SiI3132 PCI Express to Serial ATA Controller

Data Sheet

Document # SiI-DS-0138-E

May, 2010

Further Information

To request other materials, documentation, and information, contact your local Silicon Image, Inc. sales office or visit the Silicon Image, Inc. web site at www.siliconimage.com.

Revision History

Revision	Date	Comment
A	4/8/2005	Derived from preliminary datasheet rev 0.3
A01	8/15/2005	Updated register description for BAR0 Offset 50_H
A02	8/11/2006	Corrected inconsistent sentences (minor fixes including mistyping)
C	2/2/2007	Removed confidential markings (no longer under NDA); updated Marking Specification.
D	2/23/2007	Changes to package drawing. New formatting applied.
E	5/7/2010	Added I^2C section; rewrote Initialization Sequence section; copyedited and brought to current standards.

Table of Contents

List of Figures

List of Tables

Overview

The Silicon Image SiI3132 controller is a two-port PCI Express to Serial ATA controller. It provides multiple-port Serial ATA connectivity with minimal host overhead and host-to-device latency. The controller supports a 1-lane 2.5 Gbit/s PCI Express bus and the Serial ATA Generation 2 transfer rate of 3.0 Gbit/s (300 MByte/s).

Features

Overall Features

- Host Protocol:
  - Optimized for transaction oriented designs with minimal Host overhead
  - Supports two command issuance mechanisms:
    - Efficient in both embedded and PC implementations
    - Reduces dependency on bridge behavior
- Supports up to 4-Mbit external flash for BIOS expansion
- Supports a master/slave I^2C interface
- Supports external flash or serial EEPROM for programmable subsystem vendor ID / subsystem product ID
- Fabricated in a 0.18 µ CMOS process with a 1.8 volt core and 3.3 volt I/Os
- Available in an 88-pin QFN package (10x10 mm, 0.4 mm lead pitch) with an ePad **An EPAD must be soldered to PCB GND**
- JTAG boundary scan

PCI Express Features

- Supports 1-lane 2.5 Gbit/s PCI Express
- Internal application interface multiplexed to 2 ports
- All registers appear in unified memory space
- All registers accessible through I/O space
- Full-chip command completion status accessible with single PCI Express access

Serial ATA Features

- Integrated Serial ATA Link and PHY logic
- Compliant with Serial ATA 1.0 specifications
- Supports Serial ATA Generation 2 transfer rate of 3.0 Gbit/s
- Plesiochronous, Single PLL architecture, 1 PLL for 2 ports
- Output Swing Control
- Supports two independent Serial ATA channels:
  - Independent Link, Transport, and data FIFO
  - Independent command fetch, scatter/gather, and command execution with hard-coded state machines — no code space or download
  - Supports Legacy Command Queuing (LCQ)
  - Supports Native Command Queuing (NCQ)
  - Supports Non-zero offsets NCQ
  - Supports Out of order data delivery NCQ
  - Supports FIS-based switching with Port Multipliers
- 31 Commands and Scatter/Gather Tables per port on-chip
- Protocol Override per Command
- Staggered Spin-up Control

Pin Diagram

Figure 1 shows the pin assignments of the controller. A description of the pin functions is in the Pin Descriptions section beginning on page 7. The package is a 10 mm x 10 mm 88-pin QFN with an ePad, which ***must*** be soldered to ground.

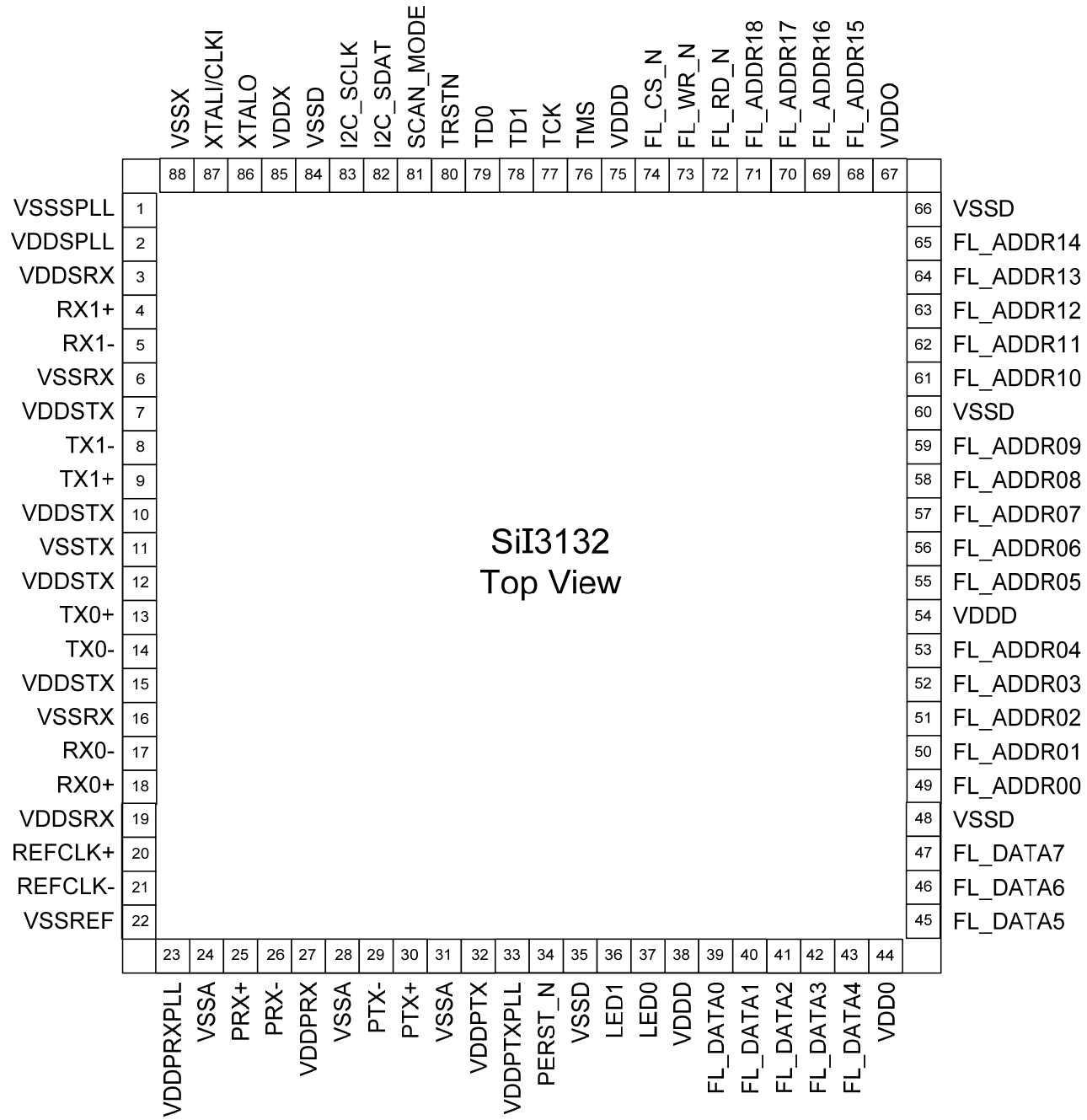


Figure 1. Pin Diagram (Top View)

Electrical Characteristics

Electrical Characteristics

Specifications are for commercial temperature range, 0 °C to +70 °C, unless otherwise specified.

Table 1. Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit
VDDO	I/O Supply Voltage	—	4.0	V
VDDD	Core Supply Voltage	—	2.15	V
VDDSRX, VDDSTX, VDDSPLL, VDDPRX, VDDPTX, VDDPTXPLL, VDDPRXPLL, VDDX	Supply Voltage for SATA and PCI Express Receivers, Transmitters, and PLLs, respectively	—	2.15	V
V_{IN}	Input Voltage	–0.3	VDD + 0.3	V
I_{OUT}	DC Output Current	—	16	mA
θ_{JA}	Thermal Resistance, Junction to Ambient, Still Air	—	22.2*	°C/W
T_{STG}	Storage Temperature	–65	150	°C

Note:** The ePad ***must be soldered to the PCB ground.

Table 2. DC Specifications

Symbol	Parameter	Condition	Type	Min	Typ	Max	Unit
VDDD	Core Supply Voltage	—	—	1.71	1.8	1.89	V
VDDSRX	SATA Receiver Supply Voltage	—	—				
VDDSTX	SATA Transmitter Supply Voltage	—	—				
VDDSPLL	SATA SerDes PLL Supply Voltage	—	—				
VDDPRX	PCI Exp Receiver Supply Voltage	—	—				
VDDPTX	PCI Exp Transmitter Supply Voltage	—	—				
VDDPTXPLL	PCI Exp Transmitter PLL Supply Voltage	—	—				
VDDPRXPLL	PCI Exp Receiver PLL Supply Voltage	—	—				
VDDX	Oscillator Supply Voltage	—	—				
VDDO	Supply Voltage (I/O)	—	—	3.0	3.3	3.6	V
$IDD_{1.8V-3G}$	Supply Current (1.8 V Supply)	3 GHz Operating	—	—	450	570	mA
$IDD_{1.8V-1.5G}$	Supply Current (1.8 V Supply)	1.5 GHz Operating	—	—	380	500	mA
V_{IH}	Input HIGH Voltage	3.3 V I/O	—	2.0	—	—	V
V_{IL}	Input LOW Voltage	3.3 V I/O	—	—	—	0.8	V
V+	Input HIGH Voltage	3.3 V I/O	Schmitt	—	1.8	2.3	V
V-	Input LOW Voltage	3.3 V I/O	Schmitt	0.5	0.9	—	V
V_H	Hysteresis Voltage	3.3 V I/O	Schmitt	0.4	—	—	V
I_{IH}	Input HIGH Current	V_{IN} = VDD	—	–10	—	10	µA
I_{IL}	Input LOW Current	V_{IN} = VSS	—	–10	—	10	µA
V_{OH}	Output HIGH Voltage	—	—	2.4	—	—	V
V_{OL}	Output LOW Voltage	—	—	—	—	0.4	V
I_{OZ}	3-State Leakage Current	—	—	–10	—	10	µA

Note: 3.3 V power consumption depends on LED, JTAG, Enclosure management status. If all are disabled, 3.3 V power consumption is insignificant.

Table 3. SATA Interface DC Specifications

Symbol	Parameter	Condition	Min	Typ	Max	Unit
V_{SATA_DOUT}	TX+/TX- differential peak-to-peak voltage swing.	Terminated by 50 Ω. BAR1 1050h [4:0] = 0x0C*	400	550	700	mV
V_{SATA_DIN}	RX+/RX- differential peak-to-peak input sensitivity	—	240	—	—	mV
V_{SATA_SQ}	RX+/RX- OOB Signal Detection Threshold	—	50	125	240	mV
V_{SATA_ACCM}	Tx AC common-mode voltage	—	—	—	50	mV
Z_{SATA_DIN}	Tx Pair Differential impedance	—	85	100	115	Ω
Z_{SATA_DOUT}	Rx Pair Differential impedance	—	85	100	115	Ω
Z_{SATA_SIN}	Tx Single-Ended impedance	—	40	—	—	Ω
Z_{SATA_SOUT}	Rx Single-Ended impedance	—	40	—	—	Ω

***Note:** 0x0A is a reset value.

Table 4. PCI Express Interface DC Specifications

Symbol	Parameter	Condition	Min	Typ	Max	Unit
V_{PCI_DOUT}	TX+/TX- differential peak-to-peak voltage swing.	Terminated by 50 Ω	800	1000	1200	mV
$V_{PCI\_DE-RATIO}$	Tx De-Emphasized Differential Output Voltage	Ratio	–3.0	–3.5	–4.0	dB
V_{PCI_DIN}	RX+/RX- differential peak-to-peak input sensitivity	—	175	—	1200	mV
Z_{PCI_DIN}	Tx Pair Differential impedance	DC impedance	80	100	120	Ω
Z_{PCI_DOUT}	Rx Pair Differential impedance	DC impedance	80	100	120	Ω
Z_{PCI_SIN}	Tx Single-Ended impedance	DC impedance	40	50	60	Ω
Z_{PCI_SOUT}	Rx Single-Ended impedance	DC impedance	40	50	60	Ω
$Z_{PCI\_RX-HIGH-IMP-DC}$	Rx Powered Down Impedance	DC impedance	200	—	—	kΩ
$Z_{PCI\_RX-IDLE-DET-DIFFp-p}$	Electrical Idle Detect Threshold	Measured at the Rx pins	65	—	175	mV

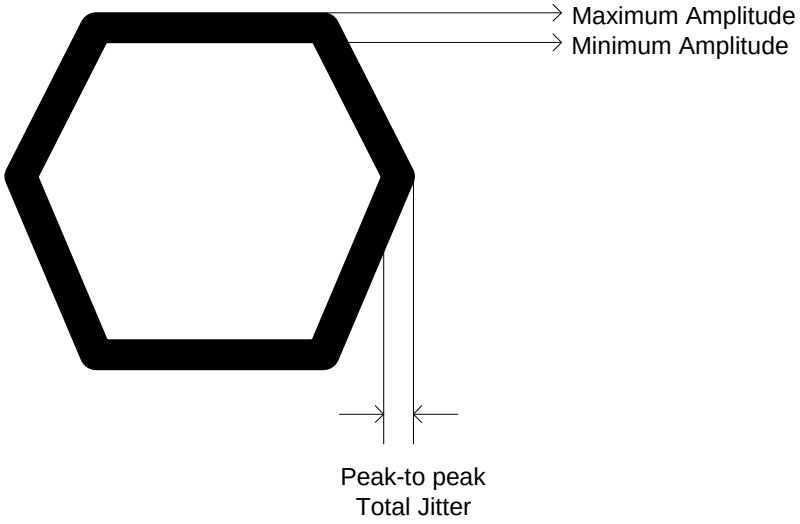


Figure 2. Eye Diagram

SATA Interface Timing Specifications

Table 5. SATA Interface Timing Specifications

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$T_{TX\_RISE\_FALL}$	Rise and Fall time at transmitter	20%–80% at Gen 1 20%–80% at Gen 2	100 67	—	273 136	ps
$T_{TX\_TOL\_FREQ}$	Tx Frequency Long Term Stability	—	–350	—	+350	ppm
$T_{TX\_AC\_FREQ}$	Tx Spread-Spectrum Modulation Deviation	CLKI = SSC AC modulation, subject to the *Downspread SSC* triangular modulation (30–33 kHz) profile per 6.6.4.5 in SATA 1.0 specification	–5000	—	+0	ppm
T_{TX_SKEW}	Tx Differential Skew	—	—	—	15	ps

SATA Interface Transmitter Output Jitter Characteristics

Table 6. SATA Interface Transmitter Output Jitter Characteristics, 1.5 Gbit/s

Symbol	Parameter	Condition	Min	Typ	Max	Unit
TJ_{5UI_15G}	Total Jitter, Data-Data 5 UI	Measured at Tx output pins peak-to-peak phase variation Random data pattern	—	80	—	ps
DJ_{5UI_15G}	Deterministic Jitter, Data-Data 5 UI	Measured at Tx output pins peak-to-peak phase variation Random data pattern	—	40	—	ps
TJ_{250UI_15G}	Total Jitter, Data-Data 250 UI	Measured at Tx output pins peak-to-peak phase variation Random data pattern	—	100	—	ps
DJ_{250UI_15G}	Deterministic Jitter, Data-Data 250 UI	Measured at Tx output pins peak-to-peak phase variation Random data pattern	—	60	—	ps

Table 7. SATA Interface Transmitter Output Jitter Characteristics, 3 Gbit/s

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$TJ_{fBAND/10\_3G}$	Total Jitter, $f_{C3dB} = f_{BAUD}/10$	Measured at SATA Compliance Point Random data pattern Load = LL Laboratory Load	—	60	—	ps
$DJ_{fBAND/10\_3G}$	Deterministic Jitter, $f_{C3dB} = f_{BAUD}/10$	Measured at SATA Compliance Point Random data pattern Load = LL Laboratory Load	—	15	—	ps
$TJ_{fBAND/500\_3G}$	Total Jitter, $f_{C3dB} = f_{BAUD}/500$	Measured at SATA Compliance Point Random data pattern Load = LL Laboratory Load	—	70	—	ps
$DJ_{fBAND/500\_3G}$	Deterministic Jitter, $f_{C3dB} = f_{BAUD}/500$	Measured at SATA Compliance Point Random data pattern Load = LL Laboratory Load	—	20	—	ps

PCI Express Interface Timing Specifications

Table 8. PCI Express Interface Timing Specifications

Symbol	Parameter	Condition	Min	Typ	Max	Unit
T_{PCI_UI}	Tx / Rx Unit Interval	SSC disabled	399.88	400	400.12	ps
$T_{PCI\_TX\_RISE\_FALL}$	Rise and Fall time at transmitter	20%–80%	0.125	—	—	UI
$T_{PCI\_TX-IDLE-MIN}$	Minimum time spent in Electrical Idle	—	50	—	—	UI
$T_{PCI\_TX-IDLE-SET-TO-IDLE}$	Maximum time to transition to a valid Electrical Idle after sending an Electrical Idle ordered set	—	—	—	20	UI
$T_{PCI\_TX-IDLE-TO-TO-DIFF-DATA}$	Maximum time to transition to valid TX specifications after leaving an Electrical Idle condition	—	—	—	20	UI
T_{PVPERL}	Power stable to PERST# inactive	—	100	—	—	ms
$T_{PERST-CLK}$	REFCLK stable before PERST# inactive	—	100	—	—	us
T_{PERST}	PERST# active time	—	100	—	—	us

PCI Express Interface Transmitter Output Jitter Characteristics

Table 9. PCI Express Interface Transmitter Output Jitter Characteristics

Symbol	Parameter	Condition	Min	Typ	Max	Unit
TJ_{PCIe}	Total Jitter	Defined by PCI Express Base Specification Rev 1.1	—	65	—	ps

CLKI SATA Reference Clock Input Requirement

Table 10. CLKI SerDes Reference Clock Input Requirement

Symbol	Parameter	Condition	Min	Typ	Max	Unit
T_{CLKI_FREQ}	Nominal Frequency	—	—	25	—	MHz
V_{CLKI_IH}	Input HIGH Voltage	—	0.7xV DDX	—	—	V
V_{CLKI_IL}	Input LOW Voltage	—	—	—	0.3xV DDX	V
T_{CLKI_J}	CLKI frequency tolerance	—	–50	—	+50	ppm
$T_{CLKI\_RISE\_FALL}$	Rise and Fall time at CLKI	25 MHz reference	—	—	4	ns
T_{CLKI_RJ}	Random Jitter	Measured at CLKI pin 10^{-12} Bit Error Ratio 1 sigma deviation	—	—	50	psrms
T_{CLKI_TJ}	Total Jitter	Measured at CLKI pin 10^{-12} Bit Error Ratio peak-to-peak phase noise	—	—	1	ns
$T_{CLKI\_RC\_DUTY}$	CLKI duty cycle	20%–80%	40	—	60	%

Power Supply Noise Requirements

Table 11. Power Supply Noise Requirement

Symbol	Parameter	Condition	Min	Typ	Max	Unit
V_{NOISE_VDDA}	1.8 V Analog Power Noise	Peak-to-peak sine wave across 500 kHz to 3 GHz frequency range. Measured with differential probe triggered by noise source	—	—	50	mV
V_{NOISE_VDDD}	1.8 V Digital Power Noise		—	—	100	mV
V_{NOISE_VDDO}	3.3 V IO Power Noise		—	—	200	mV

Pin Descriptions

PCI Express Pins

Name	Pin	Type	Dir	Description
PRX+	25	Differential	Input	Receiver.
PRX-	26			Serial receiver differential pair; must be AC coupled.
PTX+	30	Differential	Output	Transmitter.
PTX-	29			Serial transmitter differential pair; must be AC coupled with a 100 nF capacitor.
REFCLK+	20	Differential	Input	Reference Clock.
REFCLK-	21			Reference clock differential pair supplied by PCI Express system. This input signal must be compliant with PCI Express Card Electromechanical Specification, Revision 1.0a.
PERST_N	34	Schmitt	Input	Reset. PERST_N initializes the PCI Express interface and sets internal registers to their initial state.

