was no BC Bus error, the transaction is complete.

b) If a Bus Error occurred, software issues a soft reset by setting the Reset bit in the BC-Link™ Status Register.

9. The write can re-tried once BUSY is cleared.

32.7 Registers

The Microchip BC-Link™ Master is instantiated on the EC AHB and has its own Logical Device Number and Base Address as indicated in Table 32-2:

TABLE 32-2: BC-LINK™ MASTER BASE ADDRESS TABLE

BC-Link InstanceS	LDN	AHB Base Address
BC-Link	5h	F0_1400h

Table 32-3 is a register summary for the Microchip BC-Link™ Master.

TABLE 32-3: BC-LINK™ MASTER REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
BC-Link™ Status Register	-	00h	32	R/W	
BC-Link™ Address Register	-	04h	32	R/W	
BC-Link™ Data Register	-	08h	32	R/W	
BBC Clock Select Register	-	0Ch	32	R/W	

32.7.1 BC-LINK™ STATUS

TABLE 32-4: BC-LINK™ STATUS REGISTER

HOST ADDRESS	N/A						**HOST SIZE**	
EC OFFSET	00h					32-bit	**EC SIZE**	
POWER	VCC					81h	nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-‘	-	-	-	-
EC TYPE	R/W	R/WC	R/W	R/W	R	R	R	R/W
BIT NAME	RESET	BC_ ERR	BC_ ERR_ INT_EN	BC_ Busy_ CLR_ INT_EN	Reserved			BUSY

BUSY

This bit is ‘1’ when the BC interface is transferring data and while the BC-Link Master controller is in reset. It is ‘0’ otherwise. When BUSY is cleared by hardware, a positive edge interrupt is generated.

BC_Busy_CLR_INT_EN

Enable signal for the BC-Link BC_BUSY_CLR interrupt. When this bit is ‘1’, the interrupt signal is enabled. When this bit is ‘0’, the interrupt is disabled. When enabled, this interrupt occurs after a BC Bus read or write. Figure 32-2 show when the interrupt is generated when enabled.

BC_ERR_INT_EN

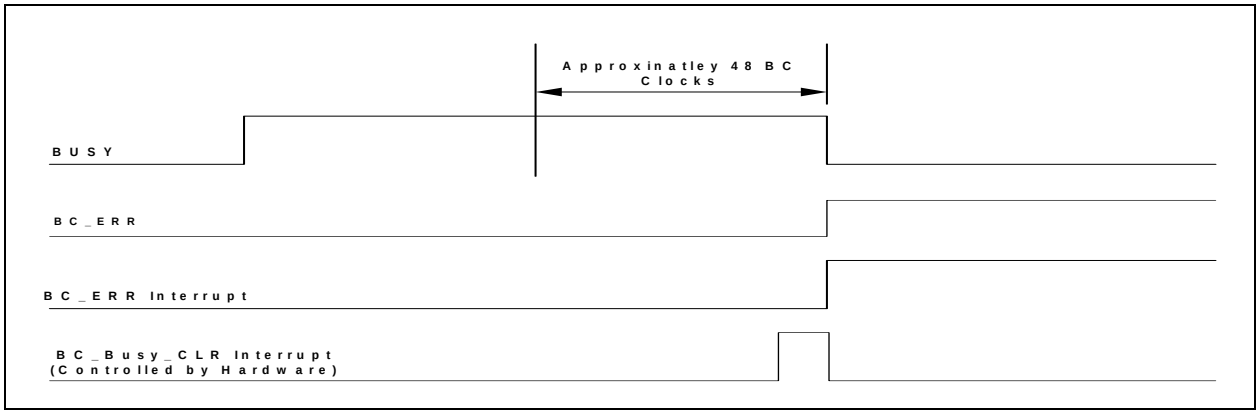
Enable signal for the BC-Link BC_ERR interrupt. When asserted 1, the interrupt signal is enabled. When 0 the interrupt is disabled. Figure 32-2 show when the interrupt is generated if enabled.

BC_ERR

This bit indicates that a Bus Error has occurred. This bit generates the BC_ERR interrupt if enabled by bit BC_BUSY-_CLKR_INT_EN. Software clears this bit when the interrupt is processed. This bit is cleared when written with a ‘1’. Writes of ‘0’ to this bit have no effect.

This bit is set to ‘1’ if bad data is received by the BC-Link Master controller (a CRC Error) or if a time-out occurs due to a non-responsive Companion device. All Companion errors cause the Companion to abort the operation and cause the BC-Link Master controller to time out. Figure 32-2 shows when the interrupt is generated if enabled.

FIGURE 32-2: BC-LINK™ INTERRUPT TIMING



Note:	When the BC_ERR bit is set, the BC-Link Master controller will continue to clock the interface with the data line high to assure the Companion interface returns to idle.

Reset

When this bit is set to ‘1’, the BC-Link Master controller will be reset and held in reset until this bit is de-asserted by hardware. Reset causes the BUSY bit to be set. Once set, this bit will not be cleared until the reset operation of the BC Interface is completed after approximately 48 BC clocks.

PROGRAMMER’S NOTE: The de-assertion of the BUSY bit on reset will not generate an interrupt, even if the BC_Busy_CLR_INT_EN bit is set to ‘1’. The BUSY bit must be polled.

32.7.2 BC-LINK™ ADDRESS

TABLE 32-5: BC-LINK™ ADDRESS REGISTER

HOST ADDRESS	N/A						**HOST SIZE**	
EC OFFSET	04h						32-bit	**EC SIZE**
POWER	VCC						00h	nWDT_RST **DEFAULT**
BUS	EC SPB							
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Address[7:0]							

32.7.3 BC-LINK™ DATA

TABLE 32-6: BC-LINK™ DATA REGISTER

HOST ADDRESS	N/A						**HOST SIZE**	
EC OFFSET	08h						32-bit	**EC SIZE**
POWER	VCC						00h	nWDT_RST **DEFAULT**
BUS	EC SPB							
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data[7:0]							

32.7.4 BC CLOCK SELECT

TABLE 32-7: BBC CLOCK SELECT REGISTER

HOST ADDRESS	N/A						**HOST SIZE**	
EC OFFSET	0Ch						32-bit	**EC SIZE**
POWER	VCC						0004h	nWDT_RST **DEFAULT**
BUS	EC SPB							
BYTE3 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	-	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Divider[7:0]							

Divider

The BC Clock is set tor MCLK / (N+1), where N is 0 to 255. The clock divider bits can only can be changed when the BC Bus is in soft RESET (when either the Reset bit is set by software or when the BUSY Bit is set by the interface).

Example settings are shown in Table 32-8, "Example Frequency Settings":

TABLE 32-8: EXAMPLE FREQUENCY SETTINGS

Divider	Frequency
0	64.52MHz
1	32.25MHz
21	2.99MHz
42	1.50 MHz
63	1MHz

Note:	The Divider register should not be set to a value less than 20, in order not to exceed the maximum BC-Link bus clock frequency.

33.0 JTAG AND XNOR

33.1 General Description

This sections describes functions for debug and test. The SCH5636 includes a. JTAG port for testing and debugging. The XNOR Chain for board test is also included in this section.

33.2 JTAG Slave for debugging ARC firmware

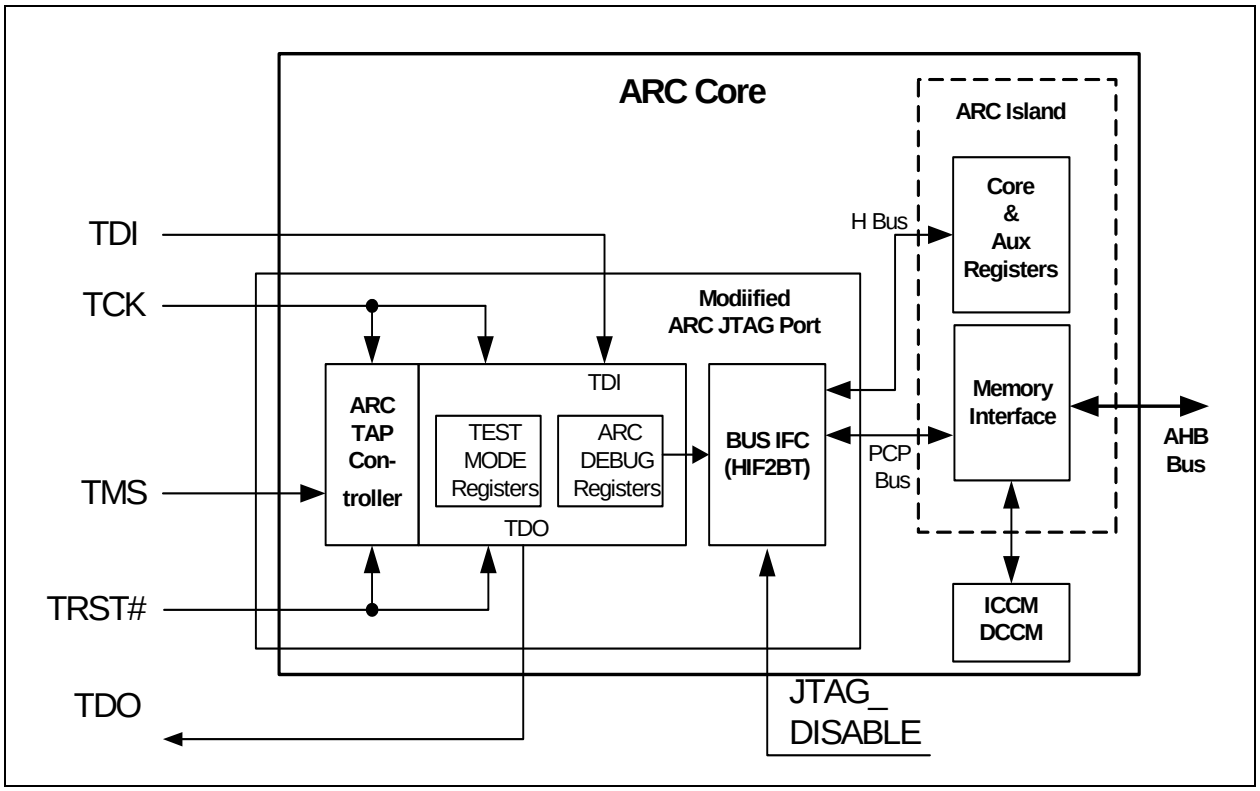
The ARC JTAG Port is defined in the ARC 600 External Interfaces Reference Manual, Chapter 2. The Microchip modifications are described in this chapter. The ARC JTAG Port has been modified by Microchip to provide additional Data Registers (see Note:.) The TEST MODE Register provide additional on-chip support specific to the SCH5636.

All of these function share the SCH5636 JTAG Interface as shown in Table 3-1, “Signal Descriptions,” on page 11.

33.2.1 ARC JTAG CAPABILITIES

- Fully compliant with IEEE1149.1 standard
- 4-bit Instruction Register
- Standard 1-bit BYPASS register
- Standard 32-bit IDCODE register
- Four JTAG registers give access to on-chip memory and register resources
- Can read or write a 32-bit quantity from or to any ARC Core Register, Aux Register or 32-bit aligned memory location. No other interfaces are provided or needed
- Accesses to Aux Registers and Memory do not require the ARC processor to be halted
- Memory accesses are always performed in units of 32 bits

FIGURE 33-1: BLOCK DIAGRAM OF ARC JTAG SYSTEM



As shown in Figure 33-1, the ARC JTAG Port sits within the ARC Core, between the JTAG signals and the ARC Island block, which contains the Core and Aux register sets and the Memory Interface to the Closely-Coupled Memories (ICCM / DCCM) and the AHB Bus. It is not associated with any form of Boundary Scan, but instead is used by an external JTAG host to access memory and register resources on behalf of a debugger program. There are no other connections between the ARC JTAG Port and the rest of the ARC Core: register access is enough to halt and restart the processor, as well as to detect that the processor has halted (a continuous poll on the STATUS32 Aux Register).

Internally, the ARC JTAG port consists of the following sub-blocks:

- The ARC Test Access Port (TAP) Controller. This block is driven by the JTAG_CLK, TMS and TRST# inputs, and provides the control signals for the JTAG data transfers. It is a single state machine consisting of 16 states, whose transitions are controlled strictly by the state of TMS on each rising edge of JTAG_CLK. The low-active TRST# input provides an asynchronous reset, though JTAG_CLK and TMS together can also bring the TAP Controller to the reset state (5 consecutive JTAG_CLK rising edges with TMS=1).
- The ARC DEBUG & TEST MODE Registers blocks. This blocks handles the data transfers as directed by the TAP, and contains the registers and shift registers internal to the JTAG Port. The holding registers in this block consist of a 4-bit Instruction register, a 1-bit Bypass register, and a set of Data registers (see Note:) of various lengths. There are three sets of Data Registers: JTAG Standard Data Registers, JTAG Debug Data Registers, & JTAG Test Mode Data Registers. The JTAG Test Mode Data Registers provide additional on-chip support specific to the SCH5636.
- The Bus Interface block. This block accepts values for the JTAG Debug Data Registers: ADDRESS, DATA, STATUS and TRANSACTION COMMAND, and uses them to request data transfers from the ARC's inner core ("Island") sub-block. \

33.3 JTAG Port Signal Interface Description

The signal pins are defined in Table 3-1, "Signal Descriptions," on page 11.

The TCK input is the clock that drives the JTAG interface. It is asynchronous to other clocks on-chip.

The TMS input is sampled on each rising edge of JTAG_CLK, and governs the transitions among the 16 states of the state machine (TAP) that controls the transfer of data.

The TDI input is the serial data input, shifted in during the Shift-IR and Shift-DR states of the TAP. It is sampled on rising edges of JTAG_CLK.

The TDO output is the serial data output. It is presented on falling edges of JTAG_CLK, 1/2 clock before each input shift, to provide setup and hold time to the next JTAG controller in the chain. The final TDO output pin, after all on-chip chaining (Figure 33-1) is held in high-impedance mode (floating) except when valid data is being presented. The enabled/disabled state of the pin is also changed on falling edges of JTAG_CLK.

The TRST# input is provides the Async JTAG RESET. Note that the reset state of the JTAG port is only local to the JTAG port: its effect is to keep the JTAG port in an idle state and to disengage it from the rest of the system, so that it does not affect other on-chip logic in this state.

33.4 Power, Clocks and Reset

See Section 36.15, "JTAG Interface Timing," on page 357 power on sequence and reset timing.

33.4.1 POWER DOMAINS

The JTAG block is powered by VTR.

33.4.2 CLOCKS

The JTAG port runs internally from the externally-provided JTAG_CLK clock pulses only. There is no requirement for JTAG_CLK to be constantly running.

The following JTAG Registers interface to the ARC Island block (as illustrated in Figure 33-1) for access to registers and memory: STATUS Register (8h), TRANSACTION COMMAND Register (9h), & ADDRESS Register (Ah), DATA Register (Bh). There is a clock relationship required between JTAG_CLK and the ARC Core clock frequency. JTAG_CLK may be asynchronous, but it must be slower than 1/2 the frequency of the ARC Core clock. In practical terms, then, JTAG_CLK should be selected to be nominally 1/4 of the minimum Core clock frequency. See APPLICATION NOTE: on page 45.

APPLICATION NOTE: The Ashling JTAG interface box is not documented to operate any slower than JTAG_CLK = 1MHz, therefore the Core must be running at 4MHz in order to run the ARC debugger through the JTAG interface using the Ashling interface.

Stopping the Core clock disables the JTAG port for debugging purposes. It does not affect IDCODE or BYPASS operation.

See Section 36.15, "JTAG Interface Timing," on page 357 for the maximum frequency f_{clk} on the JTAG_CLK pin to access a JTAG Registers other than STATUS Register (8h), TRANSACTION COMMAND Register (9h), & ADDRESS Register (Ah), DATA Register (Bh).

33.4.3 RESET

The ARC JTAG block has two resets: Async JTAG RESET by its TRST# input and Sync JTAG RESET by JTAG protocol.

33.4.3.1 Async JTAG RESET

The TRST# pin provides the Async JTAG RESET to the JTAG Registers. The TRST# pin has an active low, asynchronous assertion and a synchronous de-assertion. The JTAG Registers will be reset asynchronously (and immediately) upon the active low TRST# assertion. Once the TRST# pin has been de-asserted, a delay of three JTAG_CLK's is required in order to access the JTAG Registers. The JTAG Registers will remain in reset until the three clocks complete the synchronous TRST# pin de-assertion. See Section 36.15, "JTAG Interface Timing," on page 357.

APPLICATION NOTE: After asserting and de-asserted the TRST# pin, a Sync JTAG RESET can be applied before starting to access the JTAG Registers (to meet the TRST# synchronous de-assertion requirement.

JTAG registers, in particular the JTAG Test Mode Data Registers, are set to their initial values by the assertion of the TRST# pin, not the VTR Power On Reset. TRST# must be held low while the SCH5636 is powering up so the registers can be set to their proper default values. If TRST# is high during power up, the JTAG Test Mode Data Registers may be set to unpredictable values, which may trigger unwanted test modes.

Care should be taken during VTR power up to insure that TRST# is asserted for a longer time then the VTR rise time due to capacitive loading.

33.4.3.2 Sync JTAG RESET

It can also be reset synchronously by a JTAG_CLK / TMS sequence, in accordance with the JTAG standard. A series of 5 successive JTAG_CLK rising edges, with TMS held high throughout, will accomplish this from any state.

The ARC JTAG port, upon entering its Reset state, will be prepared to accept an Instruction or Data transfer. It will also be disengaged from external circuitry, allowing it to operate normally.

The initial contents of the Instruction Register are the IDCODE command (Ch). If a Data transfer is performed first after Reset, without an preceding Instruction transfer, then the IDCODE value will be loaded into its 32-bit shift register and presented serially, after which will appear the bits shifted in from TDI.

The initial contents of the Data registers are as listed in Table 33-1, "ARC JTAG Instruction Register Encodings," on page 319.

33.5 Interrupts

There are no interrupts assigned to the ARC JTAG block. Control of the processor is performed by monitoring, setting and clearing the H bit (Halt) in the Aux register STATUS32, and by register manipulations while the processor is halted.

However, any interrupt or other event that can be triggered by accessing registers (Core, Aux or Memory-Mapped) can be triggered through the JTAG port.

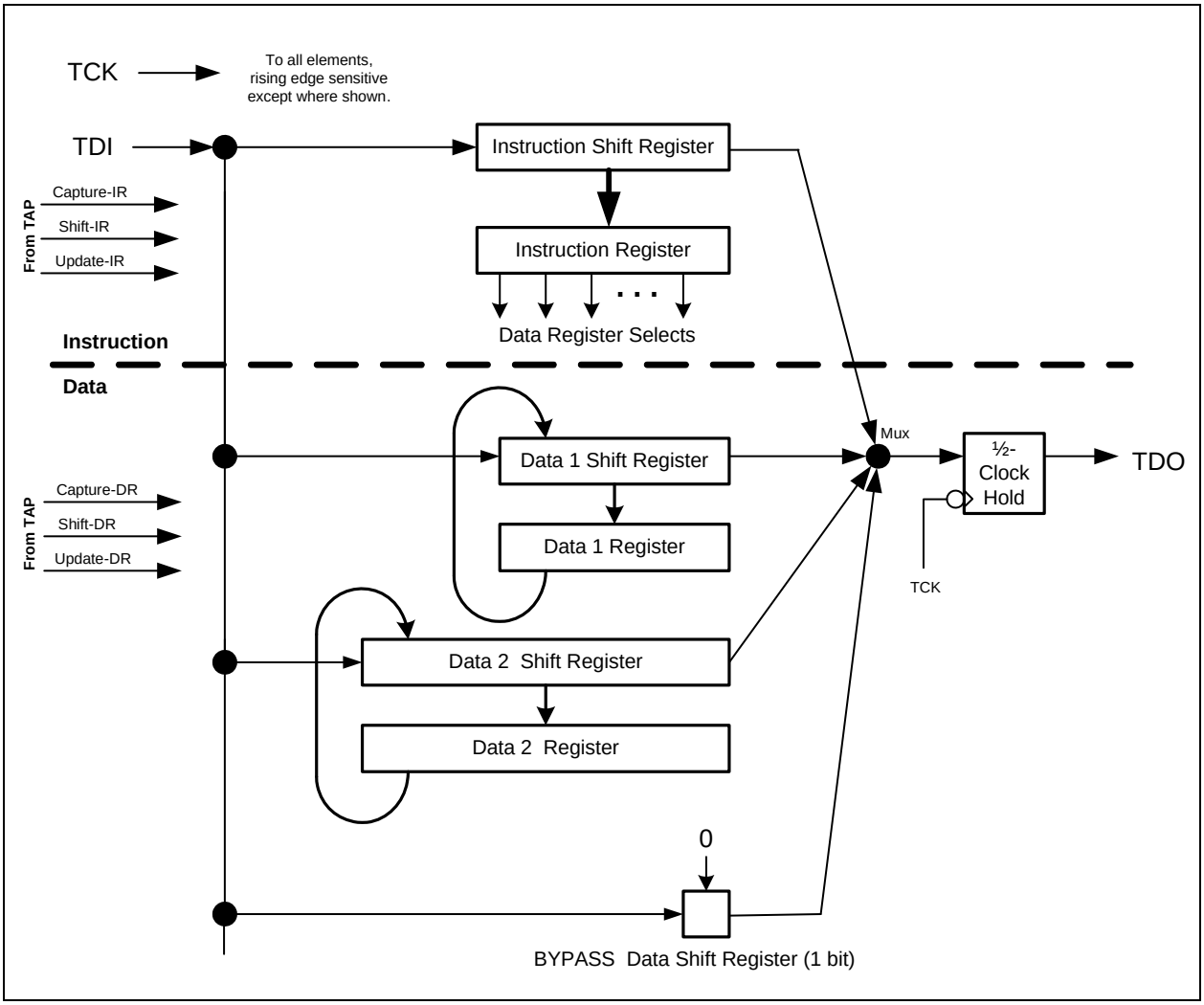
33.6 JTAG Background

The following is a simplified description, intended to provide background for the ARC JTAG port. For full details, see the JTAG specification (IEEE Standards 1149.1 and 1149.1b).

33.6.1 INTERNAL STRUCTURE

A JTAG port operates by transferring information serially into and out of an Instruction register and one or more Data registers. These registers are connected in parallel with each other, and can be of arbitrary length. See Figure 33-2.

FIGURE 33-2: STRUCTURE OF A JTAG PORT (SIMPLIFIED)



The protocol for shifting information makes a distinction between an Instruction transfer (to/from a single Instruction register) and a Data transfer (to/from one of several Data registers). The Instruction register is handled separately because it selects which specific Data register is accessed by subsequent Data transfers.

In daisy-chained JTAG controllers, the Instruction registers form one chain, and the currently-selected set of Data registers in each JTAG controller combine to form a second chain. To shorten the Data chain when not all JTAG controllers are of interest, a mandatory one-bit Data register called BYPASS is provided. There is no bypassing for the Instruction chain, so its full length must be shifted as each new instruction is transferred anywhere. Selecting the BYPASS Data register is the equivalent of a No-Operation instruction for a JTAG controller, and this instruction is always defined as a '1' in all Instruction register bits.

Each entity called a "Register" actually consists of two parts: the Register itself, and an associated Shift Register which connects to TDI and TDO. The Register may load from, and/or source information in parallel to, the Shift Register. These two parts are the same length, meaning that (for example) a 5-bit Register will be associated with a 5-bit Shift Register.

The Instruction register and the Data registers respond to decoded state signals from the TAP Controller sub-block (Section 33.6.2), which represent sub-steps of a transfer. The sub-steps they perform are **Capture**, which loads the shift register in parallel, **Shift**, which shifts information in from TDI and out on TDO, and **Update**, which writes information from the Shift Register in parallel. The **Capture-IR**, **Shift-IR** and **Update-IR** controls affect only the Instruction register. The **Capture-DR**, **Shift-DR** and **Update-DR** controls affect only the Data register that is currently selected by the contents of the Instruction register.

33.6.2 TAP CONTROLLER AND PROTOCOL

The JTAG protocol is driven by the level of the TMS (Test Mode Select) input pin at each rising edge of the JTAG_CLK clock. This is the responsibility of the TAP Controller section of the JTAG controller, which performs state transitions as illustrated in the state diagram in Figure 33-3. States whose names end with “IR” affect the Instruction register (the right-most column of states in Figure 33-3), and those ending with “DR” affect a Data register (the middle column in Figure 33-3). Note that the TMS signal goes in parallel to all JTAG ports in a chain, so they are always in the same protocol state. The sequence of accessing any register is as follows:

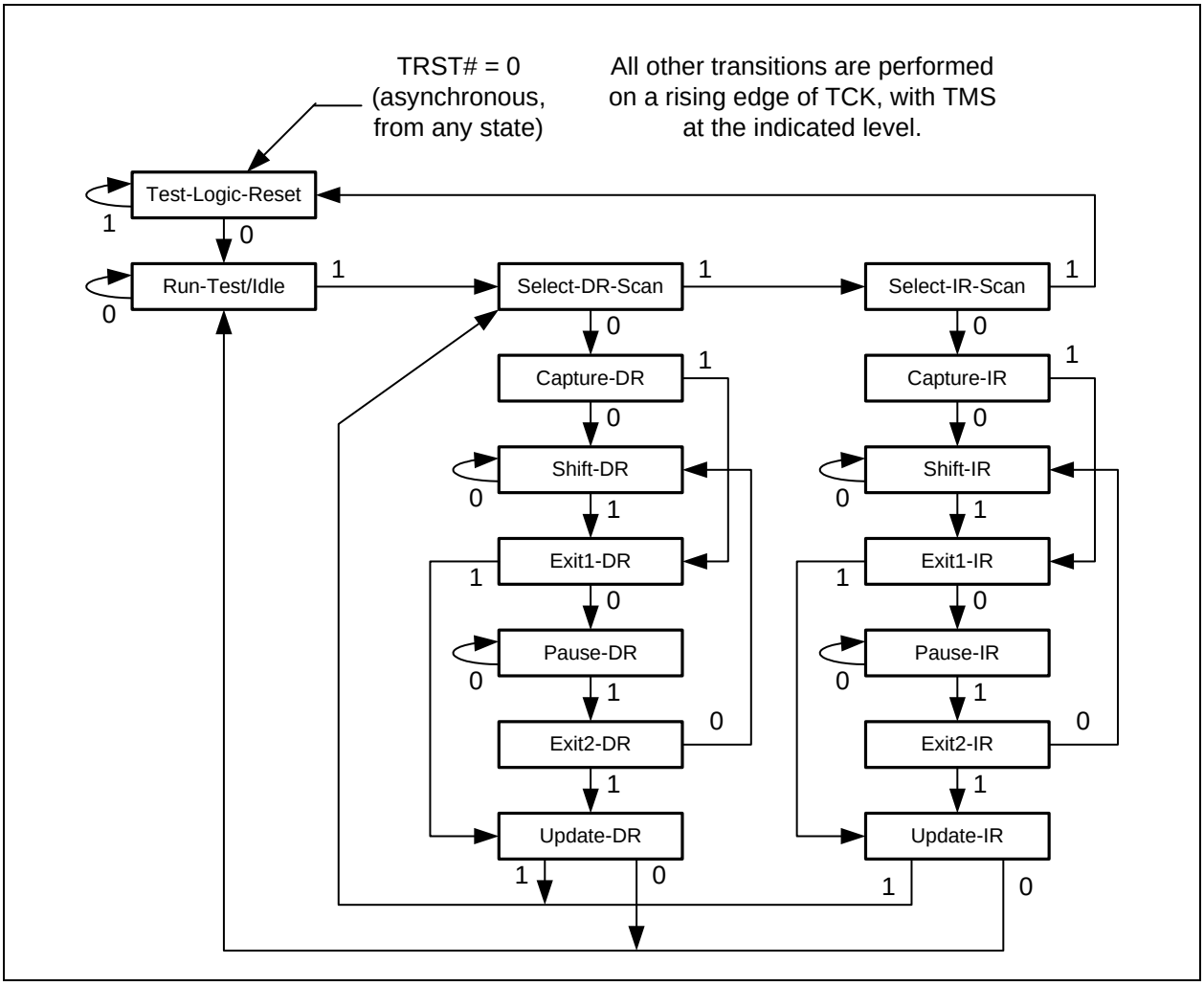
- Capture (IR or DR), which loads a shift register from its source in preparation for shifting it out. In the case of the Instruction register, this is a fixed value, and not the previous contents of the Instruction register. In the case of the BYPASS Data register, this is a fixed ‘0’ value. The Capture state is transitory, being present for only one JTAG_-CLK cycle, once per transfer.
- Shift (IR or DR), which shifts the Captured information in the Shift Register out on the TDO pin while also shifting information in from the TDI pin. The registers (by convention) shift from left to right, so the least-significant bit of a value is transferred first. This state may be held arbitrarily (holding TMS=0) to shift as many bits as desired.
- Update (IR or DR), which loads a Register from its Shift Register after the shifting has completed. The Update state is transitory, being present for only one JTAG_CLK cycle, once per transfer.

There is also a Pause state (IR or DR) which may be used to exit and re-enter the Shift state without terminating the transfer in progress. This state may be held (TMS=0) in order to delay for any desired number of JTAG_CLK cycles.

Outside of Instruction or Data transfers, there are two states which may be entered and held. These are shown in the leftmost column in Figure 33-3.

- The Test-Logic-Reset state holds the JTAG logic in its reset state. This re-initializes the registers that are internal to the JTAG logic. This state is entered asynchronously by assertion of TRST# low, and it can be seen in Figure 33-3 that, from any other state, this state will be entered by 5 successive JTAG_CLK cycles with TMS held to ‘1’.
- Run-Test/Idle holds JTAG logic idle, but not reset, between transfers.

FIGURE 33-3: TAP CONTROLLER STATE DIAGRAM



33.6.3 INTERFACE TIMING EXAMPLE

Figure 33-4 illustrates the timing relationship between data shifting and the TAP Controller's Shift states, using a 1-bit Data register as an example. (This is in fact the exact situation when the BYPASS Data register is selected: refer to FIGURE 33-2: on page 315.)

The TAP Controller changes states on each rising edge of JTAG_CLK, traversing the state table in Figure 33-3 as directed by the TMS input signal from the external interface.

Previous to the waveform in Figure 33-4, the TAP Controller has already passed through a Capture-DR state, so the 1-bit Shift Register has been pre-loaded with a "Capture Value", either from its associated parallel Register or from another source. (For the BYPASS register, this would be a fixed '0'.)

At the first rising edge of JTAG_CLK in Figure 33-4, the Shift-DR state is being entered. As yet, no valid data needs to be present on TDI or TDO.

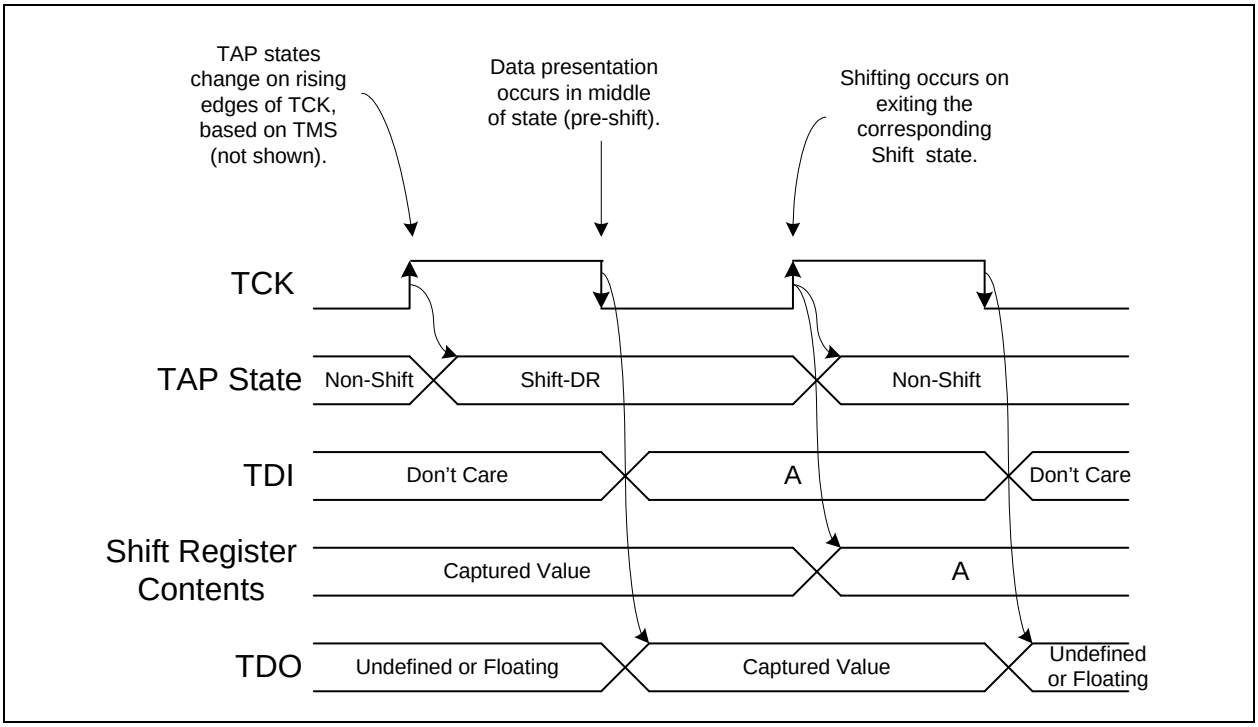
At the first falling edge of JTAG_CLK, while the Shift-DR state is active, the TDO pin begins presenting the least-significant bit of the Shift Register (the only bit, in this example), which is holding the Captured Value. At about this time also, the external interface will drive TDI to the desired new state for this Data register.

At the next rising edge of JTAG_CLK, the Shift-DR state is exited, and that same clock edge is used to actually perform the commanded shift. The TDI value “A” is shifted into the Shift Register. This same rising edge of JTAG_CLK is used by the external interface to shift in the Captured Value from TDO. The TDO output does not change yet, because it is held by a 1/2 clock delay stage (see FIGURE 33-2: on page 315), providing hold time for the external interface.

On the next falling edge, the TDO output changes. Since the Shift state is no longer present, TDO is not required at this time to present valid data, and in fact for an off-chip connection it is required to float at this time.

After this timing diagram completes, the TAP machine will continue to an Update-DR state, at which time the value A, now present in the Shift Register, will be written to its destination. (In the specific case of the BYPASS register, there is no destination, and that step will do nothing.)

FIGURE 33-4: TIMING ILLUSTRATION: 1-BIT DATA REGISTER



33.7 Registers

The ARC JTAG Port is defined in the ARC 600 External Interfaces Reference Manual, Chapter 2.

There are no JTAG registers accessible in any EC or AHB memory space. JTAG registers are accessible only through the JTAG pins themselves (Section 33.7.1, "Instruction Register," on page 318).

33.7.1 INSTRUCTION REGISTER

The Instruction Register is four bits wide. It selects among the implemented Data Registers as listed in Table 33-1. When the Tap Controller is placed into the Test-Logic-Reset state, the Instruction register is initialized to Ch, selecting the IDCODE Data Register.

Registers marked as MCHP reserved must not be modified. Modifications may lead to unpredictable and unwanted behavior.

TABLE 33-1: ARC JTAG INSTRUCTION REGISTER ENCODINGS

Instruction Register Contents	Data Register Selected	Function of Data Register	Width (Bits)	State on JTAG Reset (Hex)
0h	(Reserved: EXTEST)	Not implemented, but reserved as required by JTAG standard.	32	0000_0000h
1h	(Reserved: SAMPLE/PRELOAD)	Not implemented, but reserved as required by JTAG standard.	32	0000_0000h
2h	RESET TEST	RESET TEST Register (2h)	32	0000_0000h
3h	TEST - MCHP Reserved	—	32	0000_0000h
4h	TEST - MCHP Reserved	—	32	0000_0000h
5h	(Reserved)	(Reserved for future use.)	32	0000_0000h
6h	(Reserved)	(Reserved for future use.)	32	0000_0000h
7h	(Reserved)	(Reserved for future use.)	32	0000_0000h
8h	STATUS	STATUS Register (8h) Status of Current Debugger Transaction (Read-Only)	4	undefined (based on bus status)
9h	TRANSACTION COMMAND	TRANSACTION COMMAND Register (9h) Initiates / Specifies a Debugger Transaction	4	3
Ah	ADDRESS	ADDRESS Register (Ah) Address of a Debugger Transaction	32	0000_0000h
Bh	DATA	DATA Register (Bh) Data In / Data Out for Debugger Transactions	32	Out = 0000_0000h In = undefined
Ch	IDCODE	IDCODE Register (Ch) JTAG Standard IDCODE Register (Capture = Read-Only fixed value)	32	1000_24B1h
Dh	TEST	TEST REGISTER 4 / Reset Register (Dh)	32	0000_0000h
Eh	TEST - MCHP Reserved	—	32	0000_0000h
Fh	BYPASS	BYPASS Register (Fh) JTAG Standard BYPASS Register (Capture = Read-Only ‘0’)	1	0

33.7.2 JTAG DEBUG DATA REGISTERS

Note: Unfortunately, ARC names one of its JTAG Debug Data registers “DATA”. To avoid confusion, while maintaining the terminology in both ARC and JTAG documentation, the term “Data register” will refer to any of the JTAG Data registers, and the term “DATA register” (all upper-case) will refer to the specific JTAG Data register that is selected by Instruction Register = B. See Section 33.7.2.2, "DATA Register (Bh)," on page 320.

The Debug Data Register set of the ARC JTAG Port provide the means for an external JTAG-connected debugger system to monitor and control the execution of a program. Using the JTAG Data registers ADDRESS, DATA, TRANSACTION COMMAND and STATUS, the debugger can perform “transactions” to read or write:

- Any Aux Register, giving it the ability to start, halt or step a program, and alter the PC and/or program status
- Any addressable memory or I/O location, as an aligned 32-bit value
- Any Core Register, if the processor is in a halted state

To write to a specific register or a memory location, the debugger will place the desired register number or memory address into the ADDRESS register, place the value to be written into the DATA register, and then trigger the transfer by placing the direction and addressing space (Core register / Aux register / Memory) into the TRANSACTION COMMAND register. It will then read the STATUS register until it indicates that the transaction is finished.

To read from a specific register or memory location, the debugger will place the desired register number or memory address into the ADDRESS register, and trigger the transfer by placing the direction and addressing space (Core register / Aux register / Memory) into the TRANSACTION COMMAND register. It will then read the STATUS register until it indicates that the transaction is finished, and read the DATA register to access the value.

Optimizations are possible in repeated accesses, because of the actions of the ADDRESS and DATA registers, as described in Section 33.7.2.1 and Section 33.7.2.2.

33.7.2.1 ADDRESS Register (Ah)

The ADDRESS register is a 32-bit register which receives from the debugger either a Core Register number, an Aux Register number or an address in the Memory space (memory or I/O).

Note:	As a memory address, the low-order 2 bits of the ADDRESS register are ignored (assumed by hardware to be 00), and a full 32-bit value is referenced at that location. There is no way for the debugger to specify a smaller width of data, and so a write to a single byte (for example) is performed using a read transaction followed by a write transaction, preserving the values of the unaffected bytes.

After use, the ADDRESS register automatically increments, by 1 if a register was accessed, and by 4 if a memory location was accessed. Therefore, as long as the JTAG TAP Controller is not brought to the Test-Logic-Reset state between accesses, it is not necessary to provide a new ADDRESS register value between transactions involving successive registers or memory locations. (The Test-Logic-Reset state must be avoided because it resets the value of the ADDRESS register.)