Flash Data and Address Pins

Name	Pin	Type	Dir	Description
FL_ADDR18	71	8 mA	Input output	Flash Address. The 18 FL_ADDR*nn* pins are the Flash Memory address pins for up to 512 kB of flash memory. These pins have an internal 70 kΩ pullup resistor.
FL_ADDR17	70			
FL_ADDR16	69			
FL_ADDR15	68			
FL_ADDR14	65			
FL_ADDR13	64			
FL_ADDR12	63			
FL_ADDR11	62			
FL_ADDR10	61			
FL_ADDR09	59			
FL_ADDR08	58			
FL_ADDR07	57			
FL_ADDR06	56			
FL_ADDR05	55			
FL_ADDR04	53			
FL_ADDR03	52			
FL_ADDR02	51			
FL_ADDR01	50			
FL_ADDR00	49			
FL_DATA7	47	8 mA	Input output	Flash Data. The 8 FL_DATA*n* pins comprise the 8-bit flash memory data bus. These pins have an internal 70 kΩ pullup resistor.
FL_DATA6	46			
FL_DATA5	45			
FL_DATA4	43			
FL_DATA3	42			
FL_DATA2	41			
FL_DATA1	40			
FL_DATA0	39			

FLSAH Control, I²C, and LED Pins

Name	Pin	Type	Dir	Description
FL_RD_N	72	8 mA	Input Output	Flash Read Enable. Active LOW. This pin has an internal 70 kΩ pullup resistor.
FL_WR_N	73	8 mA	Input Output	Flash Write Enable. Active LOW. This pin has an internal 70 kΩ pullup resistor is required on each of these pins.
FL_CS_N	74	8 mA	Input Output	Flash Chip Select. Active LOW. This pin has an internal 70 kΩ pullup resistor.
I2C_SDAT	82	Schmitt 4 mA	Input Output	I²C Serial Data. Serial Interface (I²C) data line (internally connected to PHY I²C data line). This pin has an internal 70 kΩ pullup resistor.
I2C_SCLK	83	Schmitt 4 mA	Input Output	I²C Serial Clock. Serial Interface (I²C) clock (internally connected to PHY I²C clock). This pin has an internal 70 kΩ pullup resistor.
LED1	36	Open drain 12 mA	Output	Activity LEDs.
LED0	37			Activity LED drivers for channels 1 and 0.

Serial ATA Pins

Name	Pin	Type	Dir	Description
RX1+	4	Differential	Input	Receiver. Serial receiver differential pair; must be AC coupled.
RX1-	5			
RX0+	18			
RX0-	17			
TX1+	9	Differential	Output	Transmitter. Serial transmitter differential pair; must be AC coupled.
TX1-	8			
TX0+	13			
TX0-	14			
XTALI/CLKI	87	Analog	Input	Crystal In. Crystal oscillator pin for SerDes reference clock. When an external clock source is selected, the 25 MHz external clock comes through this pin. The clock must have a 1.8 V swing and the precision recommendation is ±50 ppm. Refer to Table 10 on page 6 for details.
XTALO	86	Analog	Output	Crystal Out. Crystal oscillator pin for SerDes reference clock. A 25 MHz crystal must be used.

Test Pins

Name	Pin	Type	Dir	Description
TMS	76	LVTTL	Input	JTAG Test Mode Select. A 70 kΩ pullup resistor is required on this pin.
TCK	77	Schmitt	Input	JTAG Test Clock.
TDI	78	LVTTL	Input	JTAG Test Data In. This pin has an internal 70 kΩ pullup resistor is required on this pin.
TDO	79	4 mA	Output	JTAG Test Data Out.
TRSTN	80	LVTTL	Input	JTAG Test Reset. This pin must be tied to ground if JTAG function is not used. This pin has an internal 70 kΩ pullup resistor.
SCAN_MODE	81	LVTTL	Input	Scan Mode. Used for factory testing; do not connect. This pin has an internal 60 kΩ pulldown resistor.

Power/Ground Pins

All like-named power and ground pins shown in this section are connected together within the package.

Name	Pin	Description	Supply
VDDSRX	3, 19	Receiver Power. These pins provide power for the Serial ATA receivers.	1.8 V
VSSRX	6, 16	Receiver Ground. These pins provide the ground reference for the Serial ATA receivers.	Ground
VDDSTX	7, 10, 12, 15	Transmitter Power. These pins provide power for the Serial ATA transmitters.	1.8 V
VSSTX	11	Transmitter Ground. This pin provides the ground reference for the Serial ATA transmitters.	Ground
VDDPRX	27	Receiver Power. This pin provides power for the PCI Express receivers.	1.8 V
VDDPTX	32	Transmitter Power. This pin provides power for the PCI Express transmitters.	1.8 V
VDDPTXPLL	33	PLL Power. This pin provides power for the PCI Express transmitter PLL.	1.8 V
VDDPRXPLL	23	PLL Power. This pin provides power for the PCI Express receiver PLL.	1.8 V
VSSA	24, 28, 31	PCI-E Ground. These pins provide the ground reference for the PCI Express SerDes.	Ground
VDDSPLL	2	PLL Power. This pin provides power for the Serial ATA PLL and crystal oscillator.	1.8 V
VSSSPLL	1	PLL Ground. This pin provides the ground reference for the Serial ATA PLL.	Ground
VSSREF	22	Reference Clock Ground. This pin provides the Ground reference for the PCI-Express reference clock receiver.	Ground
VDDX	85	Oscillator Power. This pin provides power for the crystal oscillator associated with the XTALI and XTALO pins.	1.8 V
VSSX	88	Oscillator Ground. This pin provides the ground reference for the crystal oscillator.	Ground
VDDO	44, 67	I/O Power. These pins provide power for the digital I/O.	3.3 V
VDDD	38, 54, 75	Digital Power. These pins provide power for the digital logic.	1.8 V
VSSD	35, 48, 60, 66, 84	Digital Ground. These pins provide the ground reference for the digital portion of the chip.	Ground

Pin List by Pin Number

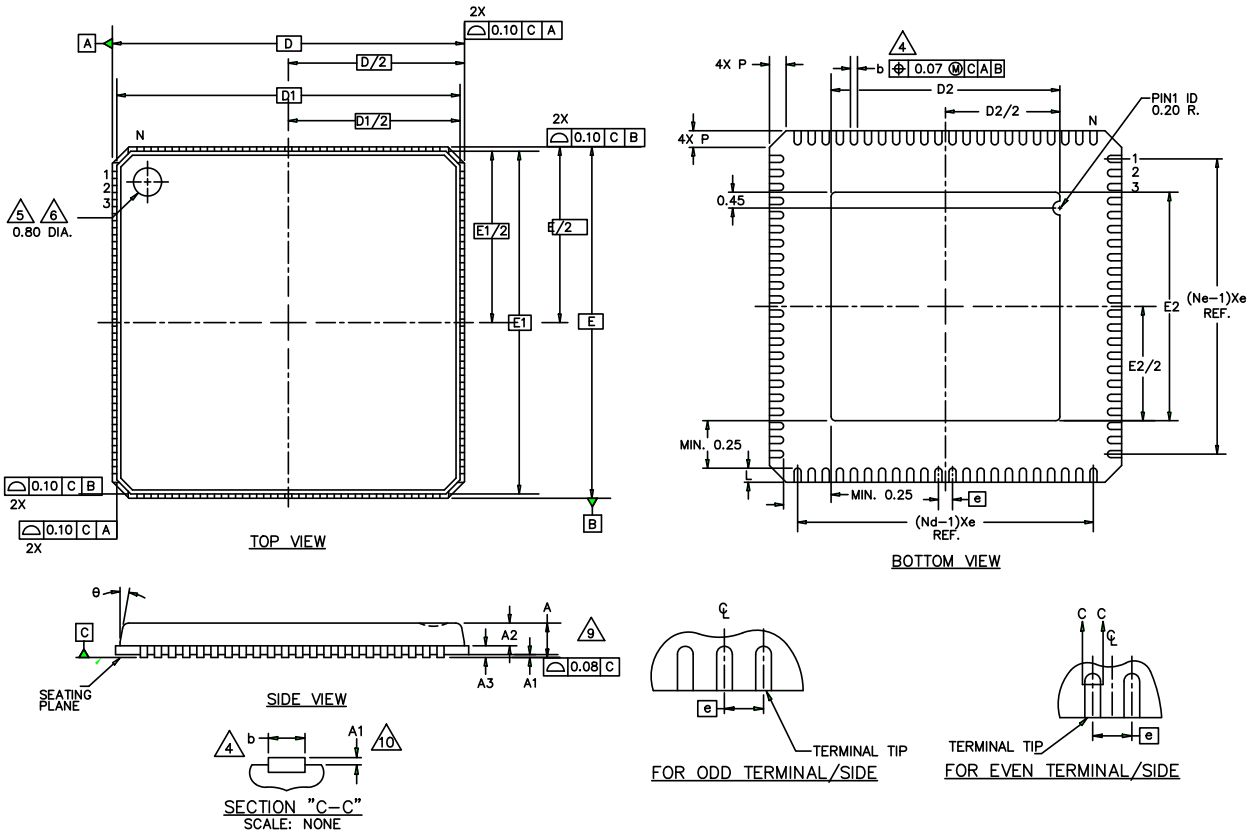
See tables above for complete descriptions and more information.

Pin	Name	Description
1	VSSSPLL	PLL ground (Serial ATA)
2	VDDSPLL	PLL power (Serial ATA)
3	VDDSRX	Receiver power (Serial ATA)
4	RX1+	Serial Receiver differential pair (Channel 1)
5	RX1-	
6	VSSRX	Receiver ground (Serial ATA)
7	VDDSTX	Transmitter power (Serial ATA)
8	TX1-	Serial Transmitter differential pair (Channel 1)
9	TX1+	
10	VDDSTX	Transmitter power (Serial ATA)
11	VSSTX	Transmitter ground (Serial ATA)
12	VDDSTX	Transmitter power (Serial ATA)
13	TX0+	Serial Transmitter differential pair (Channel 0)
14	TX0-	
15	VDDSTX	Transmitter power (Serial ATA)
16	VSSRX	Receiver ground (Serial ATA)
17	RX0-	Serial Receiver differential pair (Channel 0)
18	RX0+	
19	VDDSRX	Receiver power (Serial ATA)
20	REFCLK+	Reference Clock differential pair (PCI Express)
21	REFCLK-	
22	VSSREF	Reference Clock ground (PCI-E)
23	VDDPRXPLL	PLL power (PCI Express receiver)
24	VSSA	PCI-E ground
25	PRX+	Serial Receiver differential pair (PCI Express)
26	PRX-	
27	VDDPRX	Receiver power (PCI Express)
28	VSSA	PCI-E ground
29	PTX-	Serial Transmitter differential pair (PCI Express)
30	PTX+	
31	VSSA	PCI-E ground
32	VDDPTX	Transmitter power (PCI Express)
33	VDDPTXPLL	PLL power (PCI Express transmitter)
34	PERST_N	Reset
35	VSSD	Digital Ground
36	LED1	Activity LED (channel 1)
37	LED0	Activity LED (channel 0)
38	VDDD	Digital Power
39	FL_DATA0	Flash Data
40	FL_DATA1	
41	FL_DATA2	
42	FL_DATA3	
43	FL_DATA4	
44	VDDO	I/O Power
45	FL_DATA5	Flash Data
46	FL_DATA6	
47	FL_DATA7	
48	VSSD	Digital Ground
49	FL_ADDR00	Flash Address
50	FL_ADDR01	
51	FL_ADDR02	
52	FL_ADDR03	
53	FL_ADDR04	
54	VDDD	Digital Power
55	FL_ADDR05	Flash Address
56	FL_ADDR06	
57	FL_ADDR07	
58	FL_ADDR08	
59	FL_ADDR09	
60	VSSD	Digital Ground
61	FL_ADDR10	Flash Address
62	FL_ADDR11	
63	FL_ADDR12	
64	FL_ADDR13	
65	FL_ADDR14	
66	VSSD	Digital Ground
67	VDDO	I/O Power
68	FL_ADDR15	Flash Address
69	FL_ADDR16	
70	FL_ADDR17	
71	FL_ADDR18	
72	FL_RD_N	Flash Read Enable
73	FL_WR_N	Flash Write Enable
74	FL_CS_N	Flash Chip Select
75	VDDD	Digital Power
76	TMS	JTAG Test Mode select
77	TCK	JTAG Test Clock
78	TDI	JTAG Test Data In
79	TDO	JTAG Test Data Out
80	TRSTN	JTAG Test Reset
81	SCAN_MODE	Scan Mode
82	I2C_SDAT	I^2C Serial Data
83	I2C_SCLK	I^2C Serial Clock
84	VSSD	Digital Ground
85	VDDX	Oscillator Power
86	XTALO	Crystal Out
87	XTALI/CLKI	Crystal In
88	VSSX	Oscillator Ground
ePad	ePad	Ground

Note: The ePad ***must*** be connected to ground.

Package Drawing

An ePad ***must*** be soldered to PCB ground and the landing area must be incorporated on the PCB within the footprint of the package corresponding to the ePad. The size of this landing area should be at least the maximum size of the Exposed Pad of the package (6.65 x 6.65 mm), but can be larger. If the traces are within the maximum size of the pad, they may short to the pad when the package has an Exposed Pad with the maximum dimension.



Symbol	Dimensions (mm)		
	Min	Typ	Max
e		0.40	
L	0.30	0.40	0.50
b	0.15	0.20	0.25
D2	5.85	6.00[1]	6.65
E2	5.85	6.00[1]	6.65
A	-	0.85	0.90
A1	0.00	0.02	0.05
A2	-	0.65	0.70

Symbol	Dimensions (mm)		
	Min	Typ	Max
A3		0.20 REF	
D		10.00 BSC	
D1		9.75 BSC	
E		10.00 BSC	
E1		9.75 BSC	
θ			12°
P	0.24	0.42	0.60

Figure 3. Package Drawing 88 QFN

Marking Specification

Marking drawings are not to scale.

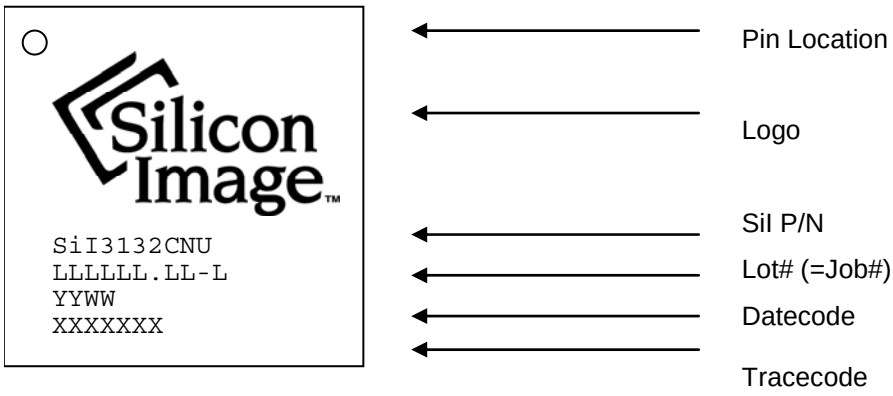


Figure 4. Marking Specification

Ordering Information

Production Part Numbers:

Part	Part Number
88-pin QFN lead free package with an exposed pad	SiI3132CNU

The universal package can be used in both lead-free and ordinary process lines.

Programming Model

SiI3132 Block Diagram

The SiI3132 controller programs the major logic modules illustrated in Figure 5.

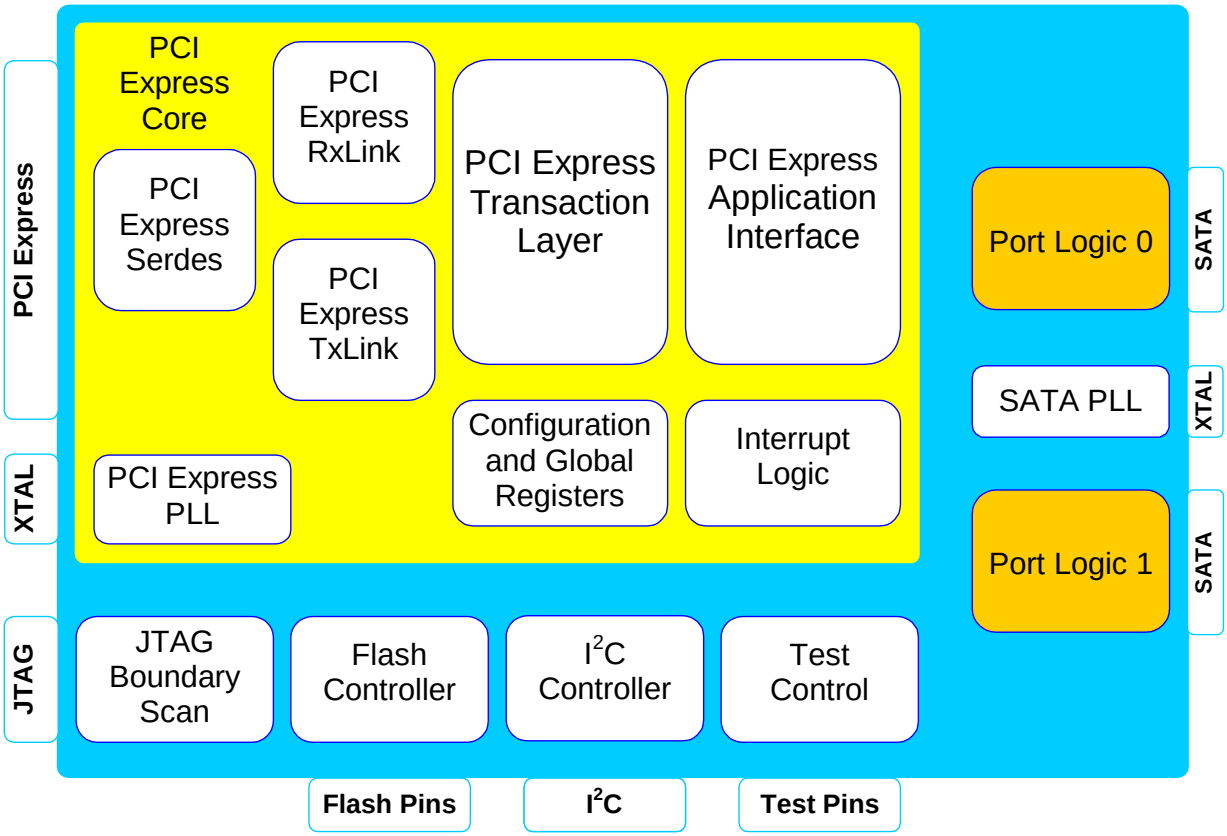


Figure 5. SiI3132 Block Diagram

The PCI Express Core logic block provides PCI Express 1.0a compatibility. The Global Register File block corresponds to the registers addressed by `Base Address Register 0`; refer to PCI Express Capability section on page 44 for more information.

The initialization function provided by the I^2C Controller and Flash Controller is described in the PCI Express Capability section on page 44.

SiI3132 SATA Port Block Diagram

The block diagram illustrated in Figure 6 shows the logic structure of each of the SiI3132 SATA Ports.