TABLE 33-2: ADDRESS REGISTER

INSTRUCTION REGISTER CONTENTS	Ah			32 bits			**REGISTER SIZE**	
POWER	VTR			0000_0000h			Async JTAG RESET **OR** Sync JTAG RESET **DEFAULT**	
BIT	**BIT31**	**BIT30**	**BIT29**	**...**		**BIT2**	**BIT1**	**BIT0**
JTAG TYPE	-	-	-	-	-	-	-	-
BIT NAME	Address[31:0]							

33.7.2.2 DATA Register (Bh)

The DATA register is a 32-bit register which is the ARC JTAG Port's portal for data values that are being read or written by a transaction. When writing to a register or memory, the DATA register will be set up by the debugger before the transaction is triggered. When reading from a register or memory, the DATA register will be read by the debugger as the last step of the transaction. See APPLICATION NOTE: on page 314.

The DATA register is not affected at the end of a write transaction, so (for example) to fill successive locations with the same value it is not necessary to provide it again, as long as the Test-Logic-Reset of the JTAG TAP Controller is not entered (which would clear it).

TABLE 33-3: DATA REGISTER

INSTRUCTION REGISTER CONTENTS	Bh			32 bits			REGISTER SIZE	
POWER	VTR			0000_0000h			Async JTAG RESET OR Sync JTAG RESET DEFAULT	
BIT	BIT31	BIT30	BIT29	…		BIT2	BIT1	BIT0
JTAG TYPE	-	-	-	-	-	-	-	-
BIT NAME	Data[31:0]							

33.7.2.3 TRANSACTION COMMAND Register (9h)

The TRANSACTION COMMAND register is written by the debugger to trigger a transaction. It is a 4-bit register, which is written with one of the values in Table 33-5 to specify the direction and addressing space of the transaction.

TABLE 33-4: DATA REGISTER

INSTRUCTION REGISTER CONTENTS	9h			4 bits	REGISTER SIZE
POWER	VTR			0h	Async JTAG RESET OR Sync JTAG RESET DEFAULT
BIT	BIT3	BIT2	BIT1	BIT0	
JTAG TYPE	-	-	-	-	
BIT NAME	Command[3:0]				

TABLE 33-5: TRANSACTION COMMAND REGISTER ENCODINGS

Encoding (Binary)	Transaction Type
0000	Write to Memory space
0001	Write to a Core register
0010	Write to an Aux register
0011	No Operation
0100	Read from Memory space
0101	Read from a Core register
0110	Read from an Aux register
0111	(obsolete Write form)
1000	(obsolete Read form)
(other)	Reserved

33.7.2.4 STATUS Register (8h)

The STATUS register is a 4-bit read-only register. It is read by the debugger to determine when a transaction has completed internally, and when the next transaction may be started. It also provides additional status information useful to the debugger.

TABLE 33-6: STATUS REGISTER

INSTRUCTION REGISTER CONTENTS	8h		4 bits	**REGISTER SIZE**
POWER	VTR		0h	Async JTAG RESET **OR** Sync JTAG RESET **DEFAULT**
BIT	**BIT3**	**BIT2**	**BIT1**	**BIT0**
JTAG TYPE	R	R	R	R
BIT NAME	-PC	-RD	FL-	ST-

(ST): Stalled

1 = The current transaction is stalled (busy)

0 = The current transaction is not stalled (not busy)

(FL): Failure

1 = The transaction has failed

0 = The transaction has not failed

A transaction will fail if it attempts to access a Core register while the processor is running. Bus errors should also set this bit.

(RD): Ready

1 = The transaction is finished (ready)

0 = The transaction is not finished

(PC): PC_SEL

This bit has no direct hardware effect. It displays the state of the PC_SEL signal, which is bit 0 of the write-only Aux register PCPORT (Aux Register #24h). This bit is initialized to ‘1’ on a processor reset, and is used internally by the debugger system as a means to communicate configuration information.

33.7.3 JTAG STANDARD DATA REGISTERS

33.7.3.1 IDCODE Register (Ch)

This is a 32-bit read-only register containing the hex value 1000_24B1. It serves to identify the ARC JTAG Port as belonging to an ARC600 core, in a component containing one processor.

IDCODE registers are required to conform to the JTAG standard, and they contain an 11-bit Manufacturer ID number.

TABLE 33-7: IDCODE REGISTER

INSTRUCTION REGISTER CONTENTS	Ch			32 bits			**REGISTER SIZE**	
POWER	VTR			1000_24B1h			Async JTAG RESET **OR** Sync JTAG RESET **DEFAULT**	
BIT	**BIT31**	**BIT30**	**BIT29**	**…**		**BIT2**	**BIT1**	**BIT0**
JTAG TYPE	R	R	R	R	R	R	R	R
BIT NAME	IDCODE[31:0]							

33.7.3.2 BYPASS Register (Fh)

The BYPASS register consists only of a 1-bit shift register cell. The Capture-DR state clears it to '0' when selected. The Update-DR state does nothing.

The function of this register is to provide the minimum amount of delay (one bit of '0') when other JTAG ports on the chain are being exercised.

TABLE 33-8: BYPASS REGISTER

INSTRUCTION REGISTER CONTENTS	Fh	1 bit	**REGISTER SIZE**
POWER	VTR	1000_24B1h	Async JTAG RESET **OR** Sync JTAG RESET **DEFAULT**
BIT	**BIT0**		
JTAG TYPE			
BIT NAME	BYPASS		

33.7.4 JTAG TEST MODE DATA REGISTERS

JTAG Test Registers are 32-bit read/write registers that are used for test functions. These registers are always available to the JTAG port.

33.7.4.1 RESET TEST Register (2h)

The RESET TEST Register is a 32-bit register used to explicitly control reset functions inside the SCH5636. The default for this register is 0000_0000h.

TABLE 33-9: RESET TEST REGISTER

INSTRUCTION REGISTER CONTENTS	2h			32 bits			REGISTER SIZE	
POWER	VTR			0000_0000h			Async JTAG RESET **DEFAULT**	
BIT	**Bit 23**	**BIT22**	**BIT21**	**BIT20**	**BIT19**	**BIT18**	**BIT17**	**BIT16**
JTAG TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Test				Test			Test
BIT	**BIT 15**	**BIT 14**	**BIT 13**	**BIT 12**	**BIT 11**	**BIT 10**	**BIT 9**	**BIT 8**
JTAG TYPE	R/W	R/W	R/W	R	R	R	R/W	R/W
BIT NAME	Test	Test	Test	Test	Test	Test	Test	Test
BIT	**BIT 7**	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**
JTAG TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/
BIT NAME	Test	Test	Test	Test	POR EN	VTR POR	VCC RESET	Res

VCC RESET

Assert VCC RESET. If this bit is ‘0’ while the field POR EN in this register is ‘1’, a VCC RESET is forced. If this bit is ‘1’, the VCC RESET circuitry returns to its normal state.

VTR POR

Asserts VTR Power On Reset: When the VTR POR active low bit is asserted ‘0’ while the field POR EN in this register is ‘1’, forces a VTR Power On Reset. When the VTR POR active low bit de-asserted ‘1’, the VCC POR circuitry returns to its normal state.

POR EN

Power On Reset Enable. When ‘1’, the reset functions controlled by VCC RESET and VTR POR are enabled. When ‘0’, the VCC RESET and VTR POR fields in this register have no effect on the POR circuitry.

TESTAll TEST bits should be set to ‘0’ when writing this register.

33.7.4.2 TEST REGISTER 4 / Reset Register (Dh)

The RESET TEST Register is a 32-bit register used to explicitly control reset functions inside the SCH5636. The default for this register is 0000_0000h.

TABLE 33-10: TEST REGISTER 4 / RESET REGISTER

INSTRUCTION REGISTER CONTENTS	Dh			32 bits			REGISTER SIZE	
POWER	VTR			0000_0000h			Async JTAG RESET **DEFAULT**	
BIT	**BIT31**	**BIT30**	**BIT29**	**BIT28**	**BIT27**	**BIT26**	**BIT25**	**BIT24**
JTAG TYPE	R	R	R	R	R	R	R	R
BIT NAME	Test	Test	Res	Test	Res	Res	Res	Test
BIT	**BIT23**	**BIT22**	**BIT21**	**BIT20**	**BIT19**	**BIT18**	**BIT17**	**BIT16**
JTAG TYPE	R	R	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Test	Test	Test	Test	Test	Test	Test	Test
BIT	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**
JTAG TYPE	R/W	R/W	R	R	R/W	R/W	R/W	R/W
BIT NAME	Test	Test	Test	Test	Test	Test		
BIT	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
JTAG TYPE	R	R	R	R/W	R/W	R/W	R/W	R/W
BIT NAME	Test	Test	Test	Test	Test	Test	Test_ XNOR_ En	ARC_ Fast_ Reset

ARC_Fast_Reset

If this bit is ‘1b’, the reset going to the ARC processor and select peripherals is reduced from its nominal 20ms duration. If this bit is ‘0b’, the ARC reset is stretched by the nominal delay.

Test_XNOR_En

If this bit is ‘1b’, the Device-Under-Test XNOR chain test mode is enabled. If this bit is ‘0b’, the XNOR mode is disabled. See Section 41.12, “XNOR Chain,” on page 682.

Note: Once the XNOR chain is enabled, a power cycle is required to re-establish JTAG operation.

33.8 JTAG Standard Port Discovery

This section provides information that is not unique to ARC, but is part of the JTAG standard, and is provided for information.

The Discovery process will identify each JTAG controller that has an IDCODE register. Part of what needs to be derived is the length of the Instruction register in each of the JTAG ports. If this cannot be derived from the IDCODE values, or if some JTAG ports do not have an IDCODE register, then the missing lengths must be provided by other means.

In the Test-Logic-Reset state, a JTAG port is required to initialize its Instruction register to select the IDCODE Data register if present, or if it is not present, then to select the BYPASS Data register.

The IDCODE Data register:

- Must be exactly 32 bits in length
- Must have ‘1’ in its first (least-significant) bit
- Must not have the pattern 000011111111 (FFh) in its first (least-significant) 12 bits.
- Will contain a completely definitive port identification, because 11 bits of it are a Manufacturer ID number assigned by the JEDEC standards organization.

A BYPASS Data register access will initialize its 1-bit shift register to ‘0’ at the Capture-DR state, effectively making the BYPASS register appear to be 1-bit read-only ‘0’.

Discovery, therefore, consists of the external JTAG host doing the following:

- Place the chain of JTAG controllers into the Test-Logic-Reset state.
- Do a Data register access, without an Instruction register access first.
  - This Data access will shift in 8 bits of ones, followed by all zeroes for the duration of the discovery phase.
- While shifting, examine the data appearing on TDO for IDCODE values.
  - A ‘0’ indicates a JTAG port that has no IDCODE register. Collect only this bit, and note that the JTAG port exists. Start looking for an IDCODE value at the next bit.
  - A ‘1’ indicates that an IDCODE register is coming. Collect this bit and the next 31 bits to identify the JTAG port. If, however, the value seen is 00h0000FF, then this is maintained to be the value provided originally on TDI, and indicates the end of the chain.

33.9 XNOR Chain

33.9.1 OVERVIEW

The XNOR Chain test mode allows users to confirm that all SCH5636 pins are in contact with the motherboard during assembly and test operations. The XNOR Chain test mode is enabled and disabled through the JTAG interface, using bit Test_XNOR_En in JTAG TEST REGISTER 4 / Reset Register (Dh).

An example of an XNOR Chain test structure is illustrated below in Figure 33-5. When the XNOR Chain test mode is enabled all pins except for the Excluded Pins shown in Section 33.9.2 are disconnected from their internal functions and forced as inputs to the XNOR Chain. This allows a single input pin to toggle the XNOR Chain output if all other input pins are held high or low. The XNOR Chain output is the GP031 pin.

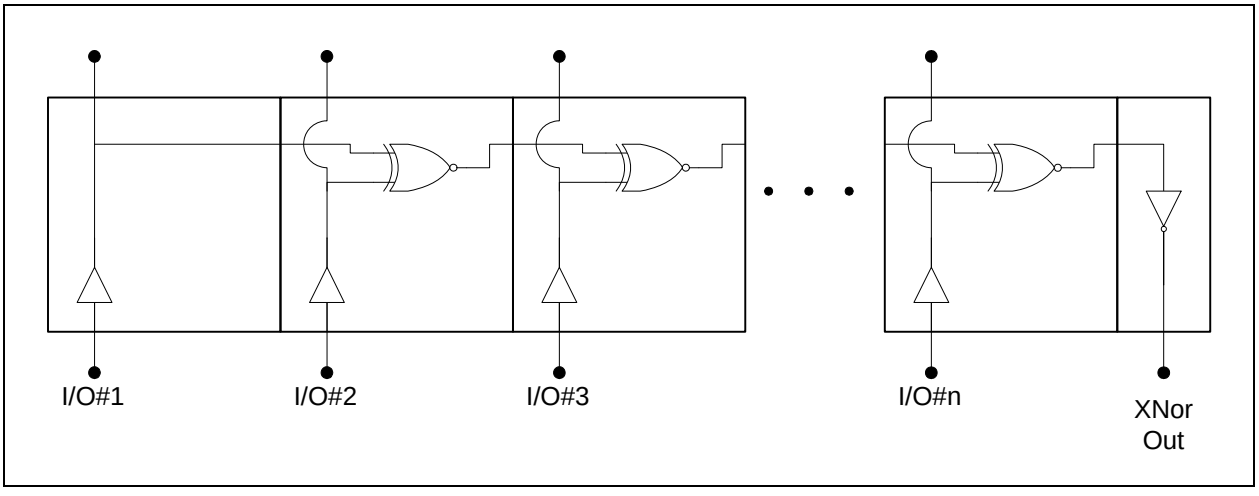
The tests that are performed when the XNOR Chain test mode is enabled require the board-level test hardware to control the device pins and observe the results at the XNOR Chain output pin; e.g., as described in Section 33.9.3, "Test Procedure," on page 327.

33.9.2 EXCLUDED PINS

The following pins are XNOR Chain Excluded Pins:

- POWER PLANE pins, VR_CAP, PECI V_{REF}
- TRST#.
- Analog pins: REMOTE1-, REMOTE1+, REMOTE2-, REMOTE2+, V_IN

FIGURE 33-5: XNOR CHAIN TEST STRUCTURE



33.9.3 TEST PROCEDURE

33.9.3.1 Setup

1. Connect the VSS and AGND pins to ground.
2. Connect the VCC0, VCC1, and VCC2 pins to an unpowered 3.3V power source.
3. Connect an oscilloscope or voltmeter to the GP031 pin.
4. All other pins should be tied to ground.

Warning: Ensure power supply is off during Setup.

33.9.3.2 Testing

1. Turn on the 3.3V power source.
2. Enable the XNOR Chain through the JTAG interface (Test_XNOR_En in JTAG TEST REGISTER 4 / Reset Register (Dh)). Note that at this point all inputs to the XNOR Chain are low and the output on the GP031 pin is high (refer to the Initial Configuration row in Table 33-11, "Toggling Inputs in Descending Pin Order").
3. Bring the highest numbered pin (N) high, where N is the number of pins to be tested as described in Note 33-1. The output on the GP031 pin should toggle (refer to Step 1 in Table 33-11.
4. In descending pin order successively bring each input high. As shown in Table 33-11 the GP031 pin toggles after each step. Continue until all inputs are high. The output on the GP031 pin is high (refer to the Final Configuration in Table 33-11).
5. The current state of the chip is now represented by the Initial Configuration row in Table 33-12, "Toggling Inputs in Ascending Pin Order".
6. Each input should now be brought low, starting at pin one (Step N+1) and continuing in ascending pin order until all inputs are low. The output on the GP031 pin is high (refer to the Final Configuration in Table 33-12.
7. Exit the XNOR Chain Test Mode by cycling VTR power.

TABLE 33-11: TOGGLING INPUTS IN DESCENDING PIN ORDER

	Pin Number (Note 33-1)							**GP031**
	N	**N - 1**	**N - 2**	**N - 3**	**N - 4**	**...**	**1**	
Initial Configuration	L	L	L	L	L	L	L	H
Step 1	H	L	L	L	L	L	L	L
Step 2	H	H	L	L	L	L	L	H
Step 3	H	H	H	L	L	L	L	L
Step 4	H	H	H	H	L	L	L	H
Step 5	H	H	H	H	H	L	L	L
...	H	H	H	H	H	…	L	…
Step N-1	H	H	H	H	H	H	L	L
Final Configuration	H	H	H	H	H	H	H	H

TABLE 33-12: TOGGLING INPUTS IN ASCENDING PIN ORDER

	Pin Number (Note 33-1)							GP031
	1	2	3	4	5	...	N	
Initial Configuration	H	H	H	H	H	H	H	H
Step N+1	L	H	H	H	H	H	H	L
Step N+2	L	L	H	H	H	H	H	H
Step N+3	L	L	L	H	H	H	H	L
Step N+4	L	L	L	L	H	H	H	H
Step N+5	L	L	L	L	L	H	H	L
...	L	L	L	L	L	…	H	…
Step N+(N-1)	L	L	L	L	L	L	H	L
Final Configuration	L	L	L	L	L	L	L	H

Note 33-1 pin numbers in these tables represent the number of pins to be tested and do not include the pins listed in Section 33.9.2, "Excluded Pins," on page 326.

34.0 SERIAL DEBUG PORT

34.1 General Description

The Serial Debug Port serially transmits MCU-originated diagnostic vectors to an external debug trace system.

The Serial Debug Port consists of the Debug Data Register, Debug Control Register, a Parallel-to-Serial Converter, a Clock/Control Interface and a two-pin external interface (Debug_CLK, Debug_DAT). See Figure 34-1.

34.2 Power, Clocks and Reset

34.2.1 POWER DOMAIN

This block is powered by the VTR Power Supply.

See Section 5.9, "Registers," on page 43 for details on power domains.

34.2.2 CLOCKS

This block has twp clock inputs, the EC Bus Clock and the 64.52MHz MCLK. The output clock is divided down from MCLK.

See Section 5.5, "Clock Sources," on page 32 for details on clocks.

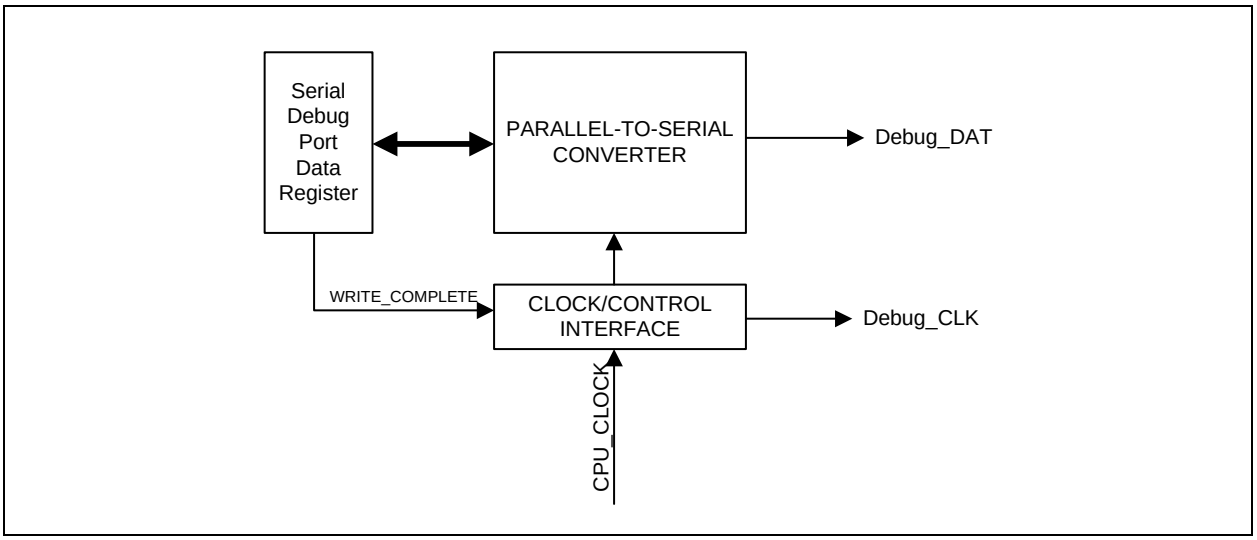
34.2.3 RESET

This block is reset on a nWDT_RST.

See Section 5.9, "Registers," on page 43 for details on reset.

34.3 Block Diagram

FIGURE 34-1: SERIAL DEBUG PORT BLOCK DIAGRAM



34.4 Block Diagram Port List

TABLE 34-1: SERIAL DEBUG PORT PORT LIST

Signal Name	Direction	Description
Debug Clock	OUTPUT	Derived from the EC Bus Clock
Debug Data	OUTPUT	Serialized Data shifter out by the Debug Clock
EC Bus Clock	INPUT	EC AHB Bus Clock

34.5 Interrupts

There are no interrupts from this block

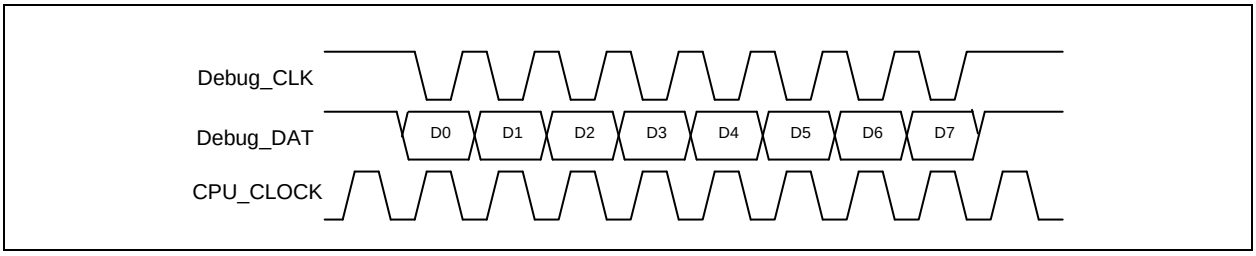
34.6 Functional Description

Writes to the Debug Data Register initiate an transfer cycle (Figure 34-2). Data from the Debug Data Register is shifted LSB first and is transmitted one byte per transfer cycle.

Data is transferred in one direction only from the Debug Data Register to the external interface. The data is shifted out on the clock edge selected by the EDGE_SEL bit in the Debug Control Register on page 331. Valid data is ensured on opposite edge of Debug_CLK after being shifted out. For example when the EDGE_SEL bit is '0' (default), Valid data is ensured on the falling edge of Debug_CLK. The Setup Time to the falling edge of Debug_CLK is 10ns minimum; the Hold Time is 1ns minimum.

The Debug_CLK and Debug_DAT outputs are '1' when the serial Debug Port is inactive. The transfer clock is EC Bus Clock.

FIGURE 34-2: DATA TRANSFER



34.7 Instance Description

There is one block instance defined in this chapter: Serial Debug Port.

34.8 Registers

The Serial Debug Port is instantiated on the EC AHB and has its own Logical Device Number and Base Address as indicated in Table 34-2:

TABLE 34-2: SERIAL DEBUG PORT BASE ADDRESS TABLE

Serial Debug Port **Instance**	**LDN from (**Table 4-3 on page 27**)**	**AHB Base Address**
MCU Debug Port	23h	F0_8C00h

The following table summarizes the registers allocated for the Serial Debug Port. The offset field in the following table is the offset from the Serial Debug Port's EC Base Address.

TABLE 34-3: SERIAL DEBUG PORT REGISTER SUMMARY

Register Name	Host I/O Offset	SPB Offset	Size	Type	Notes
Debug Data Register	-	00h	32	R/W	
Debug Control Register	-	04h	32	R/W	

34.9 Detailed Register Descriptions

34.9.1 DEBUG DATA REGISTER

The Debug Data Register is Read/Write. It always returns the last data written by the MCU or the power-on default '00h'.

TABLE 34-4: DEBUG DATA REGISTER

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	00h				32-bit		**EC SIZE**	
POWER	VTR				00h		nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	--	-	-	-	-
EC TYPE	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Data[7:0]							

Data

Debug data to be shifted out on the MCU Debug Port. While data is being shifted out, the SPB interface will 'hold-off' additional writes to the data register until the transfer is complete.

34.9.2 DEBUG CONTROL REGISTER

TABLE 34-5: DEBUG CONTROL REGISTER

HOST OFFSET	N/A				N/A		**HOST SIZE**	
EC OFFSET	04h				32-bit		**EC SIZE**	
POWER	VTR				00h		nWDT_RST **DEFAULT**	
BUS	EC SPB							
BYTE0 BIT	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
HOST TYPE	-	-	-	--	-	-	-	-
EC TYPE	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
BIT NAME	Reserved	IP_Delay			MCU_CLK		EDGE_SEL	EN

EN

0= Clock is disabled (Default)

1= Clock enabled

EDGE_SEL

0= Data is shifted out on the rising edge of the debug clock (Default)

1= Data is shifted out on the falling edge of the debug clock

MCU_CLK

The MCU Debug output clock is determined by this field according to Table 34-6, "MCU Debug Clocking".

TABLE 34-6: MCU DEBUG CLOCKING

MCU_CLK	MCU Debug Clock
0	32MHz
1	16MHz
2	8MHz
3	Reserved

IP_Delay

Inter-packet delay in terms of MCU Debug output clocks. A value of 0 provides a 1 clock inter-packet period, while a value of 7 provides 8 clocks between packets:

35.0 ELECTRICAL SPECIFICATIONS

35.1 Maximum Ratings*

Operating Temperature Range ..... 0^oC to $+70^oC$

Storage Temperature Range ..... -55^o to $+150^oC$

Lead Temperature Range ..... Refer to JEDEC Spec J-STD-020B

Positive Voltage on any pin, with respect to Ground ..... +5.5V

Negative Voltage on any pin, with respect to Ground ..... -0.3V

Supply Voltage Range V_{vtr} ..... 3.6 VDC

*Stresses above those listed above could cause permanent damage to the device. This is a stress rating only and functional operation of the device at any other condition above those indicated in the operation sections of this specification is not implied.

Note:	When powering this device from laboratory or system power supplies, it is important that the Absolute Maximum Ratings not be exceeded or device failure can result. Some power supplies exhibit voltage spikes on their outputs when the AC power is switched on or off. In addition, voltage transients on the AC power line may appear on the DC output. If this possibility exists, it is suggested that a clamp circuit be used.

TABLE 35-1: OPERATING CONDITIONS

Symbol	Parameter	MIN	TYP	MAX	Units
VBAT	Battery Backup Supply	2.0	3.0	3.6	V
VTR	Main Supply	2.97	3.3	3.63	V
PCI_CLK	PCI Clock		33		MHz
T_A	Operating Temperature	0		70	°C

35.1.1 HWM MAXIMUM RATINGS

Operating Temperature Range ..... 0°C to +70°C

Storage Temperature Range ..... -55°C to +150°C

Maximum avdd ..... +4V

Voltage on RTF<7:1>, RTS<7:1>, RTP<7:1>, RTM<7:1>, Diode Pins ..... avdd + 0.3V

Minimum Voltage on any Pin ..... -0.3V

Note:	Stresses above those listed could cause permanent damage to the device. This is a stress rating only and functional operation of the device at or above those listed in the operating sections of this specification are not implied or tested. When powering this device from liberator or test equipment, it is important that these Absolute Maximum ratings not be exceeded or device failure may result.

35.2 DC Specifications

35.2.1 ELECTRICAL CHARACTERISTICS

TABLE 35-2: DC ELECTRICAL CHARACTERISTICS

(T_A = 0°C - 70°C, V_{TR} = 3.3 VDC ±10%)

Buffer types that are 5V tolerant are listed in Table 3-1, “Signal Descriptions,” on page 11.

Parameter	Symbol	MIN	TYP	MAX	Units	Comments
I Type Input Buffer						
Low Input Level	V_{ILI}			0.8	V	TTL Levels
High Input Level	V_{IHI}	2.0			V	
I_M Type Input Buffer						
Low Input Level	V_{ILI}			0.8	V	TTL Levels
High Input Level	V_{IHI}	2.0			V	
O4 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 4 mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -4 mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
OD4 Type Buffer						
Low Output Level	V_{OL}			0.4	V	V_{OL} = 4 mA
Output Leakage	I_{OH}	-10		+10	µA	I_{OH} = 0 to V_{TR}
O8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8 mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -8 mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
OD8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	V_{OL} = 8 mA
Output Leakage	I_{OH}	-10		+10	µA	I_{OH} = 0 to V_{TR}
O12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -12mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
OD12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
OD16 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 16mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}

TABLE 35-2: DC ELECTRICAL CHARACTERISTICS (CONTINUED)

(T_A = 0°C - 70°C, V_{TR} = 3.3 VDC ±10%)

Buffer types that are 5V tolerant are listed in Table 3-1, “Signal Descriptions,” on page 11.

Parameter	Symbol	MIN	TYP	MAX	Units	Comments
IO4 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 4mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -4mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IOD4 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 4 mA
High Input Level	V_{IH}	2.0			V	
Low Input Level	V_{IL}			0.8	V	
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IO8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -8mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IOD8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8 mA
High Input Level	V_{IH}	2.0			V	
Low Input Level	V_{IL}			0.8	V	
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IO12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -12mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IOD12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
High Input Level	V_{IH}	2.0			V	
Low Input Level	V_{IL}			0.8	V	
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IOP14 Type Buffer						
High Input Level	V_{IH}	2.0			V	I_{OL} = 14mA
Low Input Level	V_{IL}			0.8	V	I_{OH} = -14mA
Low Output Level	V_{OL}	2.4		0.4	V	V_{IN} = 0 to V_{TR}
High Output Level	V_{OH}	-10			µA	
Output Leakage	I_{OL}			+10		

TABLE 35-2: DC ELECTRICAL CHARACTERISTICS (CONTINUED)

(T_A = 0°C - 70°C, V_{TR} = 3.3 VDC ±10%)

Buffer types that are 5V tolerant are listed in Table 3-1, "Signal Descriptions," on page 11.

Parameter	Symbol	MIN	TYP	MAX	Units	Comments
IO16 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 16mA
High Output Level	V_{OH}	2.4			V	I_{OH} = -16mA
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IOD16 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL}=16 mA
High Input Level	V_{IH}	2.0			V	
Low Input Level	V_{IL}			0.8	V	
Output Leakage	I_{OL}	-10		+10	µA	V_{IN} = 0 to V_{TR}
IO24 Type Buffer						I_{OL} = 24mA
Low Output Level	V_{OL}			0.4	V	I_{OH} = -24mA
High Output Level	V_{OH}	2.4			V	V_{IN} = 0 to V_{TR}
Output Leakage	I_{OL}	-10		+10	µA	
PCI_CLK Type Buffer	PCI_ICLK					See *PCI Local Bus Specification Rev. 2.2*
PCI_IO Type Buffers	PCI_IO PCI_O PCI_I					
PCI_OD Type Buffer	PCI_OD					
PROCHOT I Buffer (PECI_I)						All input and output voltages are a function of VREF buffer input.
Input voltage range	VIn	-0.3		V_{REF} + 0.3	V	
Low Input Level	VIL			0.275× V_{REF}	V	
High Input Level	VIH	0.725× V_{REF}			V	
PECI_IO						All input and output voltages are a function of VREF buffer input.
Input voltage range	VIn	-0.3		V_{REF} + 0.3	V	See PECI Specification.
Hysteresis	VHYS	0.1 × V_{REF}	0.2× V_{REF}		V	
Low Input VLevel	VIL			0.275× V_{REF}	V	
High Input Level	VIH	0.725× V_{REF}			V	
Low Output Level	VOL			0.25× V_{REF}	V	0.5mA < IOL < 1mA
High Output Level	VOH	0.75 × V_{REF}			V	IOH = -6mA

TABLE 35-2: DC ELECTRICAL CHARACTERISTICS (CONTINUED)

(T_A = 0°C - 70°C, V_{TR} = 3.3 VDC ±10%)

Buffer types that are 5V tolerant are listed in Table 3-1, "Signal Descriptions," on page 11.

Parameter	Symbol	MIN	TYP	MAX	Units	Comments
V_{REF} Buffer						Connects to VTT
Input Voltage, PECI	VI	0.95		1.26	V	
Input Voltage, AMD-TSI_{DDR2}		1.7		1.9	V	Processor dependent
Input Voltage, AMD-TSI_{DDR3}		1.42		1.575	V	
Input current	IDC			100	µA	
Input Low Current	ILEAK	-10		+10	µA	
OD_PH Type Buffer (PROCHOT#)						
Low Output Level	V_{OL}			0.3	V	I_{OL} = 23mA
High Output Level	V_{OH}		V_{REF}		V	Open Drain, V_{REF} = 1.2V

Note 35-1 All 5V Tolerant I-type & I/O-type input buffers can be pulled to 5 volts.

Note 35-2 All 5V Tolerant OD-type output buffers can be pulled to 5 volts.

Note 35-3 All 5V Tolerant O-type and I/O-type output buffers will only drive to 3.3 volts, even if pulled-up externally to 5 volts.

35.3 Power Consumption

TABLE 35-3: SCH5636 POWER CONSUMPTION

V_{CC}	V_{TR}	System "S" State	EC State	Clock State	Supply Current			Comments
						Typical (25^0 C)	MAX (70^0 C)	
3.3V	3.3V	S0-S2	Run	Ring OSC @ 64 MHz	VTR	15mA	22mA	
0V		S3	Run		VTR	11mA	17mA	
	0V	S5	Off	None	VBAT	2.5µA	4.5µA (@25^0C)	2.0V < V_{bat} < 3.0V

35.4 AC Specifications

AC Test Conditions

CAPACITANCE T_A = 25°C; fc = 1MHz; V_{cc} = 3.3 VDC

Parameter	Symbol	Limits			Units	Test Condition
		MIN	TYP	MAX		
Clock Input Capacitance	C_{IN}			20	pF	All pins except pin under test tied to AC ground
Input Capacitance	C_{IN}			10	pF	
Output Capacitance	C_{OUT}			20	pF	

35.5 HWM Operating Specifications

TABLE 35-4: HWM ELECTRICAL SPECIFICATIONS

T_A = 0°C to +70°C, AVDD = 3.3V ± 10%, VDDD = 1.2V ± 10% Unless Otherwise Specified							
Parameter	**Symbol**	**Min.**	**Typ**	**Max**	**Units**	**Conditions**	**Notes**
ADC (General)							
Resolution			11		bit		
Temperature Conversion							
Internal Diode Accuracy			0.5	<±2	°C	-10°C < T_A < 125°C	
Internal Diode Resolution			0.125		°C		
External Diode Accuracy			0.25	±1	°C	60°C < T_{DIODE} < 100°C, 0°C < T_A < 125°C	
				±3	°C	-64°C< T_{DIODE} < 191°C	
External Diode Resolution			0.0625		°C	12-bit conversion	
Series Resistance Error Correction	R_{SERIES}			100	Ohm	Total series with diode lines to block including parasitic routing impedance	
Capacitive Load	C_{LOAD}		2.2	2.5	nF	Connected across external diodes.	Note 35-5

Note 35-4 The nature of the ADC implies that the Full Scale voltage is equal to V_{REF} - 1 LSB or, V_{REF} * 2047 / 2048.

Note 35-5 The Capacitive Load will be dependent on the beta of the transistor being measured. The given value is for a beta greater or equal to 0.4. If the beta is less than 0.4, then the supported value is reduced. The current IP can support up to 10nF for a diode-connected transistor.

36.0 TIMING DIAGRAMS

36.1 LPC Clock and Reset Timing

FIGURE 36-1: PCI CLOCK TIMING

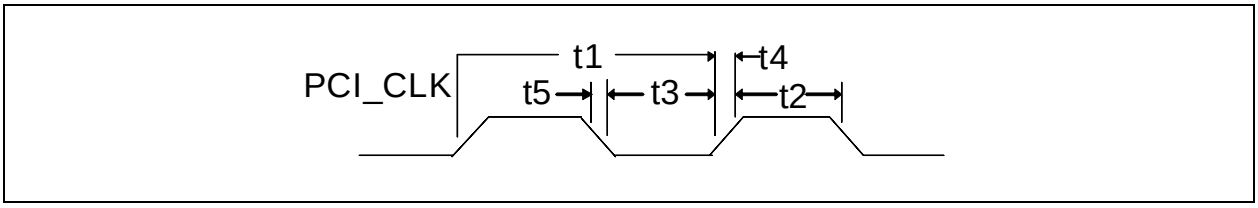


TABLE 36-1: PCI CLOCK TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Period	30		33.3	nsec
t2	High Time	11			
t3	Low Time				
t4	Rise Time			3	
t5	Fall Time				

FIGURE 36-2: RESET TIMING

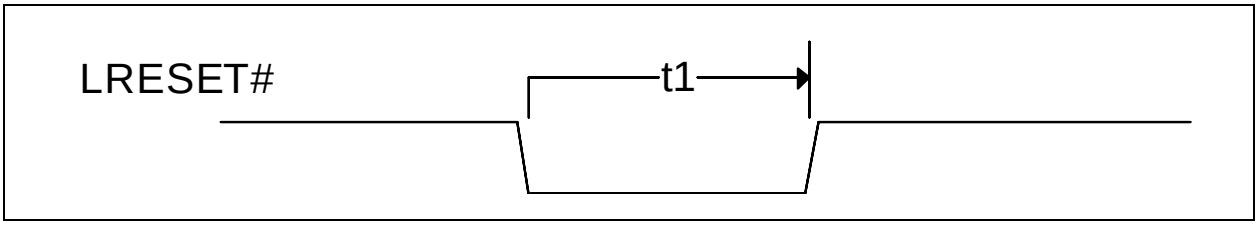


TABLE 36-2: RESET TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	LRESET# width	1			ms

36.2 LPC Bus Timing

FIGURE 36-3: OUTPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS

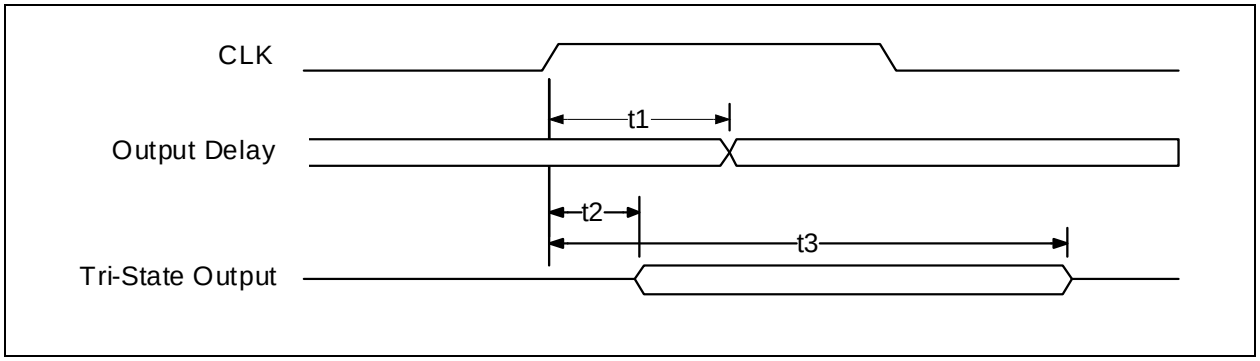


TABLE 36-3: OUTPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	CLK to Signal Valid Delay – Bused Signals	2		11	ns
t2	Float to Active Delay				
t3	Active to Float Delay			28	