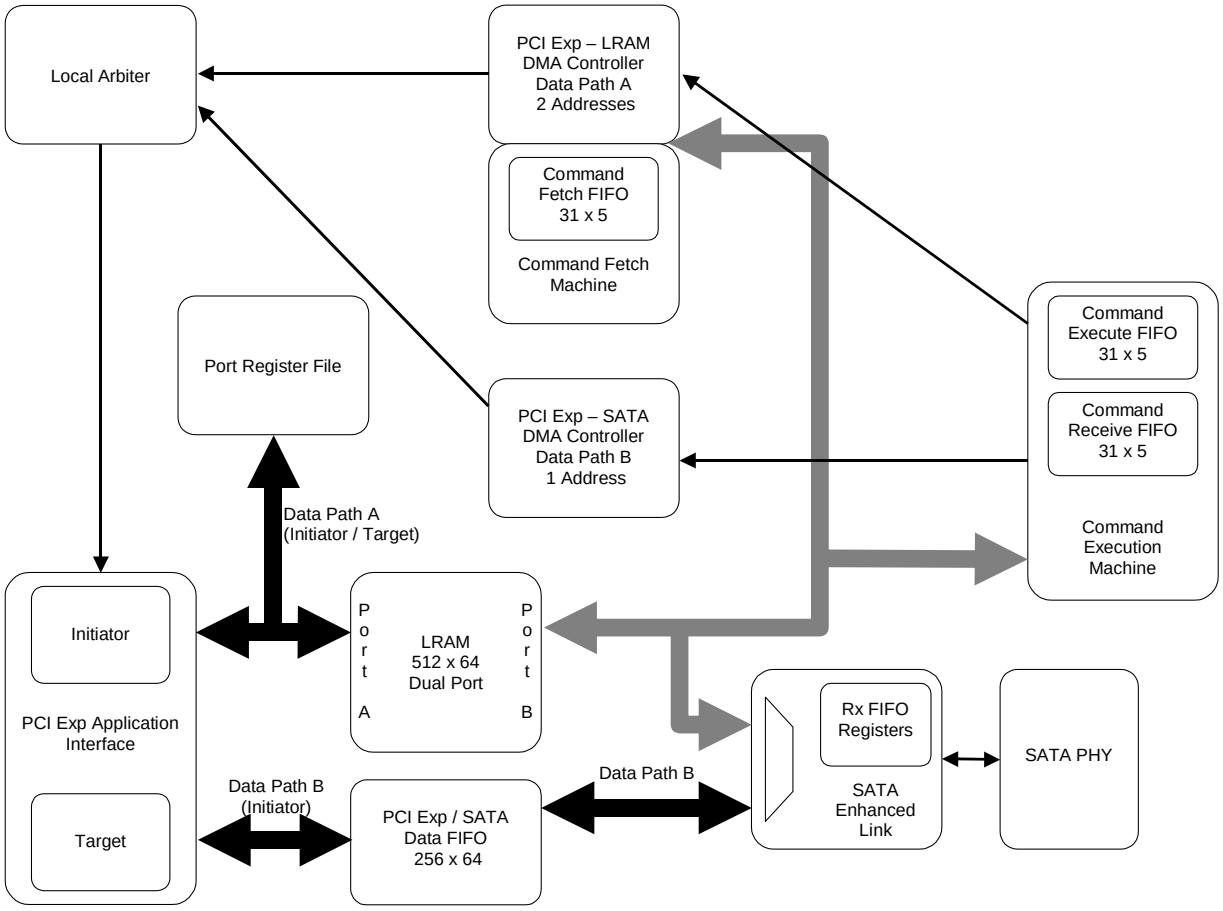


Figure 6. Port Logic Block Diagram

The Port Logic consists of:

- a Local Arbiter that arbitrates between the two DMA Controllers
- a DMA Controller for the PCI Express to LRAM Data Path
- a DMA Controller for the PCI Express to Serial-ATA Data Path
- a 512 x 64 Local RAM (LRAM) that contains: 31 LRAM slots each of which is 128 bytes (16 Qwords) and 128 bytes used to support 16 Port Multiplier devices (1 Qword per device)
- a data FIFO that contains 2048 bytes (256 Qwords)
- a state machine to fetch commands
- a state machine to execute commands
- a serial-ATA Link
- a serial-ATA PHY

Each of the two state machines has an associated FIFO which, when not empty, indicates that processing is required. The FIFO is loaded with a 5-bit command slot number to activate a state machine. The slot number can range from 0 to 30, corresponding to the maximum number of active commands supported.

Command flow begins with a host driver building a command in a non-cached region of host memory. The data structure is referred to as a PRB (Port Request Block). The 64-byte PRB is transferred into an available command slot in the LRAM by one of two methods: the direct method or the indirect method. The host driver is responsible for determining which slots are available. Either of the two command transfer methods may be used for each command transfer. The two methods are described in the following sections.

Direct Command Transfer Method – Host controlled write to Slot

In systems that have the capability to perform burst writes, this is the preferred method of command transfer. Embedded systems would most likely use this method. LRAM is directly mapped through use of `Base Address Register 1` and appears as a block of memory to the host driver. The host driver writes the PRB contents into the appropriate slot in LRAM. Ideally, this operation is performed as a single PCI Express transaction. The 5-bit slot number (0–30) is written to the *Command Execution FIFO*. The Active bit associated with the selected slot becomes set in the `Global Port Slot Status` register. The *Command Fetch FIFO* and *Command Fetch State Machine* are not used for the direct method of command transfer.

Indirect Command Transfer Method – SiI3132 controlled command transfer

The host driver builds a PRB in host memory, selects a free slot, and writes the physical address of the PRB into the Command Activation register corresponding to the selected slot. This causes the SiI3132 controller to push the 5-bit slot number (0–30) into the *Command Fetch FIFO.*

The *Command Fetch State Machine,* while in an idle state, continuously interrogates the *Command Fetch FIFO* for a non-empty condition. Upon retrieval of a 5-bit slot number from the FIFO, the *Command Fetch State Machine* retrieves the physical address of the PRB from the corresponding Command Activation register, sets the Active bit associated with the selected slot in the `Port Slot Status` register, and queues a PCI Express read of the PRB into the associated Slot in LRAM. The *Command Fetch State Machine* waits for completion of the transfer, pushes the 5-bit slot number into the *Command Execution FIFO,* and returns to the idle state, waiting for a non-empty condition in the *Command Fetch FIFO*.

The *Command Execution State Machine* is responsible for directing the flow of the command and response FISs between the command slot and the Serial ATA link, directing the flow of data between PCI Express and the Serial ATA link, and posting completion status to the host. It is also responsible for error handling when exceptions occur in the normal command flow.

Command execution begins when the idle *Command Execution State Machine* recognizes that the Serial ATA bus is in a non-busy state and the *Command Execution FIFO* is non-empty. The *Command Execution State Machine* retrieves the 5-bit slot number (0-30) from the *Command Execution FIFO* and uses it to index the command slot in LRAM. The command FIS is addressed and sent to the Serial ATA link to be sent to the device. Control flags in the command slot determine the type of data transfer. The *Command Execution State Machine* waits for a response FIS from the device and directs its activities accordingly. If the received FIS is a data FIS, the DMA address and count are determined by examining the scatter/gather entries in the PRB and, if necessary, walking a scatter/gather table. The DMA address and count are loaded into the DMA controller and the controller is armed. A DMA activate FIS causes similar behavior, with data flowing from PCI Express to the Serial ATA link. When the command has completed, the Command Completion bit in the `Port Interrupt Status` register is set to reflect the successful completion of the command. If an error occurred, the Command Error bit is set in the Port Interrupt Status register.

The basic command flow proceeds as follows:

1. The host builds a 64-byte Port Request Block (PRB) that contains:
   - The Register- Host to Device FIS to send to the SATA device.
   - Up to two scatter/gather entries to define regions of host memory to be accessed for associated read/write data. Additional scatter/gather entries may be associated with the command.
   - Various optional control flags to direct the SiI3132 controller to perform special processing, to control interrupt assertion, to vary the normal protocol flow, etc.
2. The host issues the command to the SiI3132 controller.
3. The controller executes the command, performing all interaction with the SATA device and transferring data between host memory and the SATA device.
4. The controller asserts a PCI Express interrupt to indicate command completion.
5. The host reads the SiI3132 port slot status to determine which command(s) have completed.

Data Structures

The Command Slot

Each port within the SiI3132 controller contains 31 command slots. The slots are numbered 0 through 30. Each command issued by the host occupies a single command slot. The host decides which slot to use and issues a command to the selected slot. A command slot occupies 128 bytes within the SiI3132 RAM array and consists of a 64-byte PRB (Port Request Block) and a 64-byte scatter/gather table. The host builds the PRB. It contains the Register-Host to Device FIS to transmit to the attached SATA device and up to two scatter/gather entries that define host memory regions to be used for any read/write data associated with the command. If more scatter/gather entries are required to define additional host memory regions, the controller will fetch them from host memory as needed. The host may append the additional SGT (scatter/gather table) entries to the PRB, or one of the scatter/gather entries in the PRB may be used to define an SGT that resides in host memory.

The host may issue commands to any number of available command slots. The host may freely intermix non-queued, legacy queued, native queued, PIO, and DMA command types in any available slot. Commands are always executed in the order that they were issued. The controller enforces command type issuance to the SATA device and does not allow incompatible command types to be issued to a device. This relieves the host of the burden of making sure that incompatible command types are not intermixed in the device.

The host is responsible for managing slot usage. The host must keep track of which slots have commands outstanding and which slots are available for new commands. Issuing a command to a slot that is currently in use will result in unpredictable behavior.

For queued commands, the slot number is used as the queue tag. The host must ensure that the tag number in the Register-Host to Device FIS defined in the PRB matches the slot number to which the command is issued.

The Scatter/Gather Entry (SGE)

A scatter/gather entry (SGE) defines a region of host memory to be used for data transfer associated with a command. Each scatter/gather entry defines a single contiguous physically addressed region.

Table 12. Scatter/Gather Entry (SGE)

31				0	
Data Address Low					0x00
Data Address High					0x04
Data Count					0x08
TRM (31)	LNK (30)	DRD (29)	XCF (28)	Reserved[27:0]	0x0C

The first quadword, at offset 0, contains the physical address of the region in host memory. The entire 64-bit address must be defined. On 32-bit systems the upper 32 bits must be zero. The data address may point to a region to be used for data transfer, or it may point to a Scatter/Gather Table (SGT), which is a collection of four SGEs. The LNK bit (bit 30 at offset 0x0C) defines the type of region. When LNK is zero, the region is a data region; when LNK is one, the region is a scatter/gather table that will be fetched by the SiI3132 controller to obtain a data region definition.

The Data Count field at offset 0x08 defines the length, in bytes, of the contiguous data region. When the LNK bit is set to one, indicating an SGT link, the controller ignores this field.

The TRM bit (bit 31 at offset 0x0C), when set to one, indicates that this is the final SGE associated with the command and no additional SGEs follow it.

The DRD bit (bit 29 at offset 0x0C), when set to one, directs the controller to discard the data read from the device for the length associated with the data count. When this bit is set to one, the controller ignores the data address.

The XCF bit (bit 28 at offset 0x0C) indicates whether the region defined by this SGE is to be used for data transfer (XCF set to *0*) or an external command fetch (XCF set to *1*). See the External Command PRB Structure section on page 22 for additional information on external command processing.

The Scatter/Gather Table (SGT)

The SGT is a contiguous collection of four SGEs. The PRB contains two SGEs. When more than two SGEs are required to fully define the entire data transfer of a command, the SiI3132 controller fetches them in groups of four SGEs at a

time, which is one SGT. The SGT occupies the upper 64 bytes of a command slot in SiI3132 RAM. When needed, only one SGT resides in RAM at a time. The controller fetches each required SGT, overwriting the previous SGT in RAM. Because the first two SGEs reside in the PRB RAM area, they are always available in case the controller needs to rescan the scatter/gather list for out-of-order data delivery.

SGTs must reside on a quadword-aligned boundary in host memory. In other words, bits[2:0] of the physical address of the SGT in host memory must be zero.

Table 13. Scatter/Gather Table (SGT

31				0	
SGE0 Data Address Low					0x00
SGE0 Data Address High					0x04
SGE0 Data Count					0x08
SGE0 TRM	SGE0 LNK	SGE0 DRD	SGE0 XCF	Reserved[27:0]	0x0C
SGE1 Data Address Low					0x10
SGE1 Data Address High					0x14
SGE1 Data Count					0x18
SGE1 TRM	SGE1 LNK	SGE1 DRD	SGE1 XCF	Reserved[27:0]	0x1C
SGE2 Data Address Low					0x20
SGE2 Data Address High					0x24
SGE2 Data Count					0x28
SGE2 TRM	SGE2 LNK	SGE2 DRD	SGE2 XCF	Reserved[27:0]	0x2C
SGE3 Data Address Low					0x30
SGE3 Data Address High					0x34
SGE3 Data Count					0x38
SGE3 TRM	SGE3 LNK	SGE3 DRD	SGE3 XCF	Reserved[27:0]	0x3C

The Port Request Block (PRB)

The host builds a PRB to define a command to be executed by the controller. The PRB occupies the first 64 bytes of a command slot in SiI3132 RAM. Once a command is issued, the PRB is overwritten in the RAM as necessary to keep track of command context and execution status. The host should not depend on being able to read the contents of the PRB in slot RAM after a command is issued. Upon completing command execution the PRB area of the command slot may contain status information that can be read by the host, depending on the command type. The PRB structure can take several forms, depending on the command type that it defines.

The PRB contains the following major elements:

- A Control Field to indicate the type of PRB and any features to execute.
- A Protocol Override field that can alter the normal SATA protocol flow.
- A FIS area that contains the initial FIS to be transmitted to the device upon PRB execution.
- Up to two scatter/gather entries (SGEs) to define areas of host memory that will be used for any data transfer associated with the PRB. For PACKET commands, the first SGE contains the 12- or 16-byte ATAPI command to be transmitted to the device.

Regardless of whether the command is to be issued with the direct or indirect method, the host driver should build the PRB as a structure in host memory. If the command is to be issued using the direct method, the PRB can be copied from host RAM to the appropriate slot in RAM. If the command is to be issued using the indirect method, the host driver should write the physical address of the PRB to the Command Activation register associated with the desired command slot.

The PRB must reside on a quadword-aligned boundary in host memory. In other words, bits[2:0] of the physical address of the PRB in host memory must be zero.

The PRB can take various forms, depending on the type of command being issued. The command types are:

Standard ATA Commands. This includes all the common ATA commands such as READ SECTORS, WRITE SECTORS, READ DMA, WRITE DMA, IDENTIFY DEVICE, SMART, etc. Also included are the queued

commands in both legacy and SATA native queue modes. For these commands, the PRB contains the entire Register – Host to Device FIS containing the ATA command. By default, the controller decodes the ATA command type and executes the necessary SATA protocol automatically. As an option, the host driver can execute any desired SATA protocol on a per-command basis.

PACKET Commands. ATAPI PACKET commands operate in a similar fashion to the standard ATA commands. The Register – Host to Device FIS contains the ATA PACKET command. The 12- or 16-byte ATAPI command is placed in the area normally reserved for the first SGE. The SiI3132 controller does not decode the contents of the 12- or 16-byte ATAPI command, so the host driver indicates the direction of any data transfer associated with the command.

Soft Reset. A special form of the PRB instructs the controller to transmit a soft-reset sequence to a device. The controller creates the necessary Register–Host to Device FISs required for the sequence. No SGEs are required for this PRB type. Other than the control field, the only item that needs to be populated is the PMP field, to direct the soft reset sequence to the proper device in the event that a port multiplier is attached. Upon successful command completion, the Register–Device to Host FIS is available in the command slot, allowing the host driver to view the device signature.

External Command. The external command feature allows the host driver to transmit any arbitrary FIS that will not fit in the FIS area of the PRB. This feature is useful in custom applications that have a need to send large FISs or Data FISs in a fashion that does not comply with the defined SATA protocol.

Interlocked FIS Reception. The interlocked FIS feature allows the host driver to receive any desired FIS type directly to a host memory buffer, bypassing all SATA protocol for that FIS type. To use this feature, the host first specifies the FIS type(s) to be interlocked. Then, any number of available command slots can be reserved for the reception of FISs matching the defined type(s).

The PRB Control Field

The Control Field (offset 0x00, bits [15:0]) is used to indicate the type of PRB and features that are desired. For a standard ATA command, this field will normally contain a default value of 0x0000. Table 14 describes the bit functions for each bit in the Control Field.

Table 14. Control Field Bit Definitions

Bit	Name	Description
0	control_protocol_override	The Protocol Override Field is to be used instead of the default protocol for this command.
1	control_retransmit	Allows retransmission if an error occurs during an external command transmission.
2	control_external_command	The command FIS shall be fetched from host memory. This feature is used to send arbitrary FISs that will not fit in the command FIS area of the PRB.
3	control_receive	Reserves a command slot to be used to receive an interlocked FIS as described by the port FIS_CONFIG register.
4	control_packet_read	Indicates that the packet command associated with this PRB will transfer data from the device to the host. This bit must be set for all packet commands that perform read data transfers.
5	control_packet_write	Indicates that the packet command associated with this PRB will transfer data from the host to the device. This bit must be set for all packet commands that perform write data transfers.
6	control_interrupt_mask	Setting this bit to one will prevent the SiI3132 controller from issuing a normal successful completion interrupt for this command.
7	control_soft_reset	Causes the SiI3132 controller to issue a soft reset FIS sequence to the device.
15:8	reserved	Must be zero

The PRB Protocol Override Field

The Protocol Override Field (offset 0x00, bits [31:16]) is used to specify a protocol behavior other than the default for this PRB. PRBs for which the default protocol is to be used should set this field to 0x0000. The SiI3132 controller will decode the 8-bit ATA command at PRB offset 0x0A and automatically execute the default protocol for the command. In certain cases it might be desirable to specify a non-default protocol to be used, such as with vendor specific device commands. To override the protocol, the host driver must set control_protocol_override (Control Field, bit 0) to *1* and place the desired protocol in this field. Table 15 on the next page describes the Protocol Override bit definitions.

Table 15. Protocol Override Bit Definitions

Bit	Name	Description
16	protocol_packet	This command is to be executed as an ATAPI packet command.
17	protocol_legacy_queue	This command is to be executed as an ATA legacy queued command.
18	protocol_native_queue	This command is to be executed as a SATA native queued command.
19	protocol_read	This command is expected to transfer data from device to host.
20	protocol_write	This command is expected to transfer data from host to device.
21	protocol_transparent	This command is to be executed with no protocol. After the initial command FIS is successfully sent to the device, completion status will be posted without waiting for additional device transmissions.
31:22	Reserved	Must be zero.

There is no distinction between DMA and PIO data transfers in the protocol. The SiI3132 controller is a native Serial ATA device and relies on the SATA interface protocol to determine the data transfer type between the device and the controller. From the host driver perspective, all commands, whether PIO or DMA, transfer data through use of scatter/gather entries defined in the PRB and scatter/gather tables.

Standard ATA Command PRB Structure

Table 16 shows the layout for standard ATA commands. The Control and Protocol override fields must be populated as described above.

Table 16. Port Request Block for Standard ATA Command

31								0	
Protocol Override				Control					0x00
Received Transfer Count (Reserved Input)									0x04
Features / Error	Command / Status			C	R	R	R	PMP	FIS Type	0x08
Dev/Head	Cyl High			Cyl Low				Sector Number		0x0C
Features (Exp)	Cyl High (Exp)			Cyl Low (Exp)				Sector Num (Exp)		0x10
Device Control	Reserved			Sector Count (Exp)				Sector Count		0x14
Reserved										0x18
Reserved – Must Be Zero										0x1C
SGE0 Data Address Low										0x20
SGE0 Data Address High										0x24
SGE0 Data Count										0x28
SGE0 TRM	SGE0 LNK	SGE0 DRD	SGE0 XCF	Reserved[27:0]						0x2C
SGE1 Data Address Low										0x30
SGE1 Data Address High										0x34
SGE1 Data Count										0x38
SGE1 TRM	SGE1 LNK	SGE1 DRD	SGE1 XCF	Reserved[27:0]						0x3C

The Received Transfer Count field (offset 0x04) is reserved as an input to the SiI3132 controller and should be populated with a value of all zeroes. Upon successful command completion, this field will contain the total number of

data bytes received during the command execution. The host driver may use this field to determine the transfer size for commands in which the total transfer size is unknown, such as ATAPI inquiries.

The FIS area (offset 0x08 through 0x1F) must be populated with the initial FIS to be sent to the device. This area contains the FIS header and all task file registers to describe the ATA command. Table 17 describes the FIS area fields.

Table 17. PRB FIS Area Definition

Offset	Bit(s)	Name	Description
0x08	7:0	FIS Type	The FIS type field must be populated with a valid SATA FIS type. In all but special custom cases this value will be 0x27, which defines a Register–Host to Device FIS type.
	11:8	PMP	4-bit Port Multiplier Port field that defines the port to which this command is directed. If no port multiplier is attached, this field should be populated with all zeros.
	14:12	Reserved	Must be zero.
	15	Command/Device Control	This bit must be set to *1* to indicate that this FIS contains a command.
	23:16	Task File Command	Populate with the desired ATA command type.
	31:24	Features	Populate with the desired features for this ATA command.
0x0C	7:0	Sector Number (LBA[7:0])	These fields should be populated with desired command-specific parameters.
	15:8	Cylinder Low (LBA[15:8])	
	23:16	Cylinder High (LBA[23:16])	
	31:24	Device/Head (LBA[27:24] for non-extended commands)	
0x10	7:0	Sector Number (Exp.) (LBA[31:24] for extended commands)	These fields should be populated with desired command-specific parameters.
	15:8	Cylinder Low (Exp.) (LBA[39:32] for extended commands)	
	23:16	Cylinder High (Exp.) (LBA[47:40] for extended commands)	
	31:24	Features (Exp.)	
0x14	7:0	Sector Count	These fields should be populated with desired command-specific parameters. The Reserved field must be 0 for standard ATA commands that use the Register –Host to Device FIS to initiate a command.
	15:8	Sector Count (Exp.)	
	23:16	Reserved	
	31:24	Device Control	
0x18	31:0	Reserved	This field is reserved and must be 0 for standard ATA commands that use the Register – Host to Device FIS to initiate a command.

PACKET Command PRB Structure

Table 18 shows the layout for PACKET commands. The Control and protocol override fields must be populated as described above. The PACKET PRB FIS area is structured the same as a standard ATA command. The FIS area contains the PACKET ATA command. After the initial PACKET command is transmitted, the device responds with a PIO Setup FIS, requesting a 12- or 16-byte ATAPI command. The host driver must populate the area normally used for the first SGE with the desired ATAPI command. The length of the ATAPI command is determined by the value of the packet length bit (Port Control, bit 5). If packet length is 0, 12 bytes are transmitted. If packet length is one, 16 bytes are transmitted. The packet length field must be initialized with the packet length value returned by the device in the IDENTIFY PACKET command. Table 18 on the next page shows a representative 12-byte ATAPI command layout. The highlighted ATAPI packet is an example typical of some commands; other command packets will have different formats within the highlighted bytes.

Table 18. Port Request Block for PACKET Command

31							0	
Protocol Override				Control				0x00
Received Transfer Count								0x04
Features / Error	Command / Status	C	R	R	R	PMP	FIS Type	0x08
Dev/Head	Cyl High	Cyl Low					Sector Number	0x0C
Features (Exp)	Cyl High (Exp)	Cyl Low (Exp)					Sector Num (Exp)	0x10
Device Control	Reserved	Sector Count (Exp)					Sector Count	0x14
Reserved								0x18
Reserved – Must Be Zero								0x1C
LBA	LBA (MSB)	Reserved					ATAPI opcode	0x20
XFR Length (MSB)	Reserved	LBA (LSB)					LBA	0x24
Reserved							XFR Length (LSB)	0x28
Reserved								0x2C
SGE1 Data Address Low								0x30
SGE1 Data Address High								0x34
SGE1Data Count								0x38
SGE1 TRM	SGE1 LNK	SGE1 DRD	SGE1 XCF	Reserved[27:0]				0x3C

The SiI3132 controller does not decode the ATAPI command to determine the necessity or direction of any associated data transfer. The host driver must supply this information by setting control_packet_read (control field, bit 4) or control_packet_write (control field, bit 5) for any PACKET command that requires data transfer. Failure to set one of these bits for an ATAPI command that requests data transfer will result in an Overrun or Underrun Command Error condition.

Soft Reset PRB Structure

To send a soft-reset sequence, the host driver only needs to fill in the PMP field (offset 0x8, bits[11:8]) and set control_soft_reset (control field, bit 7). The controller will send a soft reset sequence to the device and wait for a Register – Device to Host FIS to deliver the device signature and terminate the command. Upon successful command completion, the host may inspect the FIS area of the slot in SiI3132 RAM (offset 0x08 through 0x1F) to determine the returned device signature. A soft reset is executed in the same manner as other PRBs, in the order in which it was issued. Port Ready (Port Status, bit 31) must be 1 in order to issue this command. The lightly shaded areas of Table 19 depict valid fields in slot RAM following successful command completion. These fields do not need to be supplied as inputs and may remain in any state upon command issuance.

Table 19. Port Request Block for Soft Reset Command

31							0	
N/A							Control (0x0080)	0x00
N/A								0x04
Features / Error	Command / Status	C	R	R	R	PMP	FIS Type	0x08
Dev/Head	Cyl High	Cyl Low					Sector Number	0x0C
Features (Exp)	Cyl High (Exp)	Cyl Low (Exp)					Sector Num (Exp)	0x10
Device Control	Reserved	Sector Count (Exp)					Sector Count	0x14
N/A								0x18
N/A								0x0C
N/A								0x10
N/A								0x14
N/A								0x18
N/A								0x1C
N/A								0x20
N/A								0x24
N/A								0x28
N/A								0x2C
N/A								0x30
N/A								0x34
N/A								0x38
N/A								0x3C

External Command PRB Structure

An external command PRB is indicated by setting control_external_command (control field, bit 2). External commands execute in a manner similar to standard commands except that the initial command FIS is fetched from host memory instead of the PRB FIS area. By default, an external command uses the *transparent* protocol. That means the command is terminated immediately following the successful transmission of the external command FIS. If this is not the desired protocol, the host driver can set control_protocol_override (control field, bit 0) and place the desired protocol in the Protocol Override field (offset 0x00, bits [31:16]).

The external command FIS length may be of any size (up to the 8 kB SATA limit) and is automatically padded to a Dword boundary. The SiI3132 controller will frame the FIS, adding SOF, EOF, and CRC. The host memory FIS image must contain the FIS header (FIS Type, PMP, etc.). The PRB PMP field (offset 0x08, bits [11:8]) must be populated to direct the FIS to the desired port multiplier port, or must be 0 if no port multiplier is attached. For port multiplier applications, it is important that the PMP field in the host-resident FIS and the PRB match for proper operation.

The location of the external command FIS is defined in additional SGEs with the XCF bit (SGE offset 0x0C, bit 28) set to one. Any type of command may be sent using an external command, including commands that have associated data transfers. Data transfer host memory locations are defined in SGEs with the XCF bit (SGE offset 0x0C, bit 28) set to zero. SGEs used to define the external command FIS and SGEs used to define data transfer may be freely mixed in any order. The presence or absence of the XCF bit informs the controller whether an SGE should be used for the current transfer operation.

Table 20. Port Request Block for External Commands

31					0	
Protocol Override				Control		0x00
Received Transfer Count						0x04
Reserved			N/A	PMP	Reserved	0x08
N/A						0x0C
N/A						0x10
N/A						0x14
N/A						0x18
Reserved – Must Be Zero						0x1C
SGE0 Data Address Low						0x20
SGE0 Data Address High						0x24
SGE0 Data Count						0x28
SGE0 TRM	SGE0 LNK	SGE0 DRD	SGE0 XCF	Reserved[27:0]		0x2C
SGE1 Data Address Low						0x30
SGE1 Data Address High						0x34
SGE1 Data Count						0x38
SGE1 TRM	SGE1 LNK	SGE1 DRD	SGE1 XCF	Reserved[27:0]		0x3C

Interlocked Receive PRB Structure

Reserving a command slot to receive an interlocked FIS is indicated by setting control_receive (control field, bit 3). To receive an interlocked FIS into host memory, the host driver first specifies the FIS type(s) to be interlocked by writing the appropriate value to the `Port FIS Configuration` register (port registers, offset 0x1028). The PRB is populated with SGEs that define the host memory region(s) that will be used to receive the interlocked FIS. When a FIS of the defined type is received, it is written to the defined host memory area and the command is completed. If an error occurs during receipt of the FIS, or if the SGEs define an area that is not large enough to contain the entire FIS, the FIS is rejected with an R_ERR response and the command is not completed. When an interlocked FIS is received without error into a memory region that is large enough to contain it, the command is successfully completed and the host driver may use the received FIS in any manner. The command slot is then free, and can be redefined as a receive slot or as any other command type.

After successfully receiving an interlocked FIS, the low-level link will be receiving WTRM primitives from the transmitting device, which is expecting a response. By default, the controller waits for the host driver to write a response bit to the port control register. If the host driver writes Interlock Accept (`Port Control Set` register, bit 12), an R_OK response is transmitted. If the host driver writes Interlock Reject (`Port Control Set` register, bit 11), an R_ERR response is transmitted. The host driver may also choose to set Auto Interlock Accept (`Port Control Set` register, bit 14) before performing interlocked operations. By setting this bit, an R_OK response is sent for all subsequently received interlocked FISs, without additional intervention from the host driver. In this mode, it is possible to receive one or more additional interlocked FISs before the host driver has had a chance to reserve command slots to receive them. If this occurs, any interlocked FIS that arrives without a reserved slot available will be acknowledged and discarded.

Table 21. Port Request Block For Receiving Interlocked FIS

31					0	
Protocol Override				Control		0x00
Received Transfer Count						0x04
						0x08
						0x0C
						0x10
						0x14
						0x18
Reserved – Must Be Zero						0x1C
SGE0 Data Address Low						0x20
SGE0 Data Address High						0x24
SGE0 Data Count						0x28
SGE0 TRM	SGE0 LNK	SGE0 DRD	SGE0 XCF	Reserved[27:0]		0x2C
SGE1 Data Address Low						0x30
SGE1 Data Address High						0x34
SGE1 Data Count						0x38
SGE1 TRM	SGE1 LNK	SGE1 DRD	SGE1 XCF	Reserved[27:0]		0x3C

Operation

Methods to Issue Commands

Before a command is executed, it must reside in a slot in SiI3132 controller RAM and must inform the controller that the PRB is ready to be executed. To accomplish this, the host must issue the command in one of two ways:

Indirect Method. The indirect method is the most common and flexible method of issuing commands. With this method, the host builds the PRB in host memory and writes the physical address of the PRB into one of 31 Command Activation registers, each associated with a command slot. This causes the controller to fetch the PRB from host memory and deposit it in the selected slot of SiI3132 RAM. After the command is fetched, the controller automatically informs the execution unit that the command is ready for execution.

The host may issue commands through additional Command Activation registers at any time regardless of whether the previous PRB has been fetched. The controller will fetch the PRBs in the order requested when the necessary resources are available.

Direct Method. The host may write the 64-byte PRB directly into SiI3132 slot RAM. The RAM area is defined in the port register map and the host can easily calculate the slot offset to write the PRB. After the PRB is written to RAM, the host informs the execution unit that it is ready to process by writing the slot number into the command execution FIFO register.

When the direct command issue method is used, it is not possible to append scatter/gather entries to the PRB without defining a LNK in one of the PRB resident scatter/gather entries.

Reset and Initialization

The SiI3132 controller has a hierarchical reset structure that allows initialization of the entire chip, a single port, an attached device, or the internal command queue. In general, asserting a reset at a HIGH level causes all underlying circuits to be reset. There are five levels of reset and initialization possible. The resets, listed from highest to lowest level, are described in the sections that follow.

PERST# Reset

The PERST# reset pin, when asserted, holds the entire chip in a reset state. All configuration, global, and port registers are initialized to their default state. When de-asserted, PCI Express configuration space is programmable, but the global

and port register spaces and the port state machines/command queue remain in a reset state until the Global and Port Resets are de-asserted through software control.

Global Reset

The Global Reset (`Global Control` register, bit 31), when asserted, initializes all global registers, except PHY Configuration, and all port registers to the default state. All port resets are set to one (asserted) while Global Reset is asserted. The Global Reset must be cleared to 0 to allow access to the global register space or to release any Port Reset. Software may use the Global Reset to initialize all ports with a single operation.

Port Reset

Each port contains a Port Reset bit (`Port Control Set` and `Port Control Clear` registers, bit 0) that remains set to *1* after the Global Reset is cleared to zero. While Port Reset is asserted, all port registers, except Port PHY Configuration and OOB Bypass (Port Control Set/Clear, bit 25), are initialized to their default state. The port state machines are reset and the command queue is cleared. The Port Reset must be cleared to 0 by writing a *1* to bit 0 of the `Port Control Clear` register to release the Port Reset condition. Software may assert the port reset condition at any time by writing a *1* to bit 0 of the `Port Control Set` register.

Device Reset

Each port contains a Device Reset (Port Control Set, bit 1) that may be used by software to reset an attached device without affecting the contents of the port registers. Writing a *1* to bit 1 of Port Control Set causes the execution state machines and pending command queue to be initialized. Then, a COMRESET is transmitted to the attached device. The effect of this sequence is to clear any outstanding commands and reset the attached device. The Device Reset bit is self-clearing. After the reset sequence has completed, the bit will be cleared to 0.

Port Initialize

Each port contains a Port Initialize (Port Control Set, bit 2) that may be used by software to initialize the port data structures without affecting the contents of the port registers or resetting the device. Writing a *1* to bit 1 of Port Control Set causes the execution state machines and pending command queue to be initialized. The effect of this sequence is to clear any outstanding commands. The Port Initialize bit is self-clearing. After the initialization sequence has completed, the bit will be cleared to 0.

Port Ready

Each port contains a Port Ready indicator (Port Status, bit 31) that is cleared to 0 by any of the above reset conditions. The Port Ready signal, when 1, indicates that the port is ready to execute commands. For all resets except Port Initialize, the Port Ready signal will not be asserted until a PHY ready condition is achieved. When Port Initialize is set, Port Ready will be cleared to 0 then set to 1 after any currently active data transfers or FIS transmission or reception operations have completed and port initialization has completed.

Port Reset Operation

Upon release of Port Reset, the low-level power management state machine is enabled and OOB signaling is initiated to the device. The SiI3132 controller starts OOB signaling by transmitting a COMRESET to the device. If the device responds with COMINIT and the OOB sequence is successful, a PHY ready condition results, indicating that a link has been successfully established and the device may transmit an initial register FIS. At this time, the Port Ready signal will be asserted, indicating that the host driver may issue commands. If the device does not respond within the prescribed time allowed for OOB, the low-level power management machine initiates another OOB sequence after a fixed delay. The period between OOB attempts is approximately 100 ms.

Upon receipt of an initial `Register – Device to Host` FIS that clears the task file status BSY state, the port is allowed to transmit commands to the device.

Initialization Sequence

The following is an example sequence of events that software might use to initialize the controller and enumerate an attached device or port multiplier. The sequence assumes that the system is powered up, the PERST# Reset has been deasserted, and the system has enumerated the PCI Express bus. Configuration space, including the Base Address Registers, has been initialized. It is now necessary to enable each port and to determine the device type, if any, that is attached to that port.

Initialize Port and Retrieve Device Signature

These steps initialize each port and retrieves the device signature for that port. Perform these steps for each port to be initialized.

1. Remove the Global Reset by writing 0x00000000 to the `Global Control` register (Global offset 0x40).
2. Clear Port Reset by writing *1* to the Port Reset bit of the `Port Control Clear` register (Port offset (port ● 0x2000) + 0x1004, bit 0).
3. If 32-bit platform and 32-bit activation is desired, write a *1* to the 32-bit Activation bit of the `Port Control Set` register (Port offset (port ● 0x2000) + 0x1000, bit 10).
4. To enable interrupts for command completion and command errors, write 0x00000003 to the `Port Interrupt Enable Set / Port Interrupt Enable Clear` (Port offset (port ● 0x2000) + 0x1010).
5. To determine if device is present, poll the SStatus register (Port offset (port ● 0x2000) + 0x1f04) for a PHYRDY condition indicated if the DET field (bits[3:0]) have a value of 0x3.
6. Wait until the Port Ready bit in the `Port Status` register (Port offset (port ● 0x2000) + 0x1000, bit 31) is set to *1*. If desired, an interrupt may be armed in the Port Interrupt Set Register (bit 2). Any change in Port Ready state asserts an interrupt.
7. If the software supports port multipliers, build a Soft Reset PRB in host memory. Set the PMP field to 0x0F to direct the command to the control port of the port multiplier. Issue the command to any available slot.

   Important: If the software does not support port multipliers, skip this step. Sending this command will cause the port multiplier to disable legacy access to device 0.
8. Upon successful completion of the soft reset command, read the device signature from the command slot (Port offset (port ● 0x2000) + (slot ● 0x80) + 0x14(LSB), 0x0C, 0x0D, 0x0E(MSB)).
9. Continue with one of the following, depending on the device signature:
   a. If the signature is 0x96690101, perform the Port Multiplier Enumeration Procedure.
   b. If the signature is 0xEB140101, perform the ATAPI PACKET Device Procedure.
   c. If the signature is 0x00000101, perform the Disk Drive Procedure.
10. Repeat Steps 2 through 9 for each port to be initialized.

Port Multiplier Enumeration Procedure

The attached device is a port multiplier. Perform these steps.

1. Enable Port Multiplier context switching by writing a *1* to PM Enable in the `Port Control Set` register (Port offset (port ● 0x2000) + 0x1000, bit 13).
2. Read the Port Multiplier GSCR[2] register by issuing a Read Port Multiplier command to the control port. This register contains the number of device ports on the Port Multiplier.
3. Enable the PHY by writing a *1*, then a *0*, to the `SControl` register (PSCR[2]) DET field. Issue a Port Multiplier Write command for each of these operations.
4. Wait for a PHYRDY condition in the port by polling the `Port Control Set` register (PSCR[0]) .
5. Clear the X-bit and all other error bits in the `SError` register (PSCR[1]) by writing all ones to the register with a `Write Port Multiplier` command. The port is now ready for operation.
6. Issue a `Soft Reset` command with the PMP field set to the appropriate port. This will return a device signature for the attached device.
7. Issue the appropriate `Identify Device` or `Identify Packet Device` command and any associated `Set Features`, `Set Write Multiple` commands necessary to initialize the device.
8. Repeat Steps 3 through 7 for each Port Multiplier Device Port.

ATAPI PACKET Device Procedure

The attached device is an ATAPI PACKET device. Perform these steps.

1. Issue an `Identify Packet Device` command to get device specific parameters.
2. While the drive is not ready and timeout has not expired, issue the Test unit ready PACKET command.
3. If the command completes successfully, go to Step 5.
4. If the command error indicates a Device error condition due to drive not ready, do the following:
   a. Write a *1* to the `Port Control Set` Register (port ● 0x2000) + 0x1000, bit 2).
   b. Wait until Port Ready in `Port Status` register (Port offset (port ● 0x2000) + 0x1000, bit 31) is 1. If desired, an interrupt may be armed by setting bit 2 in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` to *1*. Any change in Port Ready state asserts an interrupt.
5. The drive is ready for use. Issue appropriate `Set Features`, `Set Read Multiple` commands as needed.

Disk Drive Procedure

The attached device is a disk drive. Perform these steps.

1. Issue an `Identify Device` command to get device-specific parameters
2. The drive is ready for use. Issue appropriate `Set Features`, `Set Read Multiple` commands as needed.

Interrupts and Command Completion

Each port of the controller produces a single interrupt, which is an accumulation of various possible interrupt events. In its default mode, the device combines the interrupts from the ports into a single interrupt that may be used either for INTA emulation or for a Message Signaled Interrupt. In certain embedded environments, it might be desirable for each port to generate an independent interrupt. Software may configure each port to direct its interrupt to one of four emulated interrupts. The Interrupt Steering field in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` register (Port offset 0x1010/1014, bit [31:30]) is used to direct the port interrupt. By default, this field is set to *0*, indicating that the interrupt is directed to INTA. The register may be set to one of four values, shown in Table 22.