FIGURE 36-4: INPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS

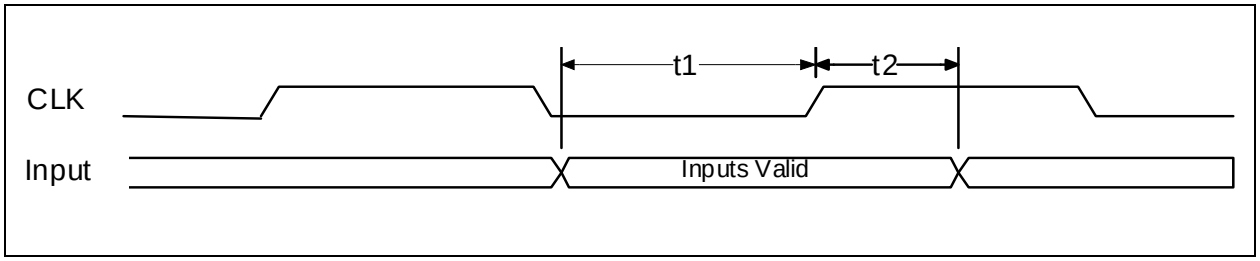
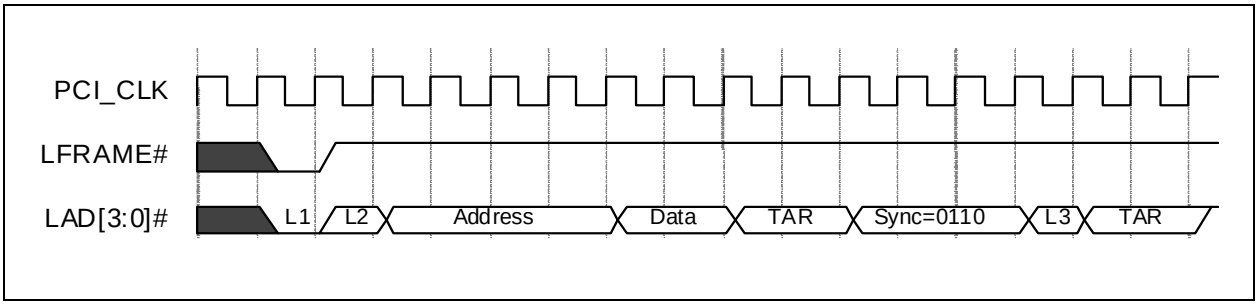


TABLE 36-4: INPUT TIMING MEASUREMENT CONDITIONS, LPC SIGNALS PARAMETERS

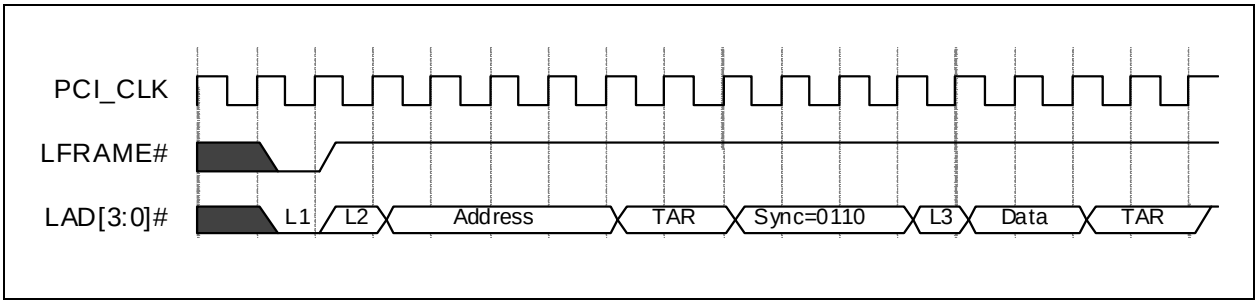
Name	Description	MIN	TYP	MAX	Units
t1	Input Set Up Time to CLK – Bused Signals	7			ns
t2	Input Hold Time from CLK	0			

FIGURE 36-5: I/O WRITE



Note: L1=Start; L2=CYCTYP+DIR; L3=Sync of 0000

FIGURE 36-6: I/O READ



Note: L1=Start; L2=CYCTYP+DIR; L3=Sync of 0000

FIGURE 36-7: DMA Request Assertion Through LDRQ#

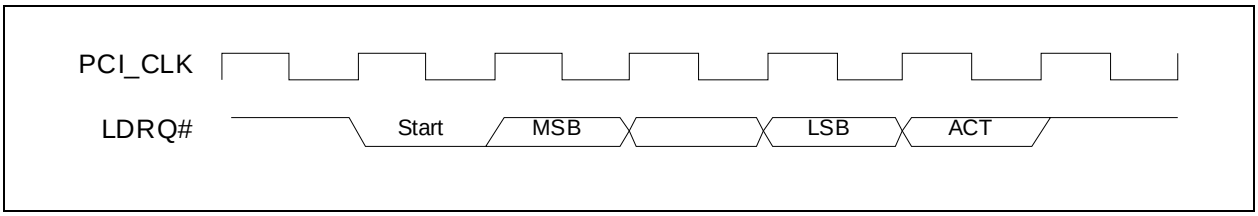
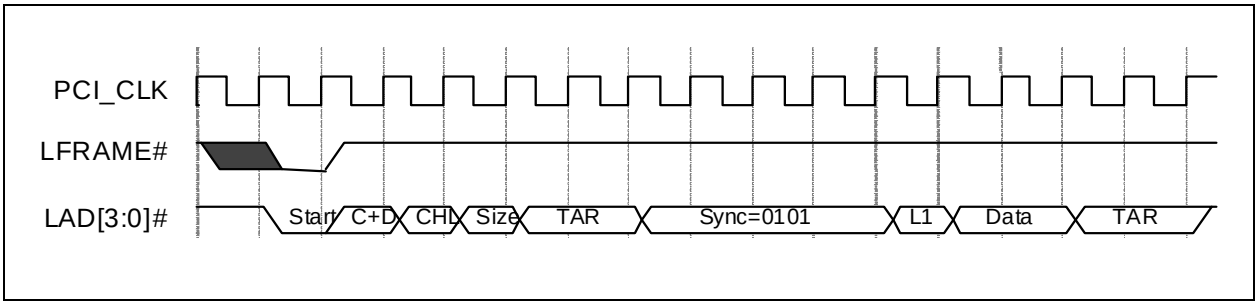
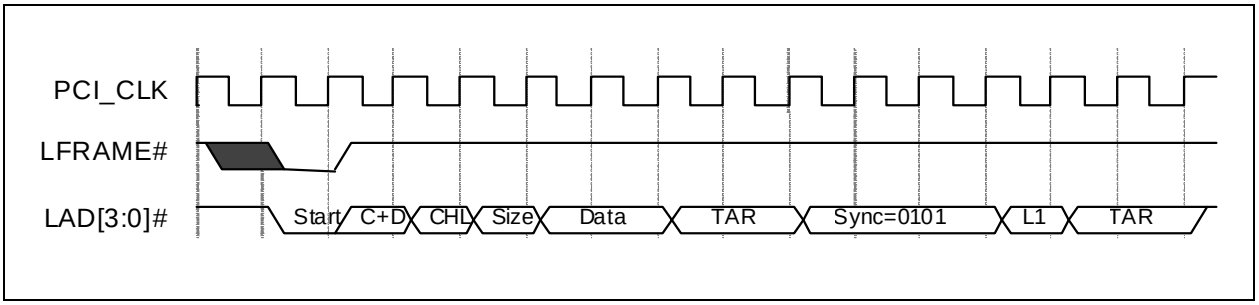


FIGURE 36-8: DMA Write (First Byte)



Note: L1=Sync of 0000

FIGURE 36-9: DMA READ (FIRST BYTE)



Note: L1=Sync of 0000

36.3 Serial IRQ Timing

FIGURE 36-10: SETUP AND HOLD TIME

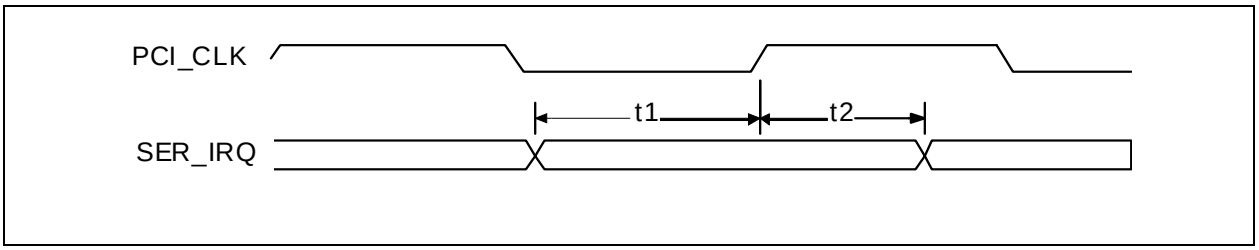


TABLE 36-5: SETUP AND HOLD TIME

Name	Description	MIN	TYP	MAX	Units
t1	SER_IRQ Setup Time to PCI_CLK Rising	7			nsec
t2	SER_IRQ Hold Time to PCI_CLK Rising	0			

36.4 Floppy Disk Controller Timings

FIGURE 36-11: FLOPPY DISK DRIVE TIMING (AT MODE ONLY)

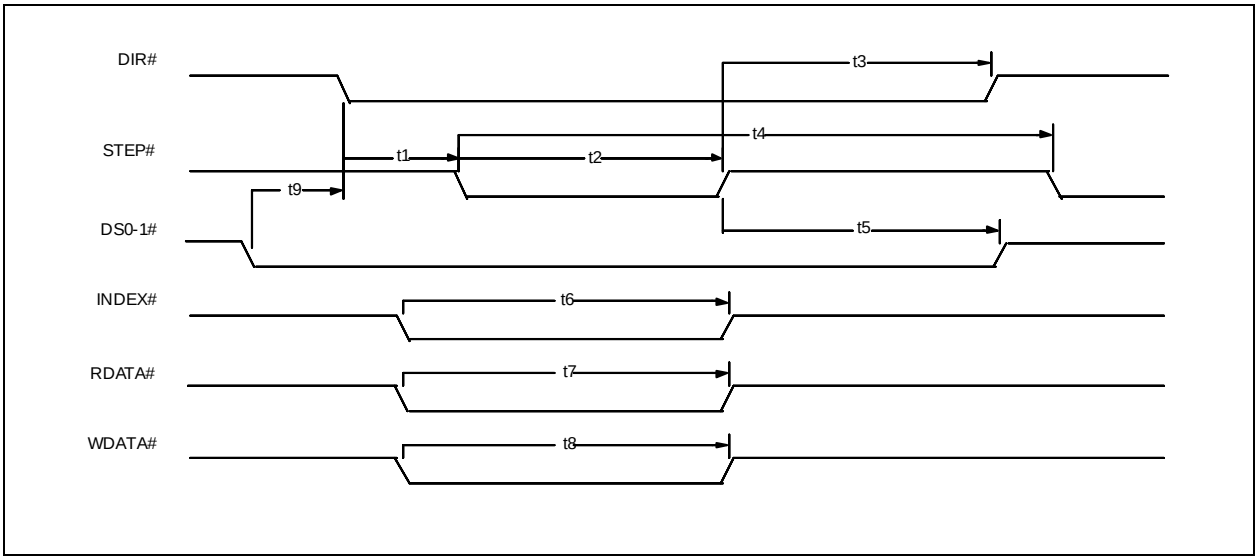


TABLE 36-6: FLOPPY DISK DRIVE TIMING (AT MODE ONLY) PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	DIR# Set Up to STEP Low		4		X*
t2	STEP# Active Time Low		24		X*
t3	DIR# Hold Time after STEP#		96		X*
t4	STEP# Cycle Time		132		X*
t5	DS0# & DS1# Hold Time from STEP# Low (Note)		20		X*
t6	INDEX# Pulse Width		2		X*
t7	RDATA# Active Time Low		40		ns
t8	WDATA# Write Data Width Low		.5		Y*
t9	DS0# & DS1#, Setup Time DIR# Low (Note)	0			ns
*X specifies one MCLK period and Y specifies one WCLK period.					

MCLK = 16 x Data Rate (at 500 kb/s MCLK = 8 MHz)

WCLK = 2 x Data Rate (at 500 kb/s WCLK = 1 MHz)

Note: The DS0 &DS1 setup and hold times must be met by software.

36.5 Parallel Port Timings

36.5.1 EPP PARALLEL PORT TIMINGS

FIGURE 36-12: EPP 1.9 Data or Address Write Cycle

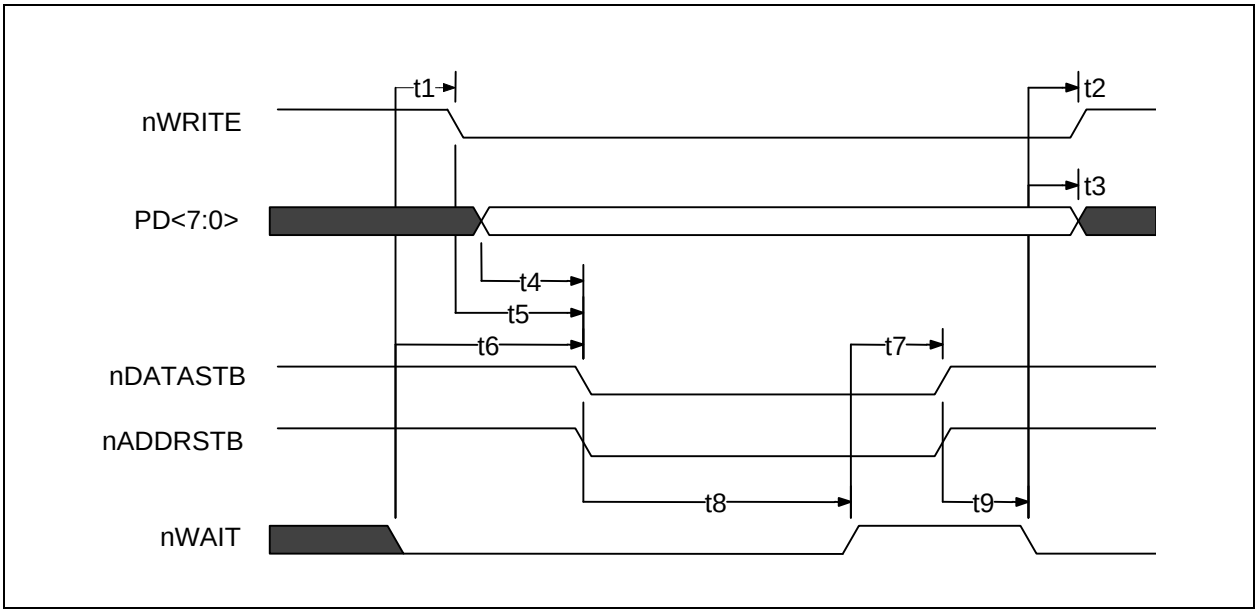


TABLE 36-7: EPP 1.9 DATA OR ADDRESS WRITE CYCLE PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	nWAIT Asserted to nWRITE Asserted (See Note)	60		185	ns
t2	nWAIT Asserted to nWRITE Change (See Note)	60		185	ns
t3	nWAIT Asserted to PDATA Invalid (See Note)	0			ns
t4	PDATA Valid to Command Asserted	10			ns
t5	nWRITE to Command Asserted	5		35	ns
t6	nWAIT Asserted to Command Asserted (See Note 36-1)	60		210	ns
t7	nWAIT Deasserted to Command Deasserted (See Note 36-1)	60		190	ns
t8	Command Asserted to nWAIT Deasserted	0		10	µs
t9	Command Deasserted to nWAIT Asserted	0			ns

Note 36-1 nWAIT must be filtered to compensate for ringing on the parallel bus cable. nWAIT is considered to have settled after it does not transition for a minimum of 50 nsec.

FIGURE 36-13: EPP 1.9 Data or Address Read Cycle

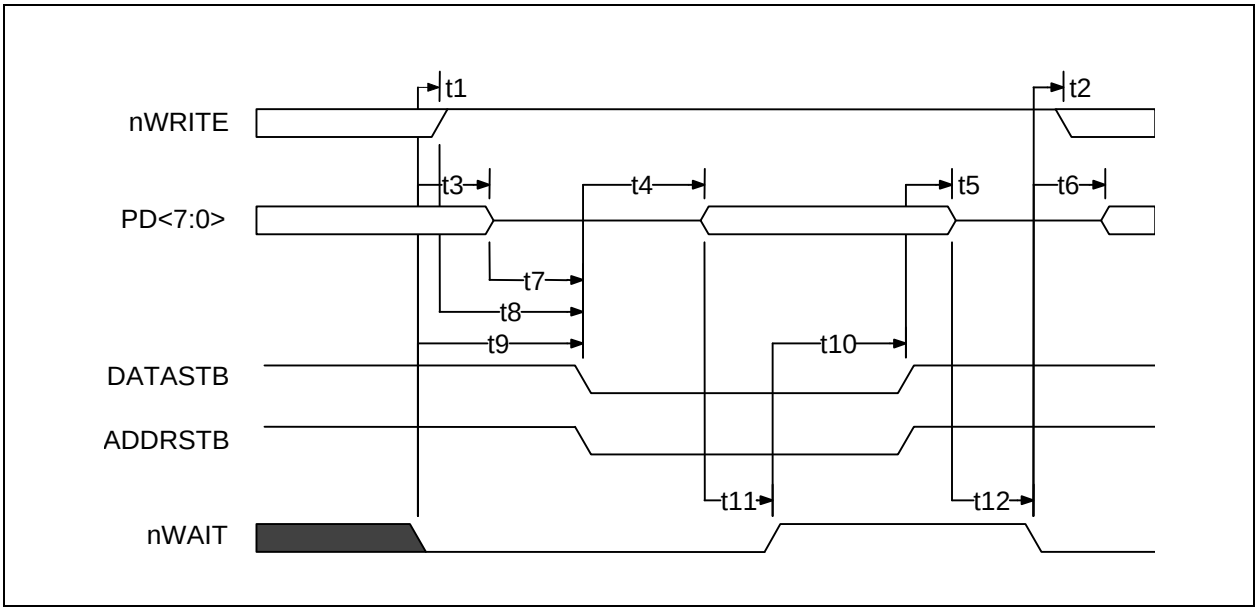


TABLE 36-8: EPP 1.9 DATA OR ADDRESS READ CYCLE

Name	Description	MIN	TYP	MAX	Units
t1	nWAIT Asserted to nWRITE Deasserted	0		185	ns
t2	nWAIT Asserted to nWRITE Modified (Note 36-1, Note 36-2)	60		190	ns
t3	nWAIT Asserted to PDATA Hi-Z (Note 36-1)	60		180	ns
t4	Command Asserted to PDATA Valid	0			ns
t5	Command Deasserted to PDATA Hi-Z	0			ns
t6	nWAIT Asserted to PDATA Driven (Note 36-1)	60		190	ns
t7	PDATA Hi-Z to Command Asserted	0		30	ns
t8	nWRITE Deasserted to Command	1			ns
t9	nWAIT Asserted to Command Asserted	0		195	ns
t10	nWAIT Deasserted to Command Deasserted (Note 36-1)	60		180	ns
t11	PDATA Valid to nWAIT Deasserted	0			ns
t12	PDATA Hi-Z to nWAIT Asserted	0			µs

Note 36-1 nWAIT is considered to have settled after it does not transition for a minimum of 50 ns.

Note 36-2 When not executing a write cycle, EPP nWRITE is inactive high.