Table 22. Interrupt Steering

Interrupt Steering Value	Interrupt
0	INTA
1	INTB
2	INTC
3	INTD

Interrupt Sources

Figure 7 on page 29 depicts a logical representation of the interrupt routing for the SiI3132 controller. For each port, the possible interrupt causes are:

Command Completion. Indicates that one or more commands have successfully completed. This interrupt is cleared in one of two ways, depending on the state of Interrupt NCoR (`Port Control Set` register, bit 3). Reading the `Global Port Slot` Status register clears this interrupt condition if Interrupt NCoR is *0*. Writing a one to bit 0 or bit 16 of the `Port Interrupt Status` register clears this interrupt condition if Interrupt NCoR is *1*. This interrupt is enabled or disabled with the corresponding bit in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` register.

Command Error. Indicates that a command did not complete successfully. The port `Port Command Error` register will contain an error code indicating the actual cause of failure. When this bit is set, Port Ready is set to *0* and no additional commands will be processed until the port is initialized by one of the reset methods and Port Ready is asserted. Writing a one to bit 1 or bit 17 of the `Port Interrupt Status` register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the `Port Interrupt Enable Set / Port Interrupt Enable Clear`.

Port Ready. Indicates that the Port Ready state has changed from zero to one. Writing a *1* to bit 2 or bit 18 of the port Interrupt Status Register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the `Port Interrupt Enable Set / Port Interrupt Enable Clear register`.

Power Management Change. Indicates that the port power management state has been modified. The current power management state can be determined by reading the port `SStatus` register. Writing a *1* to bit 3 or bit 19 of the `Port Interrupt Status` register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` register.

PHY Ready Change. Indicates that the PHY state has changed from Not Ready to Ready or from Ready to Not Ready. The current PHY state can be determined by reading the port `SStatus` register. Writing a one to bit 4 or bit 20 of the `Port Interrupt Status` register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` register.

COMWAKE Received. Indicates that a COMWAKE OOB signal has been decoded on the receiver. Writing a one to bit 5 or bit 21 of the `Port Interrupt Status` register clears this interrupt condition. This interrupt is enabled

or disabled with the corresponding bit in the Port Interrupt Enable Set / Port Interrupt Enable Clear register.

Unrecognized FIS. Indicates that the F-bit has been set in the Serror Diag field. Writing a one to bit 6 or bit 22 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the Port Interrupt Enable Set / Port Interrupt Enable Clear register.

Device Exchanged. Indicates that the X-bit has been set in the Serror Diag field. The X-bit is set upon receipt of a COMINIT from the device. Writing a one to bit 7 or bit 23 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the Port Interrupt Enable Set / Port Interrupt Enable Clear register.

8b/10b Decode Error Threshold Exceeded. Indicates that the 8b/10b Decode Error counter has exceeded the programmed non-zero threshold value. Writing any value to the Port 8B/10B Decode Error Counter register or writing a *1* to bit 8 or 24 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled by writing a non-zero value to the threshold field (bit [31:16]) of the Port 8B/10B Decode Error Counter register. Writing a *0* the threshold field disables this interrupt.

CRC Error Threshold Exceeded. Indicates that the CRC Error counter has exceeded the programmed non-zero threshold value. Writing any value to the Port CRC Error Counter register or writing a *1* to bit 9 or bit 25 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled by writing a non-zero value to the threshold field (bit [31:16]) of the Port CRC Error Counter register. Writing a *0* to the threshold field disables this interrupt.

Handshake Error Threshold Exceeded. Indicates that the Handshake Error counter has exceeded the programmed non-zero threshold value. A handshake error occurs when an R_ERR primitive is received. Writing any value to the Port Handshake Error Counter register or writing a *1* to bit 10 or bit 26 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled by writing a non-zero value to the threshold field (bit [31:16]) of the Port Handshake Error Counter register. Writing a *0* to the threshold field disables this interrupt.

SDB Notify. Indicates that a Set Device Bits FIS has been received with the N-bit set in the control field. ATAPI and Port Multiplier devices optionally use this feature to signal the host that an event has occurred that requires further scrutiny. Writing a *1* to bit 11 or bit 27 of the Port Interrupt Status register clears this interrupt condition. This interrupt is enabled or disabled with the corresponding bit in the Port Interrupt Enable Set / Port Interrupt Enable Clear register.

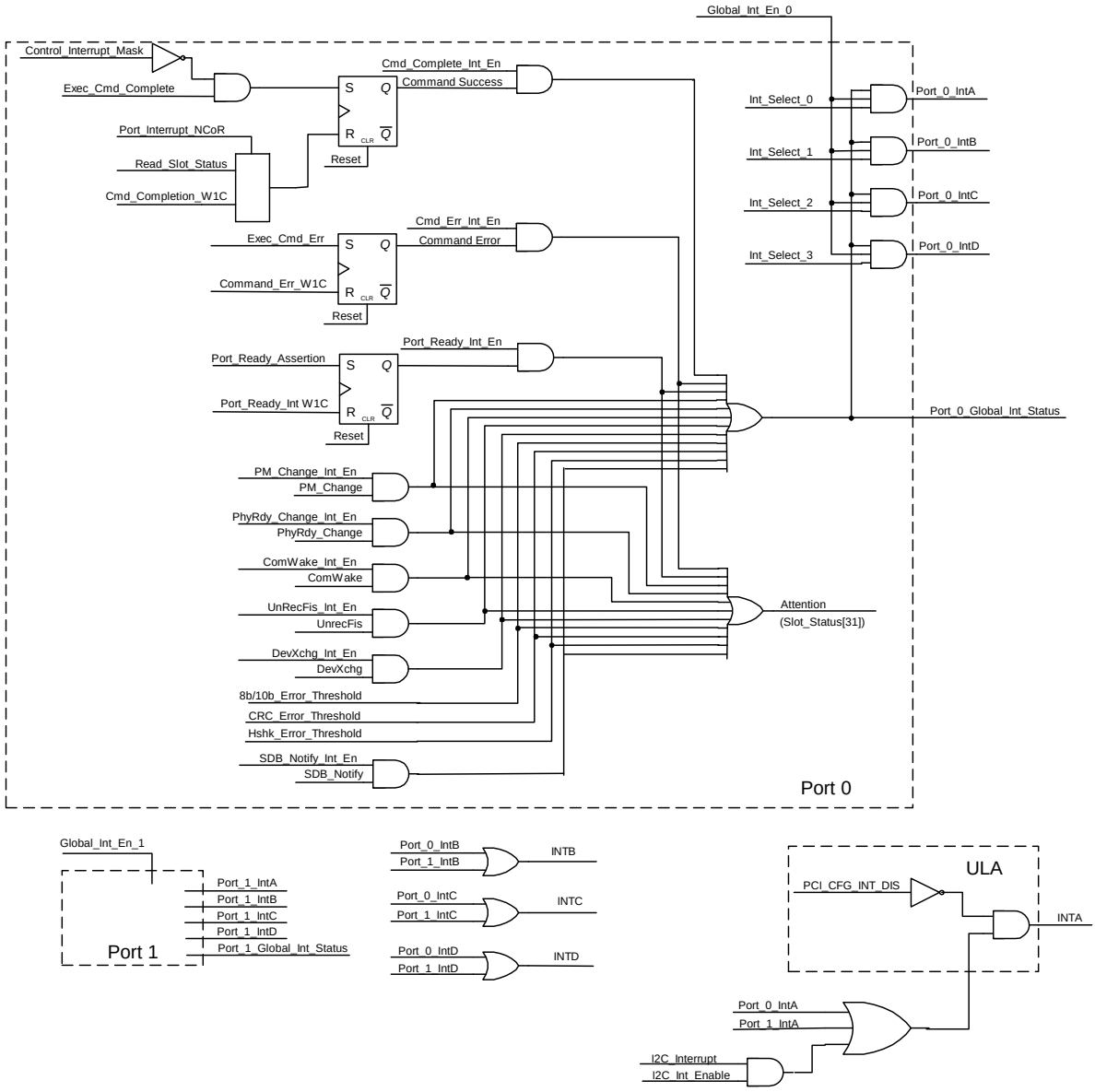


Figure 7. SiI3132 Interrupt Map

Table 23. Port Interrupt Causes and Control

Interrupt Cause	Interrupt Status Bit		To Clear:	To Enable:	To Disable:
	Masked	Raw			
Command Complete	0	16	If Interrupt W1C == 0 Read Slot Status If Interrupt W1C == 1 Write *1* to Port Interrupt Status bit 0 or 16, OR write one to desired port bit(s) in Global Interrupt Status.	Write 1 to Interrupt Enable Set bit 0	Write *1* to Interrupt Enable Clear bit 0 OR Write *1* to control_interrupt_mask in PRB Control field
Command Error	1	17	Write *1* to Interrupt Status bit 1 or 17	Write *1* to Interrupt Enable Set bit 1	Write *1* to Interrupt Enable Clear bit 1
Port Ready	2	18	Write *1* to Interrupt Status bit 2 or 18	Write *1* to Interrupt Enable Set bit 2	Write *1* to Interrupt Enable Clear bit 2
Power Management Change	3	19	Write *1* to Interrupt Status bit 3 or 19	Write *1* to Interrupt Enable Set bit 3	Write *1* to Interrupt Enable Clear bit 3
PHY Ready Change	4	20	Write *1* to Interrupt Status bit 4 or 20	Write *1* to Interrupt Enable Set bit 4	Write *1* to Interrupt Enable Clear bit 4
COMWAKE Received	5	21	Write *1* to Interrupt Status bit 5 or 21	Write *1* to Interrupt Enable Set bit 5	Write *1* to Interrupt Enable Clear bit 5
Unrecognized FIS Received	6	22	Write *1* to Interrupt Status bit 6 or 22	Write *1* to Interrupt Enable Set bit 6	Write *1* to Interrupt Enable Clear bit 6
Device Exchanged	7	23	Write *1* to Interrupt Status bit 7 or 23	Write *1* to Interrupt Enable Set bit 7	Write *1* to Interrupt Enable Clear bit 7
8b/10b Decode Error Threshold	8	24	Write *1* to Interrupt Status bit 8 or 24 OR Write any value to 8b/10b Decode Error Counter bits[15:0]	Write non-zero value to 8b/10b Decode Error Counter bits[31:16]	Write *0* to 8b/10b Decode Error Counter bits[31:16]
CRC Error Threshold	9	25	Write *1* to Interrupt Status bit 9 or 25 OR Write any value to CRC Error Counter bits[15:0]	Write non-zero value to CRC Error Counter bits[31:16]	Write *0* to CRC Error Counter bits[31:16]
Handshake Error Threshold	10	26	Write *1* to Interrupt Status bit 10 or 26 OR Write any value to Handshake Error Counter bits[15:0]	Write non-zero value to Handshake Error Counter bits[31:16]	Write zero to Handshake Error Counter bits[31:16]
Set Device Bits Notification Received	11	27	Write *1* to Interrupt Status bit 11 or 27	Write *1* to Interrupt Enable Set bit 11	Write *1* to Interrupt Enable Clear bit 11

Command Completion — The Slot Status Registers

The Slot Status registers are designed such that with a single read, an interrupt service routine can determine the successful completion state of outstanding commands, dismiss the command completion interrupt, and determine if any other enabled interrupt events are pending in a port.

The Port Slot Status register (Port offset 0x1800) or Global Port Slot Status (Global offset 0x00 + (port ● 4)) bits 0 through 30 reflect the status of each of the 31 command slots in a port. When a PRB is issued to a command slot, the corresponding bit in the Slot Status register is set to *1*, indicating that the command is in progress. When a command is successfully completed, the corresponding command slot bit is cleared in the Slot Status register. The host driver can read the Slot Status register at any time to determine the activity state of any issued commands.

By default, a successfully completed command will set the command complete bit in the Port Interrupt Status register. If the Command Complete interrupt is enabled, an interrupt will be asserted simultaneously. The host driver can optionally set control_interrupt_mask in the PRB Control field to prevent the command complete bit from being set on a per-command basis. This is useful when the host issues a series of commands and wants to be interrupted only after a selected command completes.

The command complete bit and associated interrupt are cleared when the Slot Status register is read, unless the host driver has set Interrupt No Clear on Read (Port Control Set or Port Control Clear register, bit 3) to *1*. If Interrupt No Clear on Read is set, the host driver must write *1* to the Command Complete bit in the Port Interrupt Status register in order to clear the command complete bit and associated interrupt.

The Attention Bit

Bit 31 of the Slot Status register is the Attention bit. When set to *1*, it indicates that an enabled interrupt source, other than command completion, is asserted. It is possible that the Slot Status register indicates an Attention condition while also showing that commands have successfully completed in bits 0 through 30. The interrupt service routine should always post-process any completed commands in addition to servicing a possible Attention condition. The Attention bit is set only for interrupt conditions that have been enabled as described in the Interrupt Sources section on page 27. The Attention bit remains set to *1* in the Slot Status register until all enabled interrupt conditions are cleared.

Interrupt Service Procedure

The SiI3132 controller services interrupt events with minimal host overhead. There are two methods the host can use to determine the cause of an interrupt within any of the ports.

- The Global Interrupt Status register (Global offset 0x44) can be read to determine which ports are causing the interrupt. Then, the Slot Status register for the interrupting ports can be read to determine the cause of an interrupt.
- Alternately, all port Slot Status registers can be read in a single burst operation from the Global Register space starting at Global offset 0x00. If Interrupt No Clear on Read (Port Control Set or Port Control Clear, Bit 3) is *0*, any command complete interrupt is cleared. The host driver should then compare the outstanding command status in bits 0 through 30 to its internal copy of outstanding commands to determine which, if any, commands have successfully completed.

Once the successful command completions have been noted, the host should check the Attention bit (bit 31) to determine if any other enabled interrupt events are pending on the port. If the Attention bit is one, the host should read the Port Interrupt Status register (Port offset (port ● 0x2000) + 0x1008) to determine the cause for the Attention condition. Once the Attention condition has been resolved and cleared, normal processing can continue.

Interrupt No Clear on Read

By default, the Command Completion interrupt condition is cleared when the port Slot Status register is read. In some cases, such as in debug environments, clearing of the Command Completion interrupt might not be the desired effect of reading the Slot Status register. In these cases, the host driver should set the Interrupt No Clear on Read bit (bit 3) in the Port Control Set register. When this bit is set, the host must clear the Command Completion interrupt by one of the following methods:

- Write one to the corresponding port interrupt status bit(s) in the Global Interrupt Status register (Global offset 0x44). This method clears Command Complete interrupts for multiple ports in a single write operation..
- Write *1* to bit 0 or bit 16 of the Port Interrupt Status register (Port offset (port ● 0x2000) + 0x1008).This clears Command Complete only for the corresponding port.

Error Processing

When an error occurs during command processing, the controller records the error condition and halts execution until the host driver is able to restore normal operation. It does not attempt to automatically recover from error conditions.

Rather, it provides the host with the necessary information to handle the error condition. Errors that occur during command execution cause the Command Error bit to be set to one in the Port Interrupt Status register (Port offset (port●0x2000) + 0x1008) and an error code to be placed in the Port Command Error register (Port offset (port ● 0x2000) + 0x1024). Refer to the Port Command Error section on page 63 for a complete list of possible error codes. Then execution halts and Port Ready (Port Status Register, bit 31) is cleared to zero. Only the port with the error condition halts and all other ports will continue to process normally. If the Command Error interrupt is enabled, an interrupt is asserted and the Attention bit is asserted in the port Slot Status register. The corresponding Slot Status bit for the command in error will not be cleared to *0*, because the command did not complete successfully. If only non-queued commands are outstanding, the slot number for the command in error is available in the Port Status register, bits[20:16]. The host may use this information to ascertain which outstanding command caused the error condition.

To recover from a Command Error condition, it is necessary to initialize the port by one of the Port Reset methods described in the Reset and Initialization starting on page 24. It might not be necessary to reset the device in all error cases. In fact, to properly recover from native queued error conditions, it may be necessary to send additional commands to the device in error to obtain additional error information. At the minimum, it will be necessary to assert a Port Initialize and wait for Port Ready before additional commands may be issued.

Errors may be grouped into three categories to determine the proper recovery action:

Recoverable errors. Error codes 1 and 2 are device specific errors. These errors occur when the device returns an error bit in the final register FIS or in a Set Device Bits FIS. Depending upon the severity of the error type reported by the device, it may not be necessary to reset the device. If the error code is 1, the register FIS received from the device is available in the command slot PRB. The host may determine the error reported by the device by examining the error register field of this structure. See the Error Recovery Procedures section below for more information regarding error recovery procedures.

Locally detected data errors. Error code 3 is a unique error type. It indicates that the controller detected an error during command execution but the device failed to report the error upon command completion. For non-queued commands, this type of error type can be treated in the same as a recoverable error. If queued commands are outstanding, the device must be reset because it is necessary to make sure that all queued commands are flushed from the device when an error condition occurs. Since the device did not report an error, it is unlikely that the queue has been flushed in the device.

Fatal Errors. All other error codes indicate that an error condition has occurred that requires both the device and the internal operational state of the SiI3132 controller to be reset. The most common method to perform this function is to issue a Device Reset.

Error Recovery Procedures

When a device returns error status for an outstanding command, the controller will halt command processing, post an error type of 1 or 2 in the Port Command Error register, set the command error bit in the Port Interrupt Status register and, if enabled, assert an interrupt to the host. The host driver can try to attempt error recovery without resetting the device that issued the error. Note that error recovery procedures should only be attempted for error types 1 and 2. Error type 3 is also recoverable if no queued commands are outstanding. Silicon Image recommends that all other error types result in a reset of the affected device(s).

If the device in error is directly attached to the SiI3132 controller port, the host may issue a Port Initialize by setting bit 2 in the Port Control Set register and waiting for a Port Ready condition. The host may then re-issue any commands that were outstanding when the error occurred. If native queued commands were outstanding, the host should issue a READ LOG EXTENDED for Log Page 10h to determine the details of the error condition. Refer to the Serial ATA II specification for further details on error handling with native queuing.

If the device in error is attached to a port multiplier, it is necessary for the host driver to wait until all outstanding commands to other devices attached to the port multiplier have completed before issuing the Port Initialize function. This is accomplished through a series of steps:

1. The host driver must note the PM port number for the device in error by extracting the PMP field (bit[8:5]) from the Port Context register (port offset 0x1E04). The PMP field contains the PM port number for the device in error. It is then necessary to determine if any commands are outstanding for non-error devices. If there are no commands outstanding for non-error devices, the host driver may proceed to step 4 to issue a Port Initialize command and wait for a Port Ready condition before reissuing commands.

2. If commands are outstanding to non-error devices, the host should set the Port Resume bit (bit 6) in the `Port Control Set` register. Setting this bit forces a Device Busy condition for the currently selected PM port (the port to which the device in error is attached) so that no additional issued commands will be sent to the device in error. Processing of commands that have already been issued continues.
3. The host driver must monitor command completion progress and determine when all commands for non-error devices have completed. Please note that the `Port Slot Status` register will still have a bit set for each outstanding command on the device in error. These bits will not be cleared and the host must ignore them while waiting for command completion on non-error devices. If another recoverable error occurs while waiting for commands to complete, the host driver must follow the same recovery steps for the new device in error, starting with step 1 above. It is possible to have multiple devices in an error recovery state concurrently. When the host driver detects that all commands for non-error devices have completed, it must perform the following steps.
   a. Clear Port Resume `Port Control Clear` register, Bit 6).
   b. Clear bit[16:13] in the `Port Device Status` register for the device(s) in error ((port ● 0x2000) + 0xF80 + (PM port of device in error ● 8)). This action clears the Device_Busy, Native_Queue, Legacy_Queue, and Service_Pending bits to ready the device for further command processing.
   c. Write zeroes (0x00000000) to the `Port Device QActive` register for the device(s) in error ((port ● 0x2000) + 0xF84 + (PM port of device in error ● 8)). This action ensures that all queued command context is removed before re-issuing commands.
   d. Issue a `Port Initialize` command and wait for Port Ready condition.
4. The host driver may now resume normal command processing. The host driver must determine which commands need to be reissued to the device in error. If native queued commands were outstanding to the device in error, the host must issue a READ LOG EXTENDED command to clear the pending error condition and determine the tag number (slot number) of the command in error before resuming command processing.