FIGURE 36-14: EPP 1.7 Data or Address Write Cycle

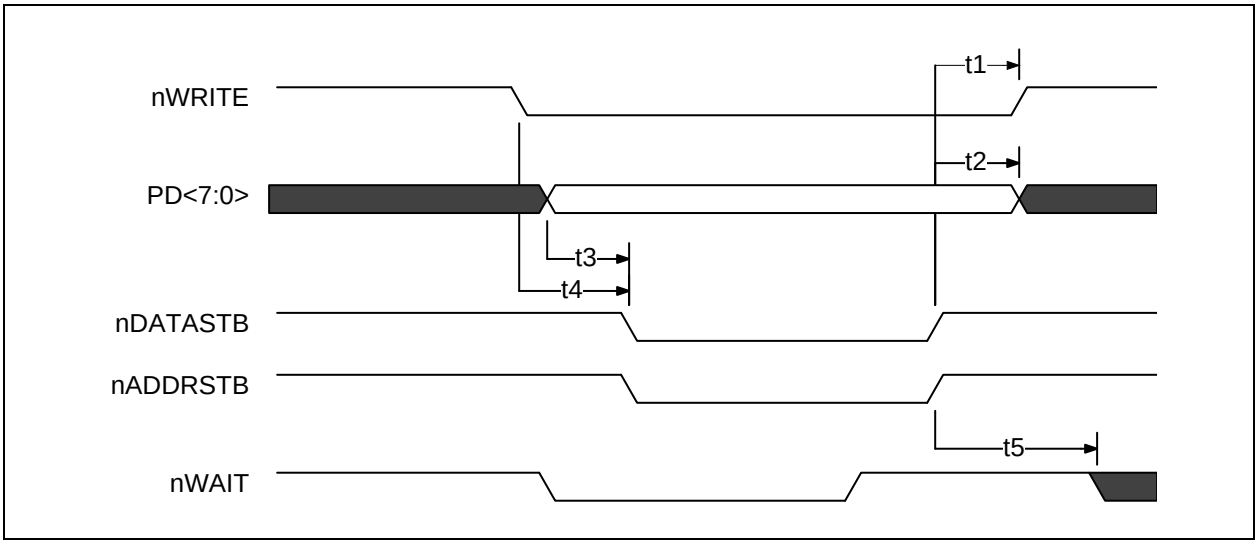


TABLE 36-9: EPP 1.7 DATA OR ADDRESS WRITE CYCLE

Name	Description	MIN	TYP	MAX	Units
t1	Command Deasserted to nWRITE Change	0		40	ns
t2	Command Deasserted to PDATA Invalid	50			ns
t3	PDATA Valid to Command Asserted	10		35	ns
t4	nWRITE to Command	5		35	ns
t5	Command Deasserted to nWAIT Deasserted	0			ns

FIGURE 36-15: EPP 1.7 Data or Address Read Cycle

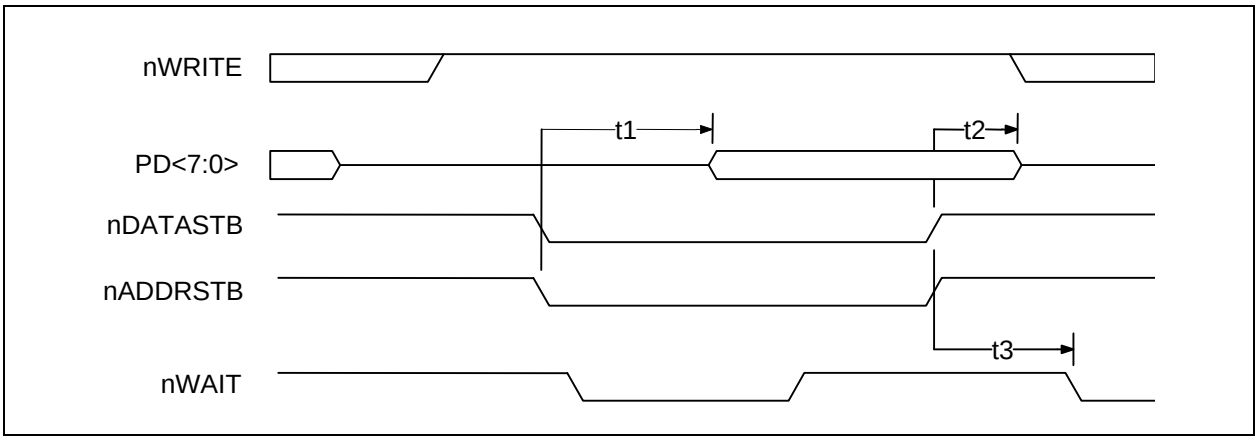


TABLE 36-10: EPP 1.7 DATA OR ADDRESS READ CYCLE PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Command Asserted to PDATA Valid	0			ns
t2	Command Deasserted to PDATA Hi-Z	0			ns
t3	Command Deasserted to nWAIT Deasserted	0			ns

36.5.2 ECP PARALLEL PORT TIMING

Parallel Port FIFO (Mode 101)

The standard parallel port is run at or near the peak 500KBytes/sec allowed in the forward direction using DMA. The state machine does not examine ACK# and begins the next transfer based on Busy. Refer to FIGURE 36-16: on page 348.

ECP Parallel Port Timing

The timing is designed to allow operation at approximately 2.0 Mbytes/sec over a 15ft cable. If a shorter cable is used then the bandwidth will increase.

Forward-Idle

When the host has no data to send it keeps HostClk (STROBE#) high and the peripheral will leave PeriphClk (Busy) low.

Forward Data Transfer Phase

The interface transfers data and commands from the host to the peripheral using an interlocked PeriphAck and HostClk. The peripheral may indicate its desire to send data to the host by asserting nPeriphRequest.

The Forward Data Transfer Phase may be entered from the Forward-Idle Phase. While in the Forward Phase the peripheral may asynchronously assert the nPeriphRequest (nFault) to request that the channel be reversed. When the peripheral is not busy it sets PeriphAck (Busy) low. The host then sets HostClk (STROBE#) low when it is prepared to send data. The data must be stable for the specified setup time prior to the falling edge of HostClk. The peripheral then sets PeriphAck (Busy) high to acknowledge the handshake. The host then sets HostClk (STROBE#) high. The peripheral then accepts the data and sets PeriphAck (Busy) low, completing the transfer. This sequence is shown in FIGURE 36-17: on page 348.

The timing is designed to provide 3 cable round-trip times for data setup if Data is driven simultaneously with HostClk (STROBE#).

Reverse-Idle Phase

The peripheral has no data to send and keeps PeriphClk high. The host is idle and keeps HostAck low.

Reverse Data Transfer Phase

The interface transfers data and commands from the peripheral to the host using an interlocked HostAck and PeriphClk.

The Reverse Data Transfer Phase may be entered from the Reverse-Idle Phase. After the previous byte has been accepted the host sets HostAck (ALF#) low. The peripheral then sets PeriphClk (ACK#) low when it has data to send. The data must be stable for the specified setup time prior to the falling edge of PeriphClk. When the host is ready to accept a byte it sets HostAck (ALF#) high to acknowledge the handshake. The peripheral then sets PeriphClk (ACK#) high. After the host has accepted the data, it sets HostAck (ALF#) low, completing the transfer. This sequence is shown in FIGURE 36-18: on page 349.

Output Drivers

To facilitate higher performance data transfer, the use of balanced CMOS active drivers for critical signals (Data, HostAck, HostClk, PeriphAck, PeriphClk) are used in ECP Mode. Because the use of active drivers can present compatibility problems in Compatible Mode (the control signals, by tradition, are specified as open-drain), the drivers are dynamically changed from open-drain to push-pull. The timing for the dynamic driver change is specified in the *IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard, Rev. 1.14*, July 14, 1993, available from Microsoft. The dynamic driver change must be implemented properly to prevent glitching the outputs.

FIGURE 36-16: PARALLEL PORT FIFO TIMING

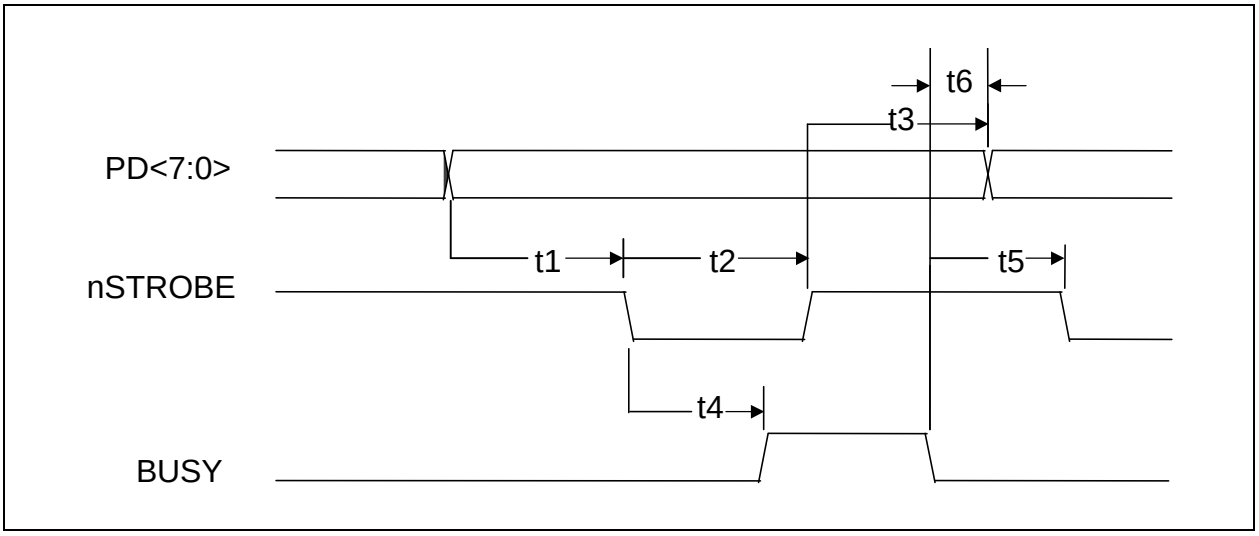


TABLE 36-11: PARALLEL PORT FIFO TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	PDATA Valid to STROBE# Active	600			ns
t2	STROBE# Active Pulse Width	600			ns
t3	PDATA Hold from STROBE# Inactive (See Note 36-1)	450			ns
t4	STROBE# Active to BUSY Active			500	ns
t5	BUSY Inactive to STROBE# Active	680			ns
t6	BUSY Inactive to PDATA Invalid (See Note 36-1)	80			ns

Note 36-1 The data is held until BUSY goes inactive or for time t3, whichever is longer. This only applies if another data transfer is pending. If no other data transfer is pending, the data is held indefinitely.

FIGURE 36-17: ECP PARALLEL PORT FORWARD TIMING

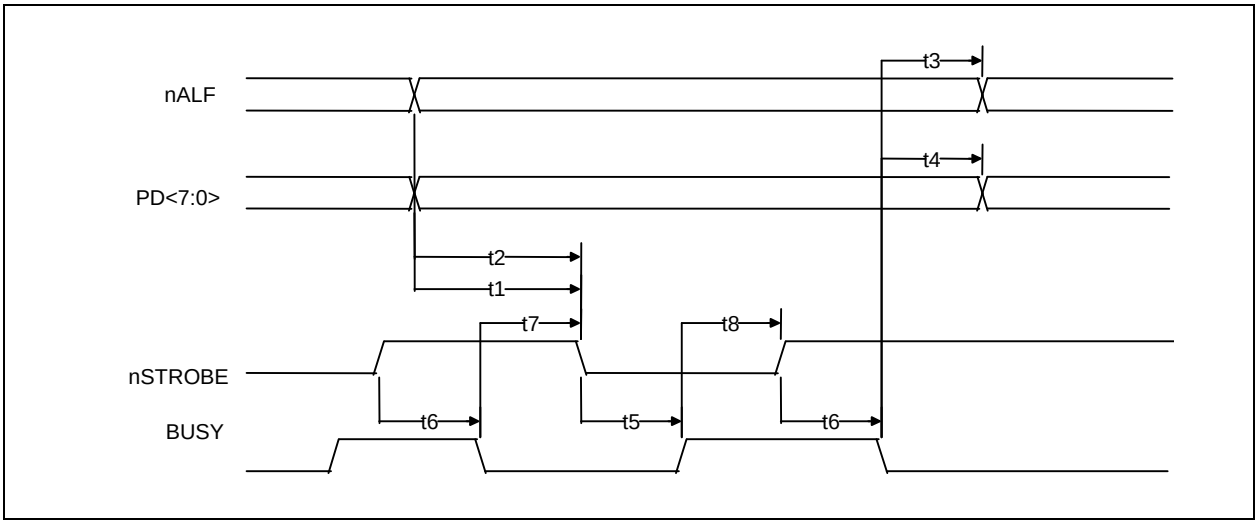


TABLE 36-12: ECP PARALLEL PORT FORWARD TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	ALF# Valid to STROBE# Asserted	0		60	ns
t2	PDATA Valid to STROBE# Asserted	0		60	ns
t3	BUSY Deasserted to ALF# Changed (Note 36-2, Note 36-3)	80		180	ns
t4	BUSY Deasserted to PDATA Changed (Note 36-2, Note 36-3)	80		180	ns
t5	STROBE# Asserted to Busy Asserted	0			ns
t6	STROBE# Deasserted to Busy Deasserted	0			ns
t7	BUSY Deasserted to STROBE# Asserted (Note 36-2, Note 36-3)	80		200	ns
t8	BUSY Asserted to STROBE# Deasserted (Note 36-3)	80		180	ns

Note 36-2 Maximum value only applies if there is data in the FIFO waiting to be written out.

Note 36-3 BUSY is not considered asserted or deasserted until it is stable for a minimum of 75 to 130 ns.

FIGURE 36-18: ECP PARALLEL PORT REVERSE TIMING

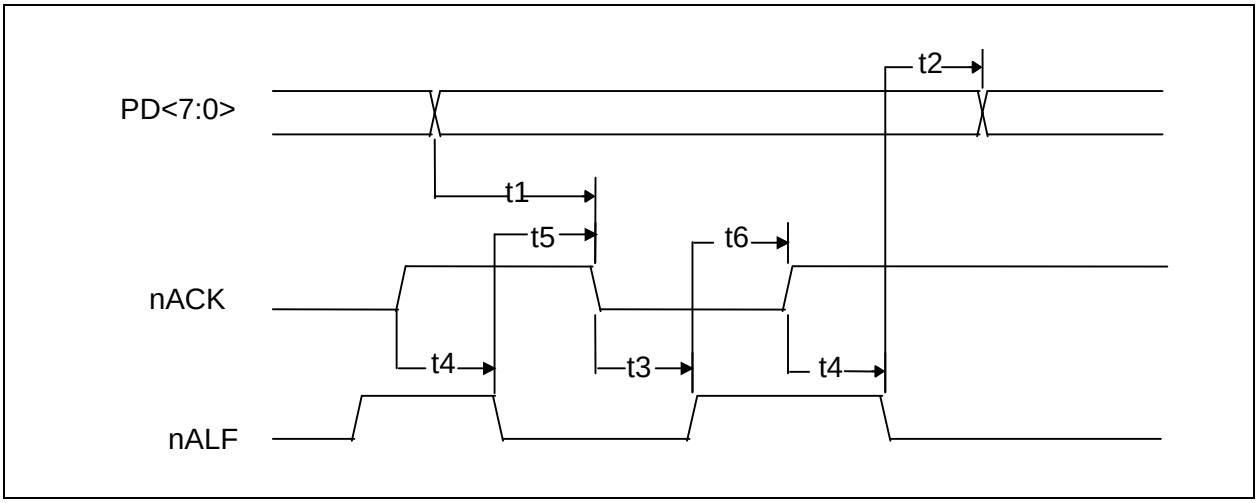


TABLE 36-13: ECP PARALLEL PORT REVERSE TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	PDATA Valid to ACK# Asserted	0			ns
t2	ALF# Deasserted to PDATA Changed	0			ns
t3	ACK# Asserted to ALF# Deasserted (Note 36-4, Note 36-5)	80		200	ns
t4	ACK# Deasserted to ALF# Asserted (Note 36-5)	80		200	ns
t5	ALF# Asserted to ACK# Asserted	0			ns
t6	ALF# Deasserted to ACK# Deasserted	0			ns

Note 36-4 Maximum value only applies if there is room in the FIFO and terminal count has not been received. ECP can stall by keeping ALF# low.

Note 36-5 ACK# is not considered asserted or deasserted until it is stable for a minimum of 75 to 130 ns.

36.6 Serial Port (UART) Data Timing

FIGURE 36-19: SERIAL PORT DATA

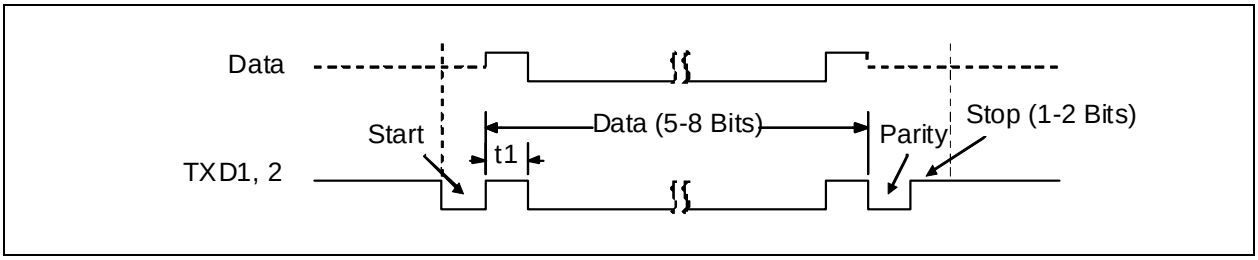


TABLE 36-14: SERIAL PORT DATA PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Serial Port Data Bit Time		t_{BR} (Note 36-6)		nsec

Note 36-6 t_{BR} is 1/Baud Rate. The Baud Rate is programmed through the divisor latch registers. Baud Rates have percentage errors indicated in Table 12-20, “UART Baud Rates (1.8432MHz source),” on page 136.