Note: It is a good idea to clear Port Resume (`Port Control Clear` register, Bit 6) whenever `Port Initialize` or `Port Device Reset` command is issued. This ensures that the Port Resume bit is always cleared when starting normal processing in the event that an abnormal exit is taken from the error recovery procedure.

Auto-Initialization

The controller supports an external flash or EEPROM device for BIOS extensions and user-defined PCI configuration header data.

Auto-Initialization from Flash

The SiI3132 controller initiates the flash detection and configuration space loading sequence on the release of PERST#. It begins by reading the highest two addresses ($7FFFF_H$ and $7FFFE_H$), checking for the correct data signature pattern (AA_H and 55_H, respectively). If the data signature pattern is correct, the controller continues to sequence the address downward, reading a total of twelve bytes. If the Data Signature is correct (55_H at $7FFFC_H$), the last eight bytes are loaded into the PCI Configuration Space registers.

If both flash and EEPROM are installed, the PCI Configuration Space registers will be loaded with data from the EEPROM.

While the sequence is active, the controller responds to all PCI bus accesses with a Target Retry.

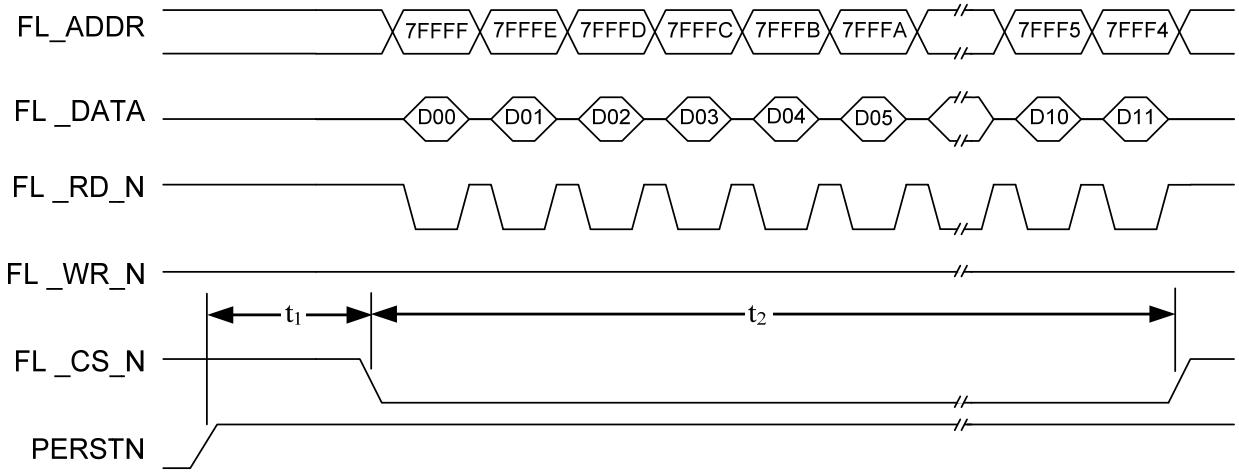


Figure 8. Auto-Initialization from Flash Timing

Table 24. Auto-Initialization from Flash Timing

Parameter	Value	Description
t_1	660 ns	PCI reset to flash Auto-Initialization cycle begin
t_2	4200 ns	Flash Auto-Initialization cycle time

Table 25. Flash Data Description

Address	Data Byte	Description
$7FFFF_H$	D00	Data Signature = AAH
$7FFFE_H$	D01	Data Signature = 55H
$7FFFD_H$	D02	AA = 120 ns flash device / Else, 240 ns flash device
$7FFFC_H$	D03	Data Signature = 55H
$7FFFB_H$	D04	PCI Device ID [23:16]
$7FFFA_H$	D05	PCI Device ID [31:24]
$7FFF9_H$	D06	PCI Class Code [15:08]
$7FFF8_H$	D07	PCI Class Code [23:16]
$7FFF7_H$	D08	PCI Sub-System Vendor ID [07:00]
$7FFF6_H$	D09	PCI Sub-System Vendor ID [15:08]
$7FFF5_H$	D10	PCI Sub-System ID [23:16]
$7FFF4_H$	D11	PCI Sub-System ID [31:24]

Auto-Initialization from EEPROM

The SiI3132 controller initiates the EEPROM detection and configuration space loading sequence after the flash read sequence. The SiI3132 controller supports EEPROMs with an I^2C serial interface. The sequence of operations consists of the following.

1. START condition defined as a HIGH-to-LOW transition on I2C_SDAT while I2C_SCLK is HIGH.
2. Control byte = 1010 (Control Code) + 000 (Chip Select) + 0 (Write Address).
3. Acknowledge.
4. Starting address field = 00000000.
5. Acknowledge.
6. Sequential data bytes separated by Acknowledges.
7. STOP condition.

While the sequence is active, the controller responds to all PCI bus accesses with a Target Retry.

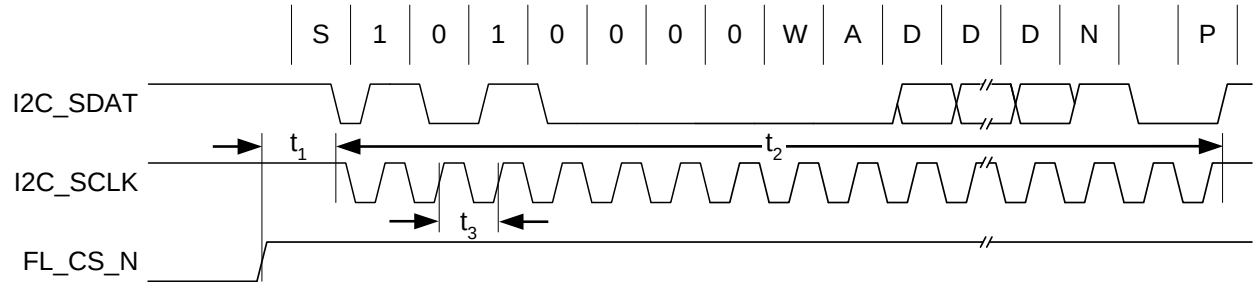


Figure 9. Auto-Initialization from EEPROM Timing

Table 26. Auto-Initialization from EEPROM Timing

Parameter	Value	Description
t_1	26.00 µs	End of Auto-Initialization from flash to start of Auto-Initialization from EEPROM
t_2	1.4 ms	Auto-Initialization from EEPROM cycle time
t_3	10 µs	EEPROM serial clock period

Table 27. Auto-Initialization from EEPROM Timing Symbols

Parameter	Description
S	START condition
W	R/W 0 = Write Command, 1 = Read Command
A	Acknowledge
D	Serial data
N	No-Acknowledge
P	STOP condition

Table 28. EEPROM Data Description

Address	Data Byte	Description
00_H	D00	Memory Present Pattern = AAH
01_H	D01	Memory Present Pattern = 55H
02_H	D02	Data Signature = AAH
03_H	D03	Data Signature = 55H
04_H	D04	PCI Device ID [23:16]
05_H	D05	PCI Device ID [31:24]
06_H	D06	PCI Class Code [15:08]
07_H	D07	PCI Class Code [23:16]
08_H	D08	PCI Sub-System Vendor ID [07:00]
09_H	D09	PCI Sub-System Vendor ID [15:08]
$0A_H$	D10	PCI Sub-System ID [23:16]
$0B_H$	D11	PCI Sub-System ID [31:24]

Register Definitions

This section describes the registers within the SiI3132 controller.

PCI Configuration Space

As shown in Table 29, the PCI Configuration Space registers define the operation of the controller on the PCI Express bus.

Table 29. SiI3132 PCI Configuration Space

Address Offset	Register Name			
00_H	Device ID		Vendor ID	
04_H	PCI Status		PCI Command	
08_H	PCI Class Code			Revision ID
$0C_H$	BIST	Header Type	Latency Timer	Cache Line Size
10_H	Base Address Register 0			
14_H				
18_H	Base Address Register 1			
$1C_H$				
20_H	Base Address Register 2			
24_H	Reserved			
28_H	Reserved			
$2C_H$	Subsystem ID		Subsystem Vendor ID	
30_H	Expansion ROM Base Address			
34_H	Reserved			Capabilities Ptr
38_H	Reserved			
$3C_H$	Max Latency	Min Grant	Interrupt Pin	Interrupt Line
40_H	Reserved			
44_H	Reserved			
48_H	Reserved			Hdr Wr Ena
$4C_H$	Reserved			
50_H	Reserved			
54_H	Power Management Capabilities		Next Capability	Pwr Mgt Cap ID
58_H	Data	Reserved	Control and Status	
$5C_H$	Message Control		Next Capability	MSI Cap ID
60_H	Message Address			
64_H				
68_H	Reserved		Message Data	
$6C_H$	Reserved			
70_H	PCI Express Capabilities Register		Next Capability	PCI Exp Cap ID
74_H	Device Capabilities			
78_H	Device Status		Device Control	
$7C_H$	Link Capabilities			
80_H	Link Status		Link Control	
84_H–EF_H	Reserved			
$F0_H$–FF_H	Indirect Access			
100_H	Advanced Error Reporting Capability			
104_H	Uncorrectable Error Status			
108_H	Uncorrectable Error Mask			
$10C_H$	Uncorrectable Error Severity			
110_H	Correctable Error Status			
114_H	Correctable Error Mask			
118_H	Advanced Error Capabilities and Control			
$11C_H$-$12B_H$	Header Log			

Device ID – Vendor ID

Address Offset: 00_H

Access Type: Read /Write

Reset Value: 0x3132_1095

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Device ID																Vendor ID															

This register defines the Device ID and Vendor ID associated with the SiI3132 controller. The register bits are defined as follows:

- **Bit [31:16]**: Device ID (R/W) – Device ID. The value in this bit field is one of the following:
  a. The default value of 0x3132 to identify the device as a Silicon Image SiI3132 controller.
  b. The value loaded from an external memory device; if an external memory device, flash or EEPROM, is present with the correct signature, the Device ID is loaded from that device after reset. See the Auto-Initialization from Flash section on page 34 for more information.
  c. System programmed value; if bit 0 of the Configuration register (48_H) is set, the Device ID is system programmable.
- **Bit [15:00]**: Vendor ID (R) – Vendor ID. This field defaults to 0x1095 to identify the vendor as Silicon Image.

PCI Status – PCI Command

Address Offset: 04_H

Access Type: Read/Write/Write-One-to-Clear

Reset Value: 0x0010_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Det Par Err	Sig Sys Err	Rcvd M Abort	Rcvd T Abort	Sig T Abort	Reserved		Det M Par Err	Reserved			Capabilities List	Int Status	Reserved								Int Disable	Reserved	SERR Enable	Reserved	Par Error Resp	Reserved			Bus Master	Memory Space	IO Space

This register defines the various control functions associated with the PCI bus. The register bits are defined below.

- **Bit 31**: Det Par Err (R/W1C) – Detected Parity Error.
- **Bit 30**: Sig Sys Err (R/W1C) – Signaled System Error.
- **Bit 29**: Rcvd M Abort (R/W1C) – Received Master Abort.
- **Bit 28**: Rcvd T Abort (R/W1C) – Received Target Abort.
- **Bit 27**: Sig T Abort (R/W1C) – Signaled Target Abort.
- **Bit 24**: Det M Par Err (R/W1C) – Detected Master Data Parity Error.
- **Bit 20**: Capabilities List (R) – PCI Capabilities List. This bit is hardwired to *1* to indicate that the controller implements Capabilities registers for Power Management, PCI-X, and Message Signaled Interrupt.
- **Bit [19]**: Interrupt Status (R).
- **Bit [26:25,23:21,18:11,9,7,5:3]**: Reserved (R) – This bit field is reserved and returns zeros on a read.
- **Bit [10]**: Interrupt Disable (R/W).
- **Bit 08**: SERR Enable (R/W) – SERR Enable.
- **Bit 06**: Par Error Resp (R/W) – Parity Error Response Enable.
- **Bit 02**: Bus Master (R/W) – Bus Master Enable. This bit set enables the controller to act as PCI bus master, for example, to issue Memory Requests.
- **Bit 01**: Memory Space (R/W) – Memory Space Enable. This bit set enables the controller to respond to memory space accesses.
- **Bit 00**: I/O Space (R/W) – I/O Space Enable. This bit is hardwired to *0*; the controller does not respond to I/O space accesses.

PCI Class Code – Revision ID

Address Offset: 08_H

Access Type: Read/Write

Reset Value: 0x0180_0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PCI Class Code																								Revision ID							

This register defines the various control functions associated with the PCI bus. The register bits are defined below.

- **Bit [31:08]**: PCI Class Code (R) – PCI Class Code. This value in this bit field is one of the following:
  a. The default value of 018000h for Mass Storage Class.
  b. The value loaded from an external memory device; if an external memory device – flash or EEPROM – is present with the correct signature, the PCI Class Code is loaded from that device after reset. See the Auto-Initialization from Flash section on page 34 for more information.
  c. System programmed value; if bit 0 of the Configuration register (48_H) is set the PCI Class Code is system programmable.
- **Bit [07:00]**: Revision ID (R) – Chip Revision ID. This bit field is hardwired to indicate the revision level of the chip design; revision 01_H is defined by this specification.

BIST – Header Type – Latency Timer – Cache Line Size

Address Offset: $0C_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
BIST								Header Type								Latency Timer								Cache Line Size							

This register defines the various control functions associated with the PCI bus. The register bits are defined below.

- **Bit [31:24]**: BIST (R). This bit field is hardwired to 00_H.
- **Bit [23:16]**: Header Type (R). This bit field is hardwired to 00_H.
- **Bit [15:08]**: Latency Timer (R). This field is hardwired to 00_H.
- **Bit [07:00]**: Cache Line Size (R/W). This bit field is Read/Write for legacy purposes. The field is not used by the SiI3132 controller.

Base Address Register 0

Address Offset: 10_H

Access Type: Read/Write

Reset Value: 0x0000_0000_0000_0004

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Base Address Register 0																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Base Address Register 0																									0000100						

This register defines the addressing of the Global Registers within the SiI3132 controller. The register bits are defined as follows:

- **Bit [63:07]**: Base Address Register 0 (R/W). This register defines the base address for the 128-byte Memory Space containing the Global Registers.
- **Bit [06:00]**: (R). This bit field is hardwired to 0000100_B to indicate a 64-bit base address.

Base Address Register 1

Address Offset: 18_H

Access Type: Read/Write

Reset Value: 0x0000_0000_0000_0004

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Base Address Register 1																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Base Address Register 1																		00 0000 0000 0100													

This register defines the addressing of the Port Registers and LRAM within the controller. The register bits are defined as follows:

- **Bit [63:15]**: Base Address Register 1 (R/W). This register defines the base address for the 16 kB Memory Space containing the Port Registers.
- **Bit [14:00]**: (R). This bit field is hardwired to 0004_H to indicate a 64-bit base address.

Base Address Register 2

Address Offset: 20_H

Access Type: Read/Write

Reset Value: 0x0000_0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Base Address Register 2																												000 0001			

This register defines the addressing of the Indirect I/O registers within the SiI3132 controller. The register bits are defined as follows:

- **Bit [31:04]**: Base Address Register 2 (R/W). This register defines the base address for the 128-byte I/O Space.
- **Bit [03:00]**: (R). This bit field is hardwired to 000_0001_B.

Subsystem ID — Subsystem Vendor ID

Address Offset: $2C_H$

Access Type: Read/Write

Reset Value: 0x3132_1095

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Subsystem ID																Subsystem Vendor ID															

This register defines the Subsystem ID fields associated with the PCI bus. The register bits are as follows.

- **Bit [31:16]**: Subsystem ID (R/W) – Subsystem ID. The value in this bit field is one of the following:
  a. The default value of 0x3132
  b. The value loaded from an external memory device; if an external memory device (flash or EEPROM) is present with the correct signature, the Subsystem ID is loaded from that device after reset. See Auto-Initialization from Flash section on page 34 for more information.
  c. System programmed value; if bit 0 of the Configuration register (48_H) is set, the Subsystem ID is system programmable.
- **Bit [15:00]**: Subsystem Vendor ID (R/W) – Subsystem Vendor ID. The value in this bit field is one of the following:
  a. The default value of 0x1095
  b. The value loaded from an external memory device; if an external memory device (flash or EEPROM) is present with a correct signature, the Subsystem Vendor ID is loaded from that device after reset. See the Auto-Initialization from Flash section on page 34 for more information.
  c. System programmed value; if bit 0 of the Configuration register (48_H) is set, the Subsystem Vendor ID is system programmable.

Expansion ROM Base Address

Address Offset: 30_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Expansion ROM Base Address													000_0000_0000_0000_000																		Exp ROM Enable

This register defines the Expansion ROM base address associated with the PCI bus. The register bits are defined as follows:

- **Bit [31:19]**: Expansion ROM Base Address (R/W) – Expansion ROM Base Address. This bit field defines the upper bits of the Expansion ROM base address.
- **Bit [18:01]**: (R). This bit field is hardwired to 00000_H to indicate that the Expansion ROM address range is 512 kB.
- **Bit [00]**: Exp ROM Enable (R/W) – Expansion ROM Enable. This bit is set to enable Expansion ROM access.

Capabilities Pointer

Address Offset: 34_H

Access Type: Read

Reset Value: 0x0000_0054

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																								Capabilities Pointer							

This register defines the link to a list of new capabilities associated with the PCI bus. The register bits are defined below.

- **Bit [31:08]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [07:00]**: Capabilities Pointer (R) – Capabilities Pointer. This bit field contains 54_H, the address for the first Capabilities register set, the PCI `Power Management Capability` register.

Max Latency – Min Grant – Interrupt Pin – Interrupt Line

Address Offset: $3C_H$

Access Type: Read/Write

Reset Value: 0x0000_0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Max Latency								Min Grant								Interrupt Pin								Interrupt Line							

This register defines various control functions associated with the PCI bus. The register bits are defined as follows:

- **Bit [31:24]**: Max Latency (R) – Maximum Latency. This bit field is hardwired to 00_H.
- **Bit [23:16]**: Min Grant (R) – Minimum Grant. This bit field is hardwired to 00_H.
- **Bit [15:08]**: Interrupt Pin (R) – Interrupt Pin Used. This bit field is hardwired to 01_H to indicate that the controller uses the INTA interrupt. The INTB, INTC, and INTD interrupts can be used by enabling them in the `Port Interrupt Enable Set / Port Interrupt Enable Clear` registers but it is outside the PCI specification.
- **Bit [07:00]**: Interrupt Line (R/W) – Interrupt Line. This bit field is used by the system to indicate interrupt line routing information. The controller does not use this information.

Header Write Enable

Address Offset: 48_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																														Ind Acc Ena	Hdr Wr Ena

- **Bit [31:02]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [01]**: Ind Acc Ena (R) – Indirect Access Enable. This bit enables the Indirect Access registers at offset $F0_H$-FF_H.
- **Bit [00]**: Hdr Wr Ena (R) – Header Write Enable. This bit enables writing to registers defined as read-only by the PCI specification. This bit is required to meet PCI compliance testing and expects certain registers to be read-only. This bit is set to Enable Write Access to the registers; Device ID (03-02_H), PCI Class Code (09-$0B_H$), Subsystem Vendor ID ($2D$-$2C_H$), and Subsystem ID ($2F$-$2E_H$) in the PCI Configuration Header.

Power Management Capability

Address Offset: 54_H

Access Type: Read Only

Reset Value: 0x0622_5C01

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PME Support					PPM D2 Support	PPM D1 Support	Auxiliary Current			Dev Special Init	Reserved	PME Clock	PPM Rev			Next Capability Pointer120								Capability ID							

This register defines the power management capabilities associated with the PCI bus. The register bits are defined as follows:

- **Bit [31:27]**: PME Support (R) – Power Management Event Support. This bit field is hardwired to 00_H. The controller does not support PME.
- **Bit [26]**: PPM D2 Support (R) – PCI Power Management D2 Support. This bit is hardwired to 1.
- **Bit [25]**: PPM D1 Support (R) – PCI Power Management D1 Support. This bit is hardwired to 1.
- **Bit [24:22]**: Auxiliary Current (R) – Auxiliary Current. This bit field is hardwired to 000_B.
- **Bit [21]**: Dev Special Init (R) – Device Special Initialization. This bit is hardwired to 1 to indicate that the controller requires special initialization.
- **Bit [20]**: Reserved (R). This bit is reserved and returns 0 on a read.
- **Bit [19]**: PME Clock (R) – Power Management Event Clock. This bit is hardwired to 0.
- **Bit [18:16]**: PPM Rev (R) – PCI Power Management Revision. This bit field is hardwired to 010_B to indicate compliance with the PCI Power Management Interface Specification revision 1.1.
- **Bit [15:08]**: Next Capability Pointer (R) – PCI Next Capability Pointer. This bit field is hardwired to $5C_H$ to point to the second Capabilities register, the `MSI Capability` register.
- **Bit [07:00]**: Capability ID (R) – PCI Capability ID. This bit field is hardwired to 01_H to indicate that this is a PCI Power Management Capability.

Power Management Control + Status

Address Offset: 58_H

Access Type: Read/Write

Reset Value: 0x0C00_2000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PPM Data								Reserved								PME Status	PPM Data Scale		PPM Data Sel				PME Ena	Reserved						PPM Power State	

This register defines the power management capabilities associated with the PCI bus. The register bits are defined as follows:

- **Bit [31:24]**: PPM Data (R) – PCI Power Management Data. This bit field is hardwired to 0x0C to indicate a power consumption of 1.2 Watt.
- **Bit [23:16]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [15]**: PME Status (R) – PME Status. This bit is hardwired to 0. The controller does not support PME.
- **Bit [14:13]**: PPM Data Scale (R) – PCI Power Management Data Scale. This bit field is hardwired to 01_B to indicate a scaling factor of 100 mW.
- **Bit [12:09]**: PPM Data Sel (R/W) – PCI Power Management Data Select. This bit field is set by the system to indicate which data field is to be reported through the PPM Data bits (although current implementation hardwires the PPM Data to indicate 1.2 Watt).
- **Bit [08]**: PME Ena (R) – PME Enable. This bit is hardwired to 0. The controller does not support PME.
- **Bit [07:02]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [01:00]**: PPM Power State (R/W) – PCI Power Management Power State. This bit field is set by the system to dictate the current Power State: 00 = D0 (Normal Operation), 01 = D1, 10 = D2, and 11 = D3 (Hot).

MSI Capability

Address Offset: $5C_H$

Access Type: Read/Write

Reset Value: 0x0080_7005

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved								64-bit Addr	Multiple Message Enable			Multiple Message Capable			MSI Enable	Next Capability Pointer								Capability ID							

This register defines the MSI Capability Message Control. The register bits are defined as follows:

- **Bit [31:24]**: Reserved (R) – This bit field is reserved and returns zeros on a read.
- **Bit [23]**: 64-bit Addr (R) – 64-bit Address Capable. This bit is hardwired to 1.
- **Bit [22:20]**: Multiple Message Enable (R/W) – This bit field defaults to 000_B.
- **Bit [19:17]**: Multiple Message Capable (R/W) – This bit field defaults to 000_B.
- **Bit [16]**: MSI Enable (R/W) – This bit is set to enable Message Signaled Interrupts.
- **Bit [15:08]**: Next Capability Pointer (R) –Next Capability Pointer. This bit field is hardwired to 70_H to point to the third Capabilities register, the `PCI Express Capability` register.
- **Bit [07:00]**: Capability ID (R) – This bit field is hardwired to 05_H to indicate that this is a MSI Capability.

Message Address

Address Offset: 60_H–67_H

Access Type: Read/Write

Reset Value: 0x0000_0000_0000_0000

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Message Address Upper																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Message Address																														00	

This register specifies the memory address for an MSI memory write transaction. The memory address must be of a Dword (bits 1:0 must be 0).

MSI Message Data

Address Offset: 68_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																Message Data															

This register specifies the MSI Message Data. The register bits are defined as follows:

- **Bit [31:16]**: Reserved (R) – This bit field is reserved and returns zeros on a read.
- **Bit [15:00]**: Message Data (R/W) – This bit field specifies the Message Data for an MSI memory write transaction.

PCI Express Capability

Address Offset: 70_H

Access Type: Read Only

Reset Value: 0x0011_0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved		Interrupt Message Number					Reserved	Device Type				Version				Next Capability Pointer								Capability ID							

- **Bit [31:30,24]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [29:25]**: Interrupt Message Number (R) – This bit field is hardwired to 0.
- **Bit [23:20]**: Device Type (R) – This bit field is hardwired to 0001_B to indicate a PCI Express Legacy Endpoint device.
- **Bit [19:16]**: Version (R) – This bit field is hardwired to 01_H to indicate compliance with the PCI Express Specification revision 1.0a.
- **Bit [15:08]**: Next Capability Pointer (R) – PCI Next Capability Pointer. This bit field is hardwired to 00_H (this is the last capability).
- **Bit [07:00]**: Capability ID (R) – PCI Capability ID. This bit field is hardwired to 10_H to indicate that this is a PCI Express Capability.

Device Capabilities

Address Offset: 74_H

Access Type: Read Only

Reset Value: 0x0000_0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved				Slot Power										Reserved			Pwr Indicator	Atten Indicator	Atten Button	L1 Latency			L0s Latency			Ext Tag Sup	Phantom Functions		Max Payload		

- **Bit [31:28,17:15]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [27:18]**: Slot Power (R) – Captured Slot Power Limit Value and Limit Scale. This bit field is hardwired to 0.
- **Bit [14]**: Pwr Indicator (R) – Power Indicator Present. This bit field is hardwired to 0.
- **Bit [13]**: Atten Indicator (R) – Attention Indicator Present. This bit field is hardwired to 0.
- **Bit [12]**: Atten Button (R) – Attention Button Present. This bit field is hardwired to 0.
- **Bit [11:09]**: L1 Latency (R) – This bit field is hardwired to 000_B.
- **Bit [08:06]**: L0s Latency (R) – This bit field is hardwired to 000_B.
- **Bit [05]**: Ext Tag Sup (R) – Extended Tag Field Supported. This bit is hardwired to 0.
- **Bit [04:03]**: Phantom Functions (R) – This bit field is hardwired to 0.
- **Bit [02:00]**: Max Payload (R) – Max_Payload_Size Supported. This bit field is hardwired to 011_B to indicate that a maximum 1024-byte payload is supported.

Device Status and Control

Address Offset: 78_H

Access Type: Read/Write/Write 1 to Clear

Reset Value: 0x0000_2000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved										Trans Pending	AUX Power	Unsup Req Det	Fatal Error Det	Non-Fatal Error	Corr Error Det	Reserved	Max Read Request Size			EnSnp Not Req	Aux Pwr PM En	Phntm Fnc En	Ext Tag Fld En	Max Payload Size			En Rlxd Ord	UnsReq Rep En	Fatal Err Rep En	NonFtl Err Rep En	Corr Err Rep En

- **Bit [31:22,15]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [21]**: Trans Pending (R) – Transactions Pending.
- **Bit [20]**: AUX Power (R) – AUX Power Detected. This bit is hardwired to 0.
- **Bit [19]**: Unsup Req Det (R/W1C) – Unsupported Request Detected.
- **Bit [18]**: Fatal Error Det (R/W1C) – Fatal Error Detected.
- **Bit [17]**: Non-Fatal Error (R/W1C) – Non-Fatal Error Detected.
- **Bit [16]**: Corr Error Det (R/W1C) – Correctable Error Detected.
- **Bit [14:12]**: Max Read Request Size (R/W) – Allowable values are 000_B to 011_B (128 to 1024 bytes). Default is 010_B (512 bytes).
- **Bit [11]**: EnSnp Not Req (R) – Enable No Snoop. This bit is hardwired to 0.
- **Bit [10]**: Aux Pwr PM En (R) – Auxiliary Power PM Enable. This bit is hardwired to 0.
- **Bit [09]**: Phntm Fnc En (R) – Phantom Functions Enable. This bit is hardwired to 0.
- **Bit [08]**: Ext Tag Fld En (R) – Extended Tag Field Enable. This bit is hardwired to 0.
- **Bit [07:05]**: Max Payload Size (R/W) – Allowable values are 000_B to 011_B (128 to 1024 bytes). Default is 000_B (128 bytes).

- **Bit [04]**: En Rlxd Ord (R) – Enable Relaxed Ordering. This bit field is hardwired to 0.
- **Bit [03]**: UnsReq Rep En (R/W) – Unsupported Request Reporting Enable.
- **Bit [02]**: Fatal Err Rep En (R/W) – Fatal Error Reporting Enable.
- **Bit [01]**: NonFtl Err Rep En (R/W) – Non-Fatal Error Reporting Enable.
- **Bit [00]**: Corr Err Rep En (R/W) – Correctable Error Reporting Enable.

Link Capabilities

Address Offset: $7C_H$

Access Type: Read Only

Reset Value: 0x0000_7411

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Port Number								Reserved						L1 Exit Latency			L0s Exit Latency			ASPM Support		Maximum Link Width						Maximum Link Speed			

- **Bit [31:24]**: Port Number (R) – This bit field is hardwired to 00_H.
- **Bit [23:18]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [17:15]**: L1 Exit Latency (R) – This bit field is hardwired to 000_B.
- **Bit [14:12]**: L0s Exit Latency (R) – This bit field is hardwired to 111_B.
- **Bit [11:10]**: ASPM Support (R) – This bit field is hardwired to 01_B.
- **Bit [09:04]**: Maximum Link Width (R) – This bit field is hardwired to 000001_B.
- **Bit [03:00]**: Maximum Link Speed (R) – This bit field is hardwired to 0001_B.

Link Status and Control

Address Offset: 80_H

Access Type: Read/Write

Reset Value: 0x0011_0000 or 0x1011_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved			Slot Clk Config	Link Training	Link Train Err	Negotiated Link Width						Link Speed				Reserved								Ext Synch	Comm Clk Cfg	Retrain Link	Link Disable	RCB	Reserved	ASPM Control	

- **Bit [31:29,15:08,02]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [28]**: Slot Clk Config (R) – Slot Clock Configuration. This bit is 1 if the reference clock is detected.
- **Bit [27]**: Link Training (R) –This bit is hardwired to 0.
- **Bit [26]**: Link Train Err (R) – This bit is hardwired to 0.
- **Bit [25:20]**: Negotiated Link Width (R) – This bit field is hardwired to 000001_B.
- **Bit [19:16]**: Link Speed (R) – This bit field is hardwired to 0001_B.
- **Bit [07]**: Ext Synch (R/W) – Extended Synch.
- **Bit [06]**: Comm Clk Cfg (R/W) – Common Clock Configuration.
- **Bit [05]**: Retrain Link (R) – This bit is hardwired to 0.
- **Bit [04]**: Link Disable (R) – This bit is hardwired to 0.
- **Bit [03]**: RCB (R/W) – Read Completion Boundary.
- **Bit [01:00]**: ASPM Control (R/W)

Global Register Offset

Address Offset: $F0_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																									Dword Offset					00	

This register provides indirect addressing of a Global Register otherwise accessible directly through Base Address Register 0. The Dword address offset for an indirect access is in bits 6 to 2; bits 31 to 7, 1, and 0 are reserved and should always be *0*.

This is physically the same register that is addressed by Base Address Register 2, Offset 00_H.

Global Register Data

Address Offset: $F4_H$

Access Type: Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
As defined for indirectly accessed register																															

This register provides the indirect access addressed by the Global Register Offset register.

Port Register Offset

Address Offset: $F8_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																		Dword Offset												00	

This register provides indirect addressing of a Port Register otherwise accessible directly through Base Address Register 1. The Dword address offset for an indirect access is in bits 13 to 2; bits 31 to 14, 1, and 0 are reserved and should always be 0.

This is physically the same register that is addressed by Base Address Register 2, Offset 08_H.

Port Register Data

Address Offset: FC_H

Access Type: Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
As defined for indirectly accessed register																															

This register provides the indirect access addressed by the Port Register Offset register.

Advanced Error Reporting Capability

Address Offset: 100_H

Access Type: Read Only

Reset Value: 0x0001_0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Next Capability Pointer												Version				Extended Capability ID															

- **Bit [31:20]**: Next Capability Pointer (R) – PCI Next Capability Pointer. This bit field is hardwired to 000_H (this is the last capability).
- **Bit [19:16]**: Version (R) – This bit field is hardwired to 01_H to indicate compliance with the PCI Express Specification revision 1.0a.
- **Bit [15:00]**: Extended Capability ID (R) – PCI Capability ID. This bit field is hardwired to 0001_H to indicate that this is an Advanced Error Reporting Capability.

Uncorrectable Error Status

Address Offset: 104_H

Access Type: Read/Write 1 to Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved											Unsup Req Err	ECRC Error	Malformed TLP	Rx Overflow (0)	Unexp Comp	Comp Abort	Comp Timeout	FC Protocol Err	Poisoned TLP	Reserved							DL Protocol Err	Reserved			Training Error

- **Bit [31:21,11:05,03:01]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [20]**: Unsup Req Err (R/W1C) – Unsupported Request Error Status.
- **Bit [19]**: ECRC Error (R/W1C) – ECRC Error Status.
- **Bit [18]**: Malformed TLP (R/W1C) – Malformed TLP Status.
- **Bit [17]**: Rx Overflow (R) – Receiver Overflow Status; always 0.
- **Bit [16]**: Unexp Comp (R/W1C) – Unexpected Completion Status.
- **Bit [15]**: Comp Abort (R/W1C) – Completer Abort Status.
- **Bit [14]**: Comp Timeout (R/W1C) – Completion Timeout Status.
- **Bit [13]**: FC Protocol Err (R/W1C) – Flow Control Protocol Error Status. This bit is hardwired to 0 (as are its mask and error severity bits).
- **Bit [12]**: Poisoned TLP (R/W1C) – Poisoned TLP Status.
- **Bit [04]**: DL Protocol Err (R/W1C) – Data Link Protocol Error Status.
- **Bit [00]**: Training Error (R) – This bit is hardwired to 0 (as are its mask and error severity bits).

Uncorrectable Error Mask

Address Offset: 108_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved											Unsup Req Err	ECRC Error	Malformed TLP	Rx Overflow (0)	Unexp Comp	Comp Abort	Comp Timeout	FC Protocol Err	Poisoned TLP	Reserved								DL Protocol Err	Reserved		Training Error

The bits of this register are the mask bits for corresponding bits of the Uncorrectable Error Status register.

Uncorrectable Error Severity

Address Offset: $10C_H$

Access Type: Read/Write

Reset Value: 0x0004_0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved											Unsup Req Err	ECRC Error	Malformed TLP	Rx Overflow (0)	Unexp Comp	Comp Abort	Comp Timeout	FC Protocol Err	Poisoned TLP	Reserved								DL Protocol Err	Reserved		Training Error

The bits of this register are the error severity bits for corresponding bits of the Uncorrectable Error Status register.

Correctable Error Status

Address Offset: 110_H

Access Type: Read/Write 1 to Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																			Replay Timeout	Reserved			REPLAY_NUM	Bad DLLP	Bad TLP	Reserved					Rx Error

- **Bit [31:13,11:09,05:01]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [12]**: Replay Timeout (R/W1C) – Replay Timer Timeout Status.
- **Bit [08]**: REPLAY_NUM (R/W1C) – REPLAY_NUM Rollover Status.
- **Bit [07]**: Bad DLLP (R/W1C) – Bad DLLP Status.
- **Bit [06]**: Bad TLP (R/W1C) – Bad TLP Status.
- **Bit [00]**: Rx Error (R/W1C) – Receiver Error Status. This bit is hardwired to 0 (as is the corresponding mask bit).

Correctable Error Mask

Address Offset: 114_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																			Replay Timeout	Reserved			REPLAY_NUM	Bad DLLP	Bad TLP	Reserved						Rx Error

The bits of this register are the mask bits for corresponding bits of the Correctable Error Status register.

Advanced Error Capabilities and Control

Address Offset: 118_H

Access Type: Read/Write

Reset Value: 0x0000_00A0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																							ECRC Chk En	ECRC Chk Cap	ECRC Gen En	ECRC Gen Cap	First Error Pointer				

- **Bit [31: 09]**: Reserved (R) – These bits are reserved and return zero on a read.
- **Bit [08]**: ECRC Chk En (R/W) – ECRC Check Enable.
- **Bit [07]**: ECRC Chk Cap (R) – ECRC Check Capable. This bit is hardwired to 1.
- **Bit [06]**: ECRC Gen En (R/W) – ECRC Generation Enable.
- **Bit [05]**: ECRC Gen Cap (R) – ECRC Generation Capable. This bit is hardwired to 1.
- **Bit [04:00]**: First Error Pointer (R).

Header Log

Address Offset: $11C_H$ / 120_H / 124_H / 128_H

Access Type: Read Only

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Header Log (1st Dword)																															
Header Byte 0								Header Byte 1								Header Byte 2								Header Byte 3							
Header Log (2nd Dword)																															
Header Byte 4								Header Byte 5								Header Byte 6								Header Byte 7							
Header Log (3rd Dword)																															
Header Byte 8								Header Byte 9								Header Byte 10								Header Byte 11							
Header Log (4th Dword)																															
Header Byte 12								Header Byte 13								Header Byte 14								Header Byte 15							

This 16-byte register contains the header of a TLP associated with an error.

Internal Register Space – Base Address 0

These registers are 32 or 64 bits wide and are the Global Registers of the SiI3132 controller. Access to this register space is through the PCI Memory space.

Table 30. SiI3132 Internal Register Space – Base Address 0

Address Offset	Register Name	
00_H	Port 0 Slot Status	
04_H	Port 1 Slot Status	
08_H–$3F_H$	Reserved	
40_H	Global Control	
44_H	Global Interrupt Status	
48_H	PHY Configuration	
$4C_H$–$5F_H$	Reserved	
60_H	I^2C Control	
64_H	I^2C Status	
68_H	I^2C Slave Address	
$6C_H$	I^2C Data Buffer	
70_H	Flash Address	
74_H	GPIO	Flash Data
78_H–$7F_H$	Reserved	

Global Port Slot Status

Address Offset: 00_H-07_H

Access Type: Read

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Attention	Slot Status																														

These two registers provide the Status for the 31 Command Slots for each of the 2 ports. These registers also appear in Port register space. Reading this register clears the Command Completion Status for the port if the Interrupt No Clear on Read bit (bit 3) of the `Port Control Set` register is *0*. The register bits are defined as follows:

- **Bit [31]**: Attention (R) – This bit indicates that something occurred in the corresponding port and it requires the attention of the host. Other port registers must be examined to determine the origin of the error. This bit is the logical OR of the masked interrupt conditions, except for Command Completion, reported in the `Port Interrupt Status` register.
- **Bit [30:00]**: Slot Status (R) – These bits are the Active status bits corresponding to Slot numbers 30 to 0. The Active status bit for a slot is set when the Slot number is written to the Command Execution FIFO using the direct command transfer method or when a Command Activation register is written using the indirect command transfer method..

Global Control

Address Offset: 40_H

Access Type: Read/Write

Reset Value: 0x8100_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Global Reset	MSIACK	I^2C Int Enable	Reserved				3Gb/s Capable	Reserved																						Port 1 Int Enable	Port 0 Int Enable

This register controls various functions of the chip.

- **Bit [31]**: Global Reset (R/W). This bit, when set to 1, asserts a port reset to all ports. This bit must be cleared to 0 to allow normal operation. Once set by this bit, all port resets will remain set to one until explicitly cleared to zero through the individual port control clear registers. Refer to the `Port Control Set` register description on page 60 for more information.
- **Bit [30]**: MSI Acknowledge (W). Writing 1 to this bit acknowledges a Message Signaled Interrupt (MSI) and permits generation of another MSI. This bit is cleared immediately after the acknowledgement is recognized by the control logic, hence the bit will always be read as a 0. If all interrupt conditions are removed following an MSI, it is not necessary to assert this Acknowledge; another MSI will be generated when an interrupt condition occurs.
- **Bit [29]**: I^2C Int Enable (R/W). This bit, when set to 1, allows assertion of an interrupt when the I^2C interrupt is asserted. When set to 0, the interrupt is masked.
- **Bit [28:25,23:2]**: Reserved (R). These bits are reserved and will return zeroes when read.
- **Bit [24]**: 3Gb/s Capable (R). This bit is hardwired to 1.
- **Bit [1:0]**: Port Interrupt Enable (R/W). These bits, when set to 1, allow assertion of an interrupt of the corresponding ports. When set to 0, the corresponding port interrupts are masked.