36.7 Keyboard/Mouse Port Timings

FIGURE 36-20: KEYBOARD/MOUSE RECEIVE/SEND DATA TIMING

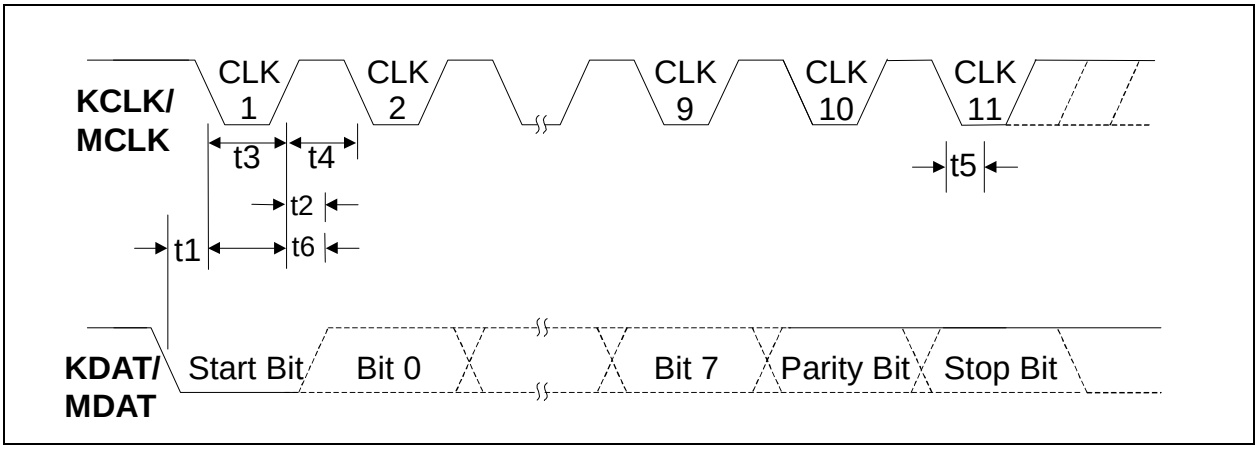


TABLE 36-15: KEYBOARD/MOUSE RECEIVE/SEND DATA TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Time from DATA transition to falling edge of CLOCK (Receive)	5		25	µsec
t2	Time from rising edge of CLOCK to DATA transition (Receive)	5		T4-5	µsec
t3	Duration of CLOCK inactive (Receive/Send)	30		50	µsec
t4	Duration of CLOCK active (Receive/Send)	30		50	µsec

TABLE 36-15: KEYBOARD/MOUSE RECEIVE/SEND DATA TIMING PARAMETERS (CONTINUED)

Name	Description	MIN	TYP	MAX	Units
t5	Time to keyboard inhibit after clock 11 to ensure the keyboard does not start another transmission (Receive)	>0		50	µsec
t6	Time from inactive to active CLOCK transition, used to time when the auxiliary device samples DATA (Send)	5		25	µsec

36.8 I²C/SMBus Timing

FIGURE 36-21: I²C/SMBUS TIMING

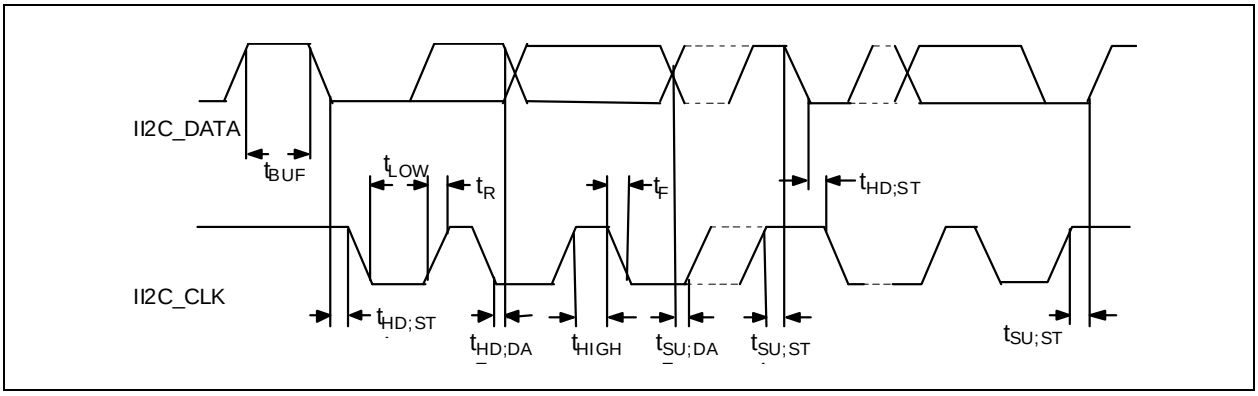


TABLE 36-16: I²C/SMBUS TIMING PARAMETERS

Symbol	Parameter	Standard-Mode		Fast-Mode		Unit
		MIN.	MAX.	MIN.	MAX.	
f_{SCL}	SCL Clock Frequency		100		400	kHz
t_{BUF}	Bus Free Time	4.7		1.3		µs
$t_{SU;STA}$	START Condition Set-Up Time	4.7		0.6		µs
$t_{HD;STA}$	START Condition Hold Time	4.0		0.6		µs
t_{LOW}	SCL LOW Time	4.7		1.3		µs
t_{HIGH}	SCL HIGH Time	4.0		0.6		µs
t_R	SCL and SDA Rise Time		1.0		0.3	µs
t_F	SCL and SDA Fall Time		0.3		0.3	µs
$t_{SU;DAT}$	Data Set-Up Time	0.25		0.1		µs
$t_{HD;DAT}$	Data Hold Time	0		0		µs
$t_{SU;STO}$	STOP Condition Set-Up Time	4.0		0.6		µs

36.9 Fan Tachometer Timing

FIGURE 36-22: FAN TACHOMETER INPUT TIMING

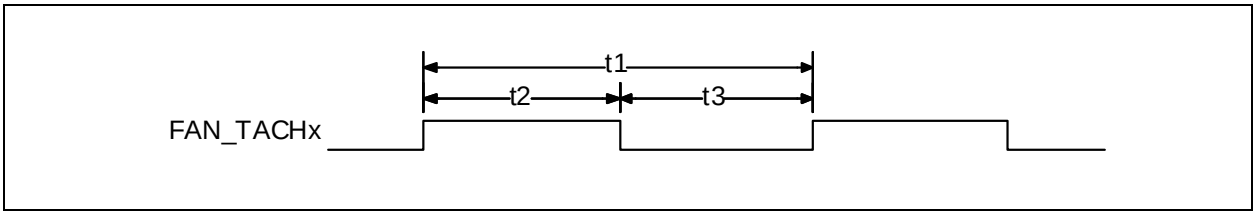


TABLE 36-17: FAN TACHOMETER INPUT TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Pulse Time	100			µsec
t2	Pulse High Time				
t3	Pulse Low Time	10			

Note 36-7 t_{TACH} is the clock used for the tachometer counter. It is 30.52 * prescaler, where the prescaler is programmed in the Fan Tachometer Timebase Prescaler register.

36.10 PWM Timing

FIGURE 36-23: PWM OUTPUT TIMING

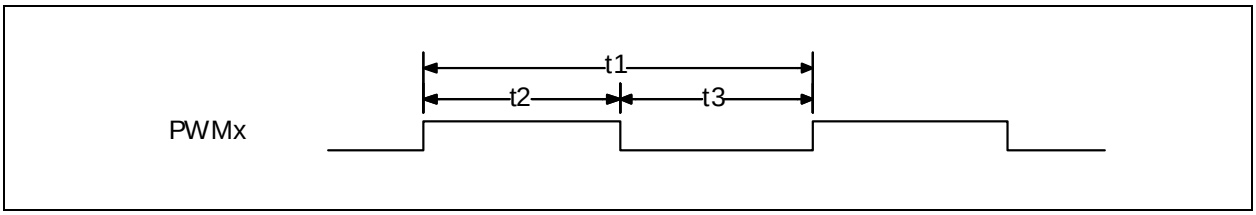


TABLE 36-18: PWM TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Period	31ns		23.3sec	
t_f	Frequency	0.04Hz		32MHz	
t2	High Time	0		11.65	sec
t3	Low Time	0		11.65	sec
t_d	Duty cycle	0		100	%

36.11 Serial Peripheral Interface (SPI) Timings

The following timings apply when CLKPOL=0, TCLKPH=0 and RCLKPH=1 (the default configuration).

FIGURE 36-24: SPI TIMING

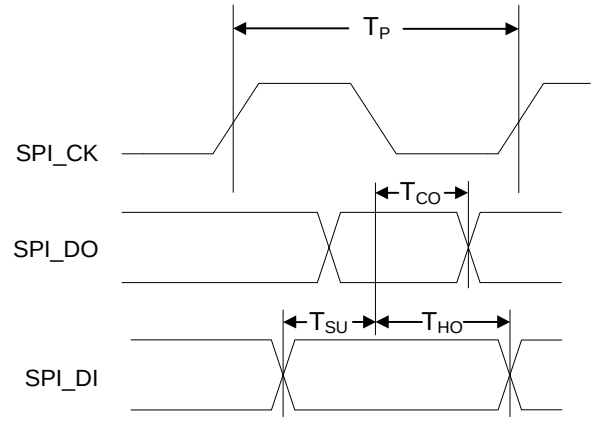


TABLE 36-19: SPI TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
T_P	Clock Period	31			ns
	Clock Duty Cycle	47		53	%
T_{CO}	Clock-to-out time, SPI Data Out, with respect to falling edge of SPI Clock	3		13	ns
T_{SU}	Setup time, SPI Data In, with respect to falling edge of SPI Clock	7.5			ns
T_{HO}	Hold time, SPI Data In, with respect to falling edge of SPI Clock	0			ns

Note: Maximum SPI Clock frequency is 32MHz.

36.12 Yellow and Green LED Interface

FIGURE 36-25: YELLOW/GREEN OUTPUT TIMING

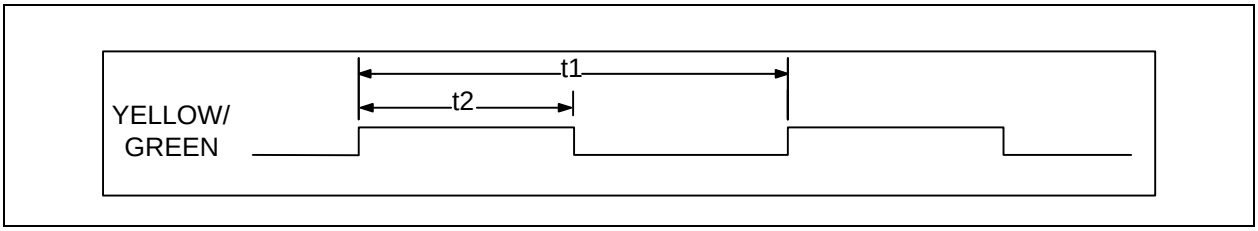


TABLE 36-20: YELLOW/GREEN OUTPUT TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Period		1	3.03 (Note 36-8)	sec
t2	Blink ON Time	0	0.5	1.52 (Note 36-8)	sec

Note 36-8 The blink rate is programmed through Bits[1:0] in LEDx register. When Bits[1:0]=00, LED is OFF. Bits[1:0]=01 indicates LED blink at 1Hz rate with a 50% duty cycle (0.5 sec ON, 0.5 sec OFF). When Bits[1:0]=11, LED is ON.

36.13 GPIO Timings

FIGURE 36-26: GPIO TIMING

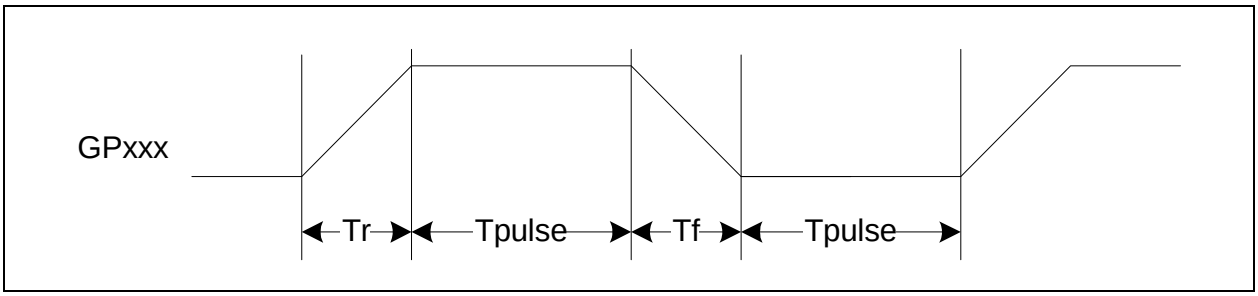


TABLE 36-21: GPIO TIMING PARAMETERS

Symbol	Parameter	MIN	TYP	MAX	Unit	Notes
t_R	GPIO Rise Time (push-pull)	1.3		2.6	ns	Pad type = IO4/IO8, C_L=10pF
t_F	GPIO Fall Time	1.2		2.6	ns	
t_R	GPIO Rise Time (push-pull)	0.9		1.8	ns	Pad type = IO12/IO16, C_L=10pF
t_F	GPIO Fall Time	0.9		2.0	ns	
t_{pulse}	GPIO Pulse Width	60			ns	

36.14 PWR_STATE Timings

The following representative timing is included to illustrate the relation of the PWR_STATE bit.

FIGURE 36-27: NORMAL POWER SUPPLY SEQUENCE (S0-S3-S0)

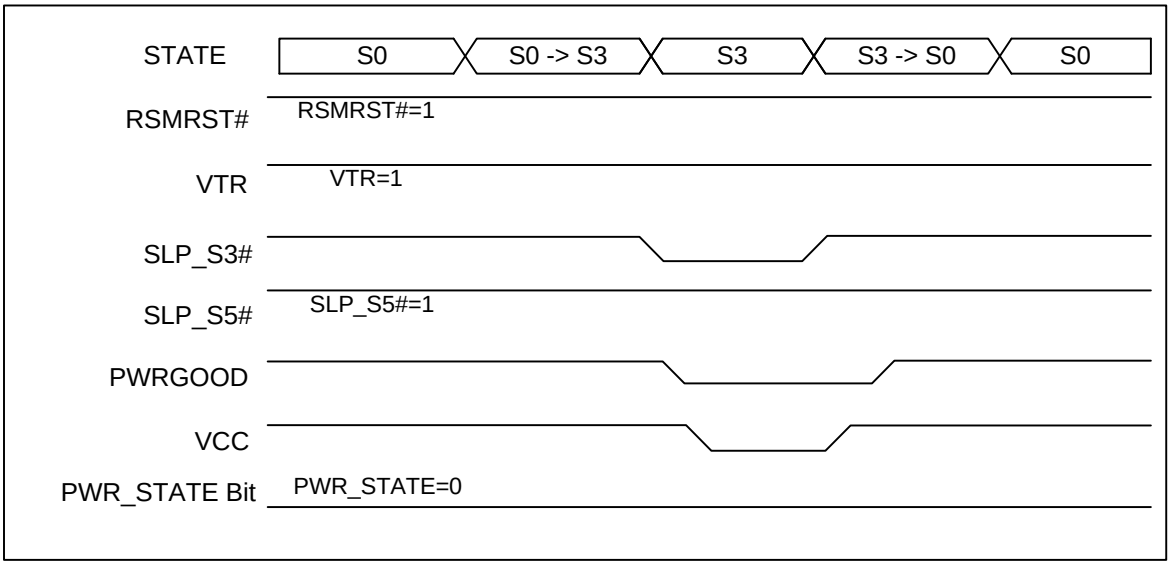


FIGURE 36-28: S0 POWER FAILURE

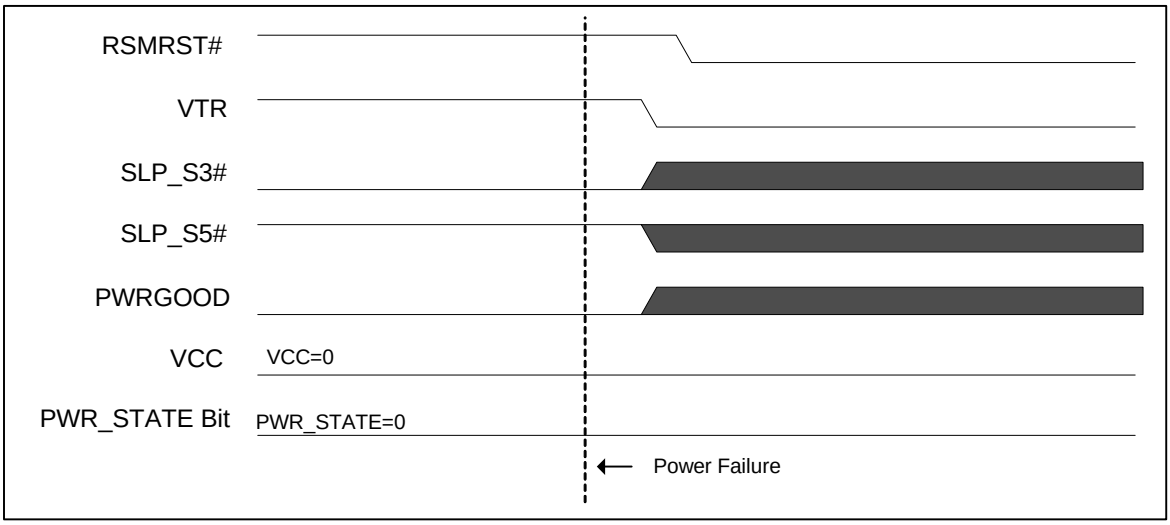


FIGURE 36-29: S3 POWER FAILURE

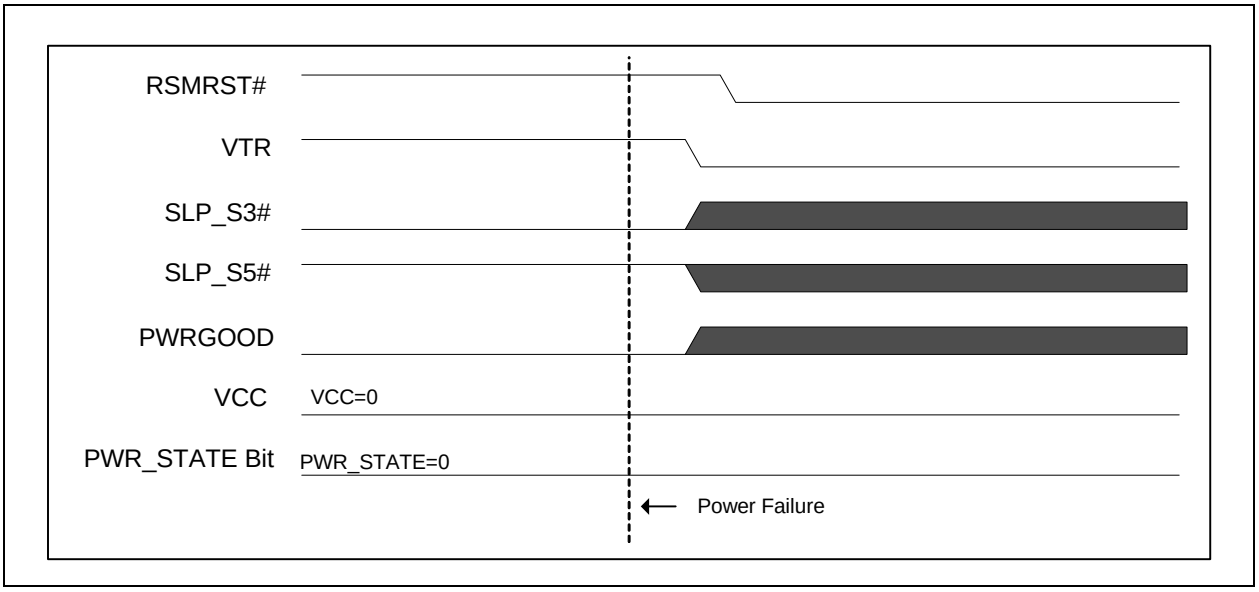


FIGURE 36-30: G3 (POWER FAILURE) TO S0

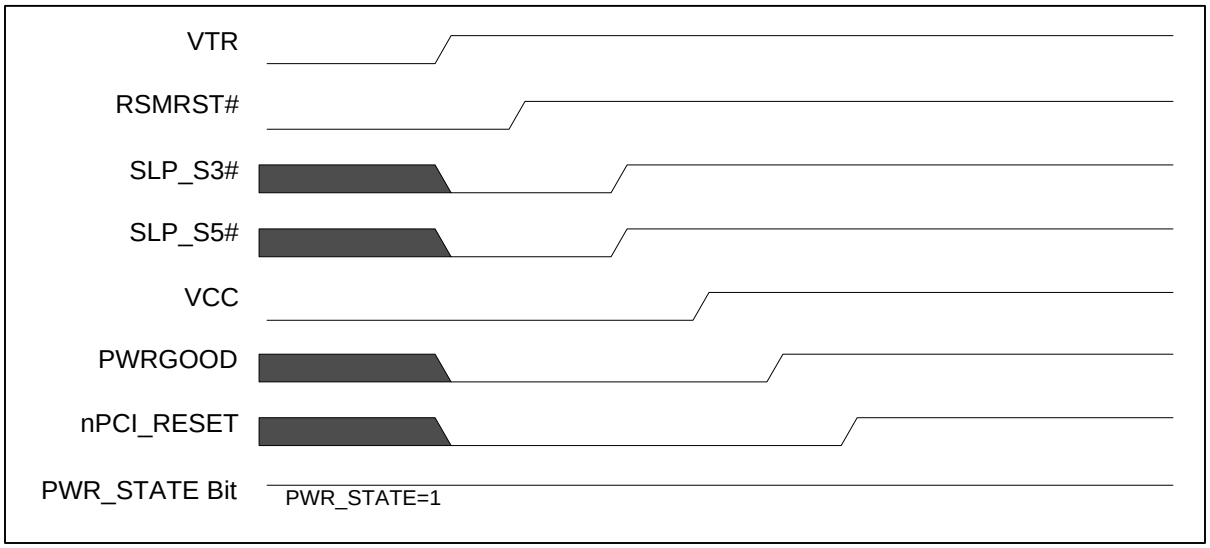
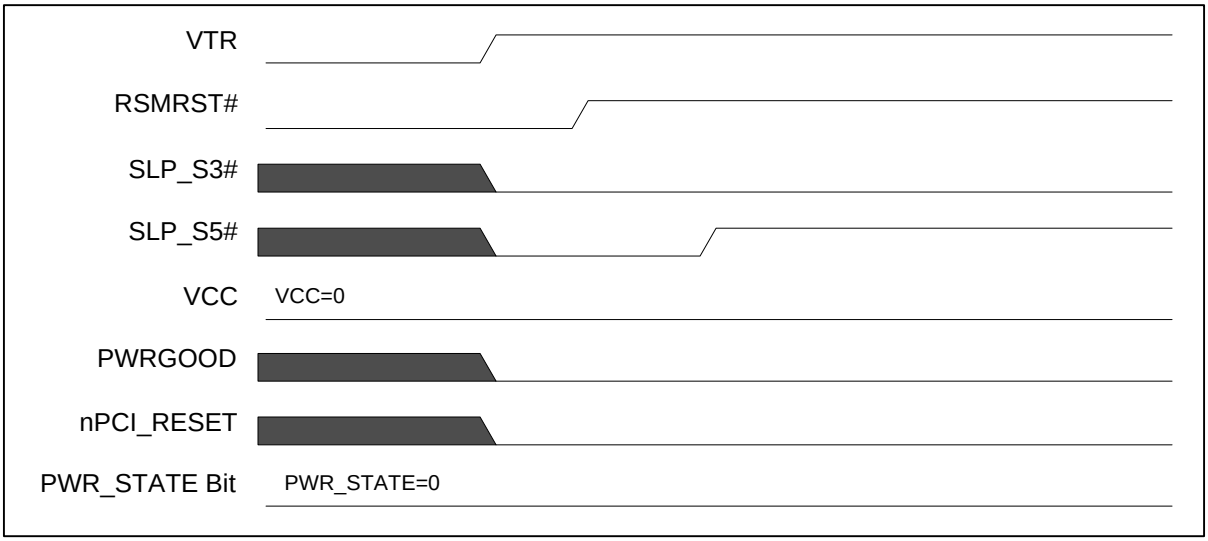


FIGURE 36-31: G3 (POWER FAILURE) TO S3



36.15 JTAG Interface Timing

FIGURE 36-32: JTAG POWER-UP & ASYNCHRONOUS RESET TIMING

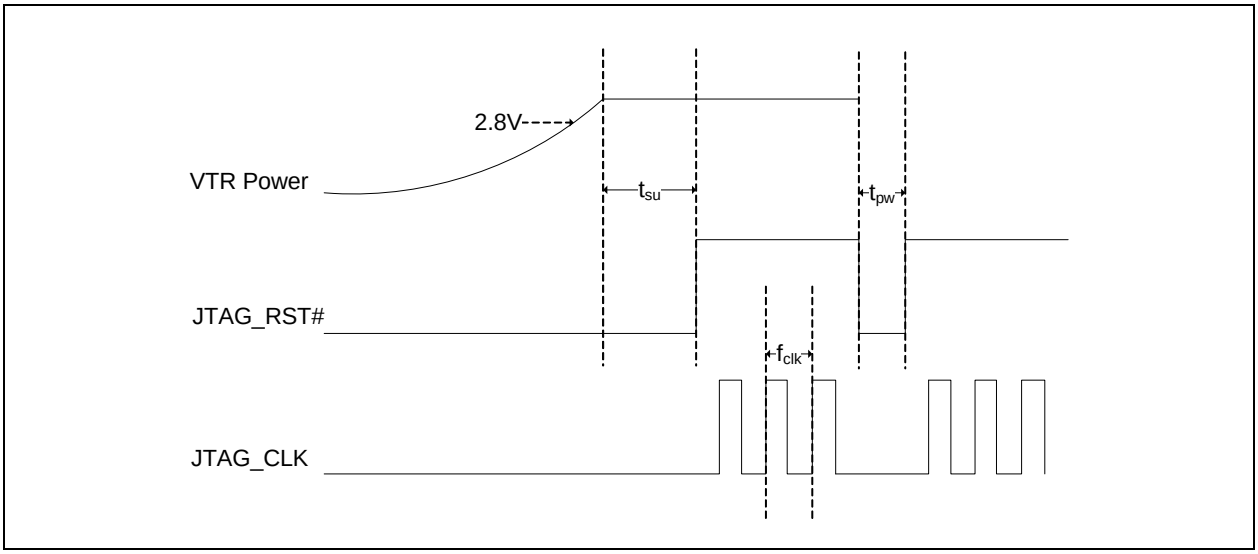


FIGURE 36-33: JTAG SETUP & HOLD PARAMETERS

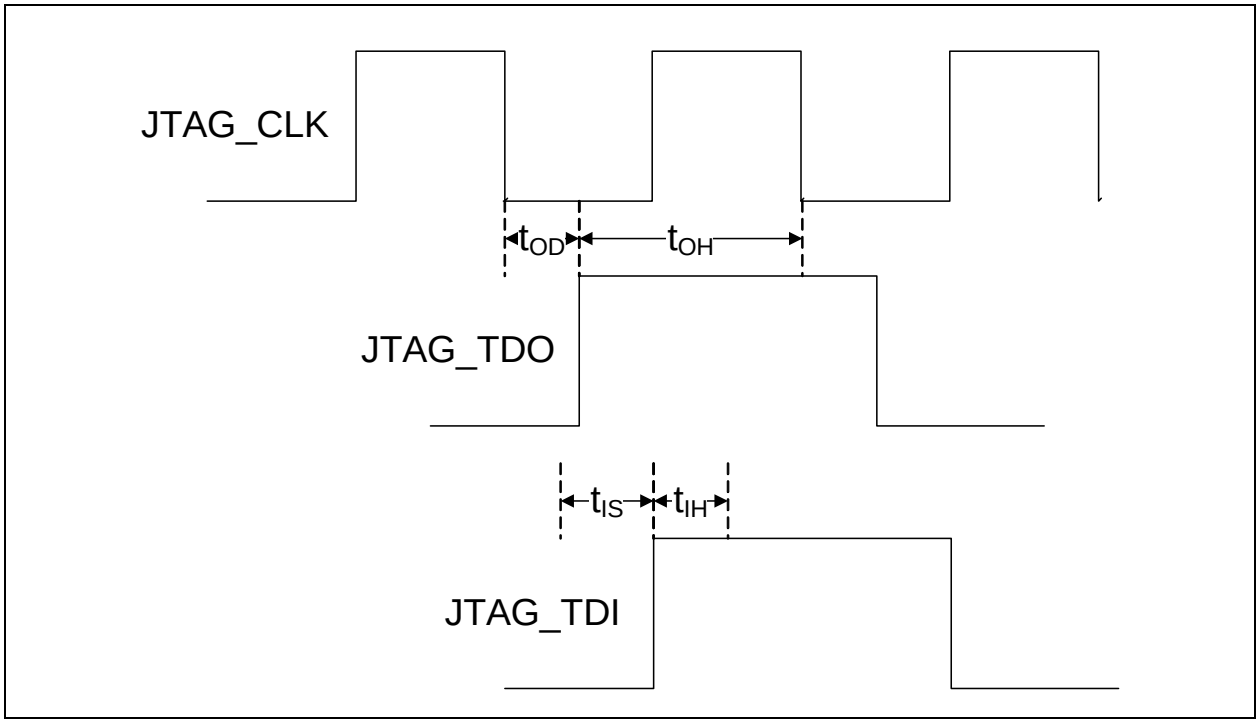


TABLE 36-22: JTAG INTERFACE TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t_{su}	JTAG_RST# de-assertion after VTR power is applied	500			µs
t_{pw}	JTAG_RST# assertion pulse width	500			nsec
f_{clk}	JTAG_CLK frequency (see note)			8	MHz
t_{OD}	TDO output delay after falling edge of TCLK.	5		10	nsec
t_{OH}	TDO hold time after falling edge of TCLK	1 TCLK - t_{OD}			nsec
t_{IS}	TDI setup time before rising edge of TCLK.	5			nsec
t_{IH}	TDI hold time after rising edge of TCLK.	5			nsec

Note: f_{clk} is the maximum frequency to access a JTAG Register. Additional JTAG_CLK frequency constraints are described in Section 33.4.2, "Clocks," on page 313.

36.16 Serial Debug Port Timing

FIGURE 36-34: SERIAL DEBUG PORT TIMING PARAMETERS

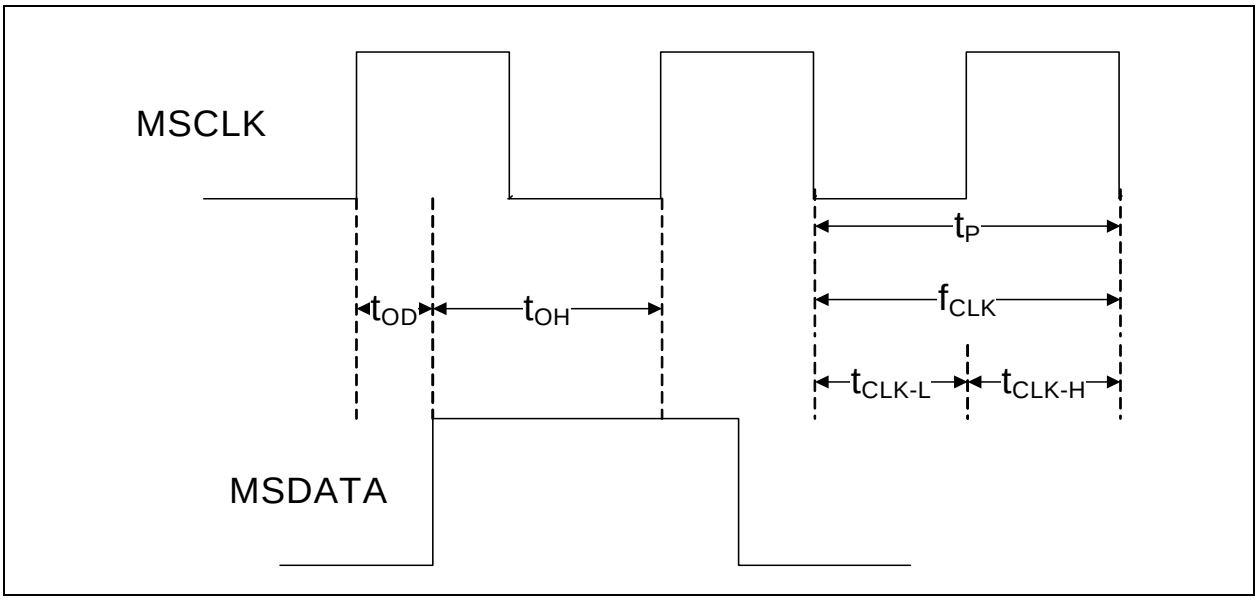


TABLE 36-23: SERIAL DEBUG PORT INTERFACE TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
f_{clk}	MSCLK frequency (see note)	8	-	32	MHz
t_{OD}	MSDATA output delay after falling edge of MSCLK.			5	nsec
t_{OH}	MSDATA hold time after falling edge of TCLK	1 MSCLK - t_{OD}			nsec
t_P	MSCLK Period.	1/fclk			µs
t_{CLK-L}	MSCLK Low Time	$t_P/2$ - 3		$t_P/2$ + 3	nsec
t_{CLK-H}	MSCLK high Time (see Note 36-9)	$t_P/2$ - 3		$t_P/2$ + 3	nsec

Note 36-9 When the EC_CLK_DIV is an odd number value greater than 2h, then $t_{CLK-L} = t_{CLK-H} + 15ns$. When the EC_CLK_DIV is 0h, 1h, or an even number value greater than 2h, then $t_{CLK-L} = t_{CLK-H}$.

36.17 Input Clock Timing

FIGURE 36-35: INPUT CLOCK TIMING

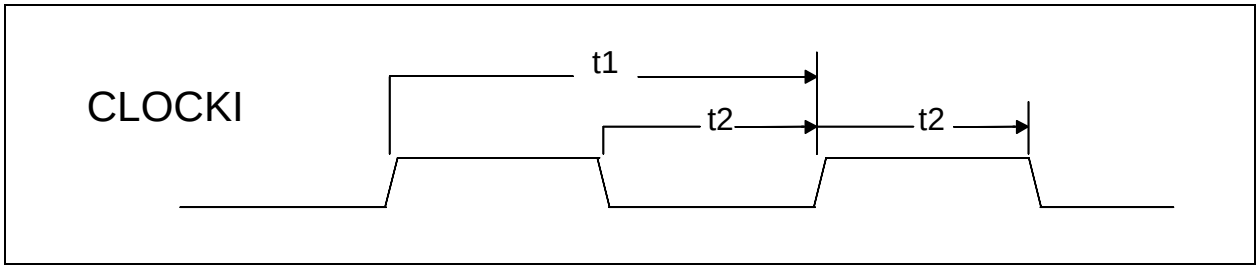
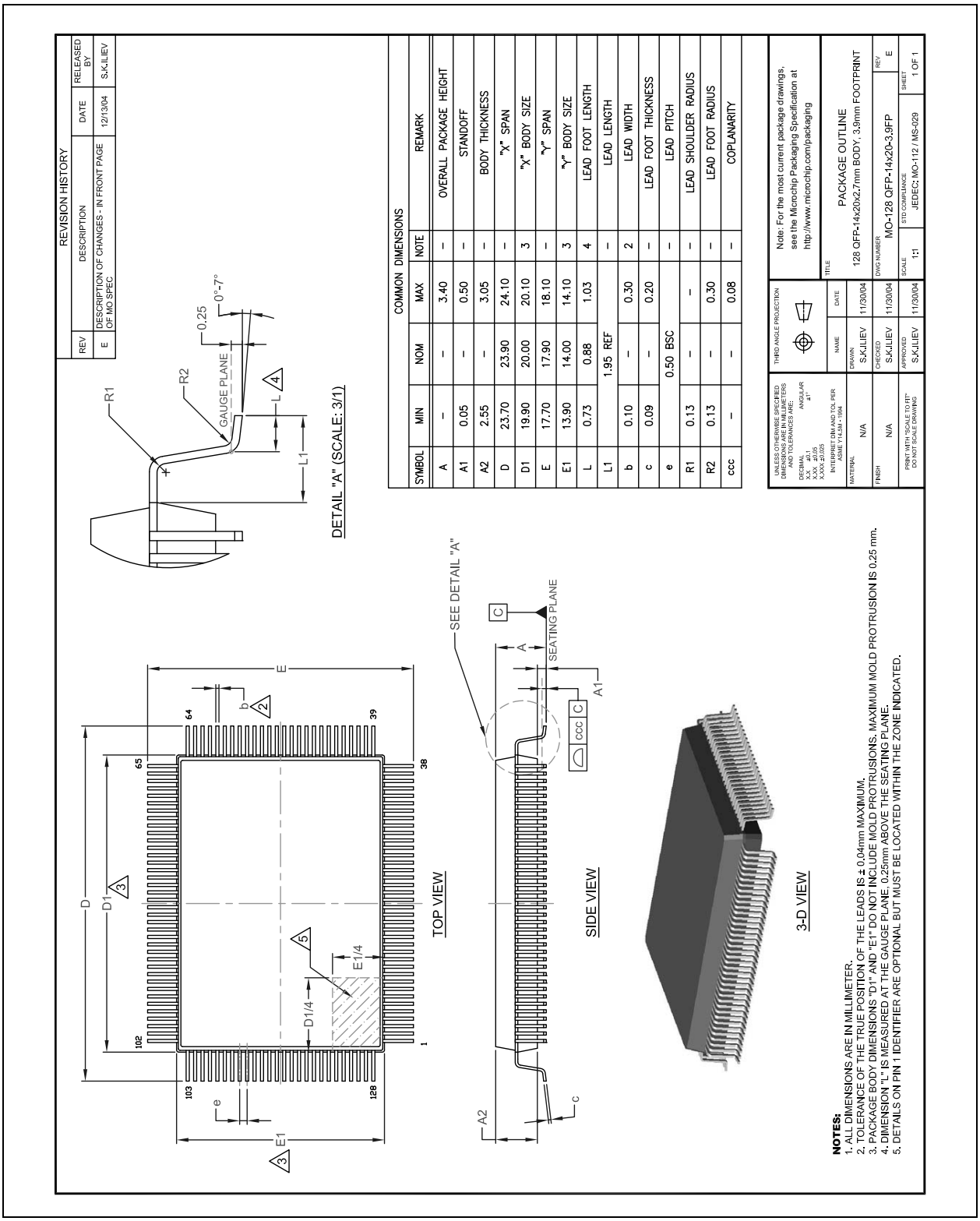


TABLE 36-24: INPUT CLOCK TIMING PARAMETERS

Name	Description	MIN	TYP	MAX	Units
t1	Clock Cycle Time for 14.318MHZ		69.84		ns
t2	Clock High Time/Low Time for 14.318MHz	20	35		ns
	Clock Rise Time/Fall Time (not shown)			5	ns

37.0 PACKAGE OUTLINE

FIGURE 37-1: 128-PIN QFP PACKAGE OUTLINE (3.9MM FOOTPRINT)



38.0 REFERENCE DOCUMENTS

This document was created using the following parent documents:

1. Intel Low Pin Count Specification, Revision 1.0, September 29, 1997
2. PCI Local Bus Specification, Revision 2.2, December 18, 1998
3. Advanced Configuration and Power Interface Specification, Revision 1.0b, February 2, 1999
4. System Management Bus Specification, Revision 1.1, December 11, 1998.
5. Plug and Play ISA Specification, Version 1.0a, Intel Corp. and Microsoft Corp., May 5, 1994
6. I^2C-BUS Specification, Version 2.1, January 2000.
7. SMBus Controller Core Interface, Revision 2.0, Core-Level Architecture Specification, SMSC, 6/18/08
8. ECE1077 MEC-04 Keyboard Scan Extension, Product Architecture Specification, Rev 0.23, January 12, 2006, SMSC Confidential
9. Intel® 82801DBM I/O Controller Hub 4 Mobile (ICH4-M), Data Sheet, Order Number: 252337-001, Intel Corp., January 2003
10. SMSC BC-Link™ Specification, Revision1.02, dated September 05, 2007
11. AN #16.1, MEC5035 JTAG Design and Layout Guide
12. MEC5035 Product Architecture Specification, Rev. 0.64, 01-04-2008.
13. IEEE Std 1149.1
14. PECI Interface Core, Rev. 1.2, Core-Level Architecture Specification, SMSC Confidential, 8/1/08.

APPENDIX A: DATA SHEET REVISION HISTORY

TABLE A-1: REVISION HISTORY

REVISION LEVEL & DATE	SECTION/FIGURE/ENTRY	CORRECTION
DS-00001984A (08-12-15)	Replaces previous SMSC version Rev. 0.30 (01-25-10).	
Rev. 0.30 (01-25-10)	Initial data sheet creation	

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO. [1] - XXX [2]	Example:
Device / Package	SCH5636-NS = 128-pin QFP
Device: SCH5636 [1]	**Note 1:** These products meet the halogen maximum concentration values per IEC61249-2-21.
Package: NS = 128-pin QFP [2]	**Note 2:** All package options are RoHS compliant. For RoHS compliance and environmental information, please visit http://www.microchip.com/pagehandler/en-us/aboutus/ehs.html .

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

ISBN: 9781632776532

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110

Canada - Toronto
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Dusseldorf
Tel: 49-2129-3766400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Venice
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Poland - Warsaw
Tel: 48-22-3325737

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

07/14/15

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Microchip:
SCH5636-NS