Global Interrupt Status

Address Offset: 44_H

Access Type: Read/Write 1 Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved		I²C Interrupt	Reserved																											Port 1 Interrupt	Port 0 Interrupt

This register is used to determine the status of various chip functions.

- **Bit [31:30]**: Reserved (R). This bit field is reserved and returns zeroes when read.
- **Bit [29]**: I^2C Interrupt (R). This bit indicates that the I^2C Interrupt is pending. The interrupt source must be cleared (in the `I2C Status` register) for this bit to be *0*. This bit will not report interrupt sources that are not enabled in the `I2C Control` register.
- **Bit [28:2]**: Reserved (R). This bit field is reserved and returns zeroes when read.
- **Bit [1:0]**: Port Interrupt Status (R/W1C). These bits, when set to 1, indicate that the corresponding port has an interrupt condition pending. Writing a *1* to any of these bits clears the corresponding Command Completion Interrupt Status, but not other interrupt sources.

PHY Configuration

Address Offset: 48_H

Access Type: Read/Write

Reset Value: 0x0000_2C40

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																PHY Config															

This register is reset to 0x00002C40.

- **Bit [31:16]**: Reserved (R/W) – This bit field is reserved and must be always be zeros.
- **Bit [15:04]**: PHY Configuration (R/W) – These bits should not be changed from their defaults as erratic operation may result.
- **Bit [03:00]**: SATA SSC (R/W) – If this bit field is set to 1111b, all channel Tx outputs Spread Spectrum Clocking.

BIST Control

Address Offset: 50_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
BISTenable	BISTpatsel	Reserved												BISTcompsel		Reserved												BISTrun			

This register is used to control Data Loopback BIST.

- **Bit [31]**: BISTenable (R/W) – This bit enables the data paths for running data loopback BIST.
- **Bit [30]**: BISTpatsel (R/W) – This bit selects whether a repeating pattern (supplied from the `BIST Pattern` register) or a pseudorandom pattern is used for running data loopback BIST. Setting the bit to *1* selects the repeating pattern.
- **Bit [29:18]**: Reserved (R/W). These bits are reserved and must be written as zeros.
- **Bit [17:16]**: BISTcompsel (R/W). This bit field selects the port from which loopback data is selected for pattern comparison.
- **Bit [15:04]**: Reserved (R/W). These bits are reserved and must be written as zeros.
- **Bit [03:00]**: BISTrun (R/W). This bit field selects the port(s) that transmit loopback data.

BIST Pattern

Address Offset: 54_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
BIST Pattern																															

This register contains the 32-bit fixed pattern that is repeatedly transmitted in data loopback when the BISTpatsel bit (bit 30) of the `BIST Control` register is set to *1*.

BIST Status

Address Offset: 58_H

Access Type: Read

Reset Value: 0x8000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
BISTgood	Reserved																			BISTerrcnt											

- **Bit [31]**: BISTgood (R) – This bit indicates that all comparisons have been good since initiating data loopback BIST. This bit is initialized to *1* when the BISTenable bit is zero in the `BIST Control` register.
- **Bit [30:12]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [11:00]**: BISTerrcnt (R). This bit field indicates the number of comparisons that have been in error since initiation of data loopback BIST. This counter is a saturating counter (it stops counting at $0FFF_H$). This counter is cleared when the BISTenable bit is 0 in the `BIST Control` register.

I²C Control

Address Offset: 60_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																Fast Mode	Unit Reset	Addr Det Int En	Arb Loss Det Int En	STOP Det Int En	Bus Error Int En	Rx Full Int En	Tx MT Int En	Gen Call Disable	Unit Enable	SCL Enable	Master Abort	Transfer Byte	ACK/NACK Ctrl	STOP	START

This register contains bits to control the operation of the I²C interface.

- **Bit [31:16]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [15]**: Fast Mode (R/W). This bit, when set, enables 400 kbit/s operation (instead of 100 kbit/s)
- **Bit [14]**: Unit Reset (R/W). This bit, when set, resets the I²C controller.
- **Bit [13]**: Addr Det Int En (R/W). Slave Address Detected Interrupt Enable – This bit enables interrupt generation when the Slave Address or the General Call Address is detected.
- **Bit [12]**: Arb Loss Det Int En (R/W). Arbitration Loss Detected Interrupt Enable – This bit enables interrupt generation upon losing arbitration in Master Mode. If this bit is not set, and arbitration is lost during transmission of the control byte, the arbitration and transmission will be automatically repeated.
- **Bit [11]**: STOP Det Int En (R/W). Slave STOP Detected Interrupt Enable – This bit enables interrupt generation when a STOP condition while in Slave Mode is detected.
- **Bit [10]**: Bus Error Int En (R/W). Bus Error Interrupt Enable – This bit enables interrupt generation when an I²C error is detected.
- **Bit [09]**: Rx Full Int En (R/W). Receive Buffer Full Interrupt Enable – This bit enables interrupt generation when a byte has been received and is in the `I2C Data Buffer` register.
- **Bit [08]**: Tx MT Int En (R/W). Transmit Buffer Empty Interrupt Enable – This bit enables interrupt generation after the data byte in the `I2C Data Buffer` register is transmitted.
- **Bit [07]**: Gen Call Disable (R/W). General Call Disable – This bit disables detection of a General Call Address.
- **Bit [06]**: Unit Enable (R/W). This bit must be set to enable any I²C controller operations.
- **Bit [05]**: SCL Enable (R/W). This bit must be set to enable the I²C Clock output for Master Mode operations.
- **Bit [04]**: Master Abort (R/W). This bit may be set along with STOP to send a STOP without first transferring a byte.
- **Bit [03]**: Transfer Byte (R/W). Set to initiate the transfer to/from the Data Buffer.
- **Bit [02]**: ACK/NACK Ctrl (R/W). Set to send a NACK instead of an ACK after receiving a data byte. An ACK is sent in response to slave address detection regardless of this bit. Sending a NACK in Slave Mode will set the Bus Error status.
- **Bit [01]**: STOP (R/W). Set to send a STOP following transfer of a byte (using Transfer Byte) or immediately (using Master Abort).
- **Bit [00]**: START (R/W). Set to send a START prior to transfer of a byte (using Transfer Byte).

I²C Status

Address Offset: 64_H

Access Type: Read/Write 1 to Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																					Bus Error Det	Slave Addr Det	Gen Call Det	Rx Full	Tx MT	Arb Loss Det	STOP Det	Bus Busy	Unit Busy	ACK/NACK Status	Read/Write

This register contains bits to report the status of the I²C interface.

- **Bit [31:11]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [10]**: Bus Error Det (R/W1C). Bus Error Detected – This bit reports the detection of an illegal NACK either received in Master Mode or sent in Slave Mode. This bit is not set by the legal NACK at the end of a transfer that signals the end to the slave.
- **Bit [09]**: Slave Addr Det (R/W1C). Slave Address Detected – This bit reports the detection of the Slave Address or of the General Call Address (when enabled). The byte received containing the address is placed into the `I2C Data Buffer` register.
- **Bit [08]**: Gen Call Det (R/W1C). General Call Address Detected – This bit reports the detection of the General Call Address.
- **Bit [07]**: Rx Full (R/W1C). I²C Data Buffer Receive Full – A data byte has been received and an ACK or NACK has been sent. This bit is set on the rising edge of the I²C clock allowing for some minimal (one bit) time for an interrupt before insertion of wait states.
- **Bit [06]**: Tx MT (R). I²C Data Buffer Transmit Empty – A data byte has been transmitted and an ACK or NACK has been received. This bit is set on the rising edge of the I²C clock allowing for some minimal (one bit) time for an interrupt before insertion of wait states.
- **Bit [05]**: Arb Loss Det (R/W1C). Arbitration Loss Detected – This bit indicates that arbitration has been lost. If the corresponding interrupt is not enabled, and the arbitration loss is detected during selection, i.e., during the control byte transmission, the arbitration will be repeated automatically; if arbitration is lost later during data transmission, no repeat is possible and is not attempted.
- **Bit [04]**: STOP Det (R/W1C). Slave STOP Detected.
- **Bit [03]**: Bus Busy (R). Indicates that the I²C bus is busy because of activity other than that generated by the I²C controller.
- **Bit [02]**: Unit Busy (R). Indicates that the I²C controller is busy.
- **Bit [01]**: ACK/NACK Status (R). Indicates status of last ACK or NACK sent or received.
- **Bit [00]**: Read/Write (R). Indicates state of the R/W# bit of the I²C slave address (either the one sent in Master Mode or the one received in Slave Mode).

I²C Slave Address

Address Offset: 68_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																									Slave Address						

This register contains the 7-bit Slave Address to which the I²C controller will respond.

I²C Data Buffer

Address Offset: $6C_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																								Data Buffer							

This register contains the buffer for I²C send/receive data.

Flash Address

Address Offset: 70_H

Access Type: Read/Write

Reset Value: 0xXXXX_XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
GPIO Enable	Reserved 0b0001				Mem Present	Mem Access Start	Mem Access Type	Reserved					Memory Address																		

This register is the address and command/status register for the flash memory interface. The register bits are defined as follows:

- **Bit [31]**: GPIO Enable (R/W). This bit, when set to 1, enables the use of the flash Data pins for General Purpose I/O.
- **Bit [30:27]**: Reserved (R). This bit field is reserved and returns 0001 on a read.
- **Bit [26]**: Mem Present (R) – Memory Present. This bit set indicates that the auto-initialization signature was read correctly from the flash Memory.
- **Bit [25]**: Mem Access Start (R/W) – Memory Access Start. This bit is set to initiate an operation to flash memory. This bit is self-clearing when the operation is complete.
- **Bit [24]**: Mem Access Type (R/W) – Memory Access Type. This bit is set to define a read operation from flash memory. This bit is cleared to define a write operation to flash memory.
- **Bit [23:19]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [18:00]**: Memory Address (R/W). This bit field is programmed with the address for a flash memory read or write access.

Flash Memory Data / GPIO Control

Address Offset: 74_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved								GPIO Control								Transition Detect								Memory Data							

This register contains the GPIO data/control fields and the flash memory data register.

- **Bit [31:24]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [23:16]**: GPIO Control (R/W). The bits of this field are written to control the output type for corresponding flash data lines; if a bit is a *1* the corresponding output is an open drain output (only driven LOW); if a bit is *0* the

corresponding output is always driven. To use a GPIO pin as an input, the control bit must be set to *1* (open-drain output) and the data bit must be set to *1* (undriven).

- **Bit [15:08]**: Transition Detect (R/C). The bits of this field report signal transition detection on the corresponding Flash data input; reading the register resets the transition detect bits.
- **Bit [07:00]**: Memory Data (R/W) – Flash Memory Data. This bit field is used for flash write data on a write operation, and returns the flash read data on a read operation. For GPIO, this field is used to write the GPIO output register and to read the GPIO input signals.

Internal Register Space – Base Address 1

These registers are 32 bits wide and are the Port Registers and LRAM of the SiI3132 controller. Access to these registers is through the PCI Memory space. Register descriptions that follow specify the address offset for Port 0; Port 1 registers are at the Port 0 offset plus 2000_H.

Table 31. SiI3132 Internal Register Space – Base Address 1

Address Offset	Register Name
000_H–$F7F_H$	Port 0 LRAM Slots
$F80_H$–FFF_H	Port Multiplier Device Status/QActive Registers
1000_H	Write: Port 0 Control Set / Read: Port 0 Status
1004_H	Write: Port 0 Control Clear
1008_H	Port 0 Interrupt Status
$100C_H$	Reserved
1010_H	Port 0 Interrupt Enable Set
1014_H	Port 0 Interrupt Enable Clear
1018_H	Reserved
$101C_H$	32-bit Activation Upper Address
1020_H	Port 0 Command Execution FIFO
1024_H	Port 0 Command Error
1028_H	Port 0 FIS Configuration
$102C_H$	Port 0 PCI Exp Request FIFO Threshold
1030_H–$103F_H$	Reserved
1040_H	Port 0 8B/10B Decode Error Counter
1044_H	Port 0 CRC Error Counter
1048_H	Port 0 Handshake Error Counter
$104C_H$	Reserved
1050_H	Port PHY Configuration
1054_H–$17FF_H$	Reserved
1800_H	Port 0 Slot Status
1804_H–$1BFF_H$	Reserved
$1C00_H$–$1CF7_H$	Command Activation Registers
$1CF8_H$–$1E03F_H$	Reserved
$1E04_H$	Port Context Register
$1E08_H$–$1EFF_H$	Reserved
$1F00_H$	Port 0 SControl
$1F04_H$	Port 0 SStatus
$1F08_H$	Port 0 SError
$1F0C_H$	Port 0 SActive (indirect location)
$1F10_H$	Port 0 SNotification
$1F14_H$–$1FFF_H$	Reserved
2000_H–$3FFF_H$	Port 1 Registers mapped as above

Port LRAM

Address Offset: 000_H-FFF_H

Access Type: Read/Write

Reset Value: indeterminate

The Port LRAM consists of 31 Slots of 128 bytes each and a 32nd slot used to hold 16 Port Multiplier Device Specific Registers.

Table 32. Port LRAM layout

Address Offset	Description
000_H–$07F_H$	Slot 0
080_H–$0FF_H$	Slot 1
100_H–$17F_H$	Slot 2
180_H–EFF_H	Slots 3–29
$F00_H$–$F7F_H$	Slot 30
$F80_H$–$F83_H$	Port Multiplier Device 0 Status Register
$F84_H$–$F87_H$	Port Multiplier Device 0 QActive Register
$F88_H$–$F8B_H$	Port Multiplier Device 1 Status Register
$F8C_H$–$F8F_H$	Port Multiplier Device 1 QActive Register
$F90_H$–$FF7_H$	Port Multiplier Device Registers for Devices 2–14
$FF8_H$–FFB_H	Port Multiplier Device 15 Status Register
FFC_H–FFF_H	Port Multiplier Device 15 QActive Register

Table 33. Port LRAM Slot layout

Address Offset	Description	
000_H–$01F_H$	Current FIS and Control	Port Request Block (PRB)
020_H–$02F_H$	Scatter/Gather Entry 0 or ATAPI command packet	
030_H–$03F_H$	Scatter/Gather Entry 1	
040_H–047_H	Command Activation Register (Actual)	
040_H–$07F_H$	Scatter/Gather Table	
$1C00_H$–$1C07_H$	Command Activation Register (Shadow)	

Note: A Port LRAM Slot is 128 bytes used to define Serial-ATA commands. The addresses shown above are for slot 0.

Port Slot Status

Address Offset: 1800_H

Access Type: Read

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Attention	Slot Status																														

This register provides the status for the 31 Command Slots for the Serial-ATA port. This register also appears along with the `Global Port Slot Status` register of the other port in Global register space. Reading this register clears the Command Completion Status for the port if the Interrupt No Clear on Read bit (bit 3) of the `Port Control Set` register is 0. The register bits are defined as follows:

- **Bit [31]**: Attention (R) – This bit indicates that something occurred in the port that requires the attention of the host. Other port registers must be examined to determine the origin of the error. This bit is the logical OR of the masked interrupt conditions reported in the `Port Interrupt Status` register.
- **Bit [30:0]**: Slot Status (R) – These bits are the Active status bits corresponding to Slot numbers 30 to 0. The Active status bit is set when a command is transferred to the Slot RAM.

Port Control Set

Address Offset: Set: 1000_H

Access Type: Write One To Set

Reset Value: N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved						OOB Bypass	Reserved									LED On	Auto Interlock Accept	PM Enable	Interlock Accept	Interlock Reject t	32–bit Activation	Scramble Disable	CONT Disable	Transmit BIST	Resume	Packet Length	LED Disable	Interrupt NCoR	Port Initialize	Device Reset	Port Reset

This register is used to direct various port operations. A *1* written to a bit position sets that bit in the control register.

- **Bit [31:26,24:16]**: Reserved (R). These bits are reserved.
- **Bit [25]**: OOB Bypass (W1S). If this bit is set, the Link will bypass the OOB initialization sequence following a reset. This bit is reset by Global Reset, not by Port Reset.
- **Bit [15]**: LED On (W1S). This bit turns on the LED Port Activity indicator regardless of the state of LED Disable (bit 4).
- **Bit [14]**: Auto Interlock Accept (W1S). When this bit is set the link will accept any interlocked FIS reception. The link will transmit R_OK in response to the received FIS.
- **Bit [13]**: PM Enable (W1S). This bit enables Port Multiplier support.
- **Bit [12]**: Interlock Accept (W1S). This bit is used to signal the link to accept an interlocked FIS reception. The link will transmit R_OK in response to the received FIS. This bit is self-clearing.
- **Bit [11]**: Interlock Reject (W1S). This bit is used to signal the link to reject an interlocked FIS reception. The link will transmit R_ERR in response to the received FIS. This bit is self-clearing.
- **Bit [10]**: 32-bit Activation (W1S). When this bit is set to 1, a write to the low 32 bits of a Command Activation register causes the contents of the 32-bit Activation Upper Address register to be written to the upper 32 bits of the Command Activation register and triggers command execution. When this bit is 0, a write to the upper 32 bits or all 64 bits of a Command Activation register is required to trigger command execution. This bit is set for environments that do not address more than 2^{32} bytes of host memory.
- **Bit [9]**: Scrambler Disable (W1S). When this bit is set to 1, the Link scrambler operation is disabled.
- **Bit [8]**: CONT Disable (W1S). When this bit is set to 1, the Link will not generate a CONT following repeated primitives.
- **Bit [7]**: Transmit BIST (W1S). This bit causes transmission of a BIST FIS.
- **Bit [6]**: Resume (W1S). This bit is used to enable processing of outstanding commands to additional devices connected to a port multiplier after a command error has occurred. When this bit is set, the internal BUSY status is set for the device corresponding to the value of the current Port Multiplier Port. This prevents additional commands from being sent to the device in error until a Port Initialize operation is performed.
- **Bit [5]**: Packet Length (W1S). This bit directs the length of the packet command to be sent for commands with packet protocol. When this bit is *0*, a 12-byte packet is sent. When this bit is *1*, a 16-byte packet is sent. This bit should be set to the same value as derived from word 0 of the data returned by the `identify packet` command.
- **Bit [4]**: LED Disable (R/W). This bit disables the operation of the LED Port Activity indicator.
- **Bit [3]**: Interrupt No Clear on Read (W1S). When this bit is set to 1, a command completion interrupt may be cleared only by writing a one to the Command Completion bit in the `Port Interrupt Status` register. When this bit is 0, reading the `Port Slot Status` register may also be used to clear the Command Completion interrupt.
- **Bit [2]**: Port Initialize (W1S). Setting this bit to *1* causes all commands to be flushed from the port and all command execution parameters to be set to an initialized state. Setting this bit to *1* causes the Port Ready bit in the `Port Status` register to be cleared to *0*. When the initialization procedure is complete, the Port Ready bit is set to *1*. This bit is self-clearing and is cleared upon execution by the port.

- **Bit [1]**: Device Reset (W1S). Setting this bit to *1* causes all commands to be flushed from the port and all command execution parameters to be set to an initialized state. Setting this bit to *1* causes the Port Ready bit in the Port Status register to be cleared to *0*. The port will generate the COMRESET primitive on the Serial ATA bus. When the out of band sequence and initialization procedure is complete, the Port Ready bit is set to 1. This bit is self-clearing and is cleared upon execution by the port.
- **Bit [0]**: Port Reset (W1S). Setting this bit to *1* causes the port to be held in a reset state. No commands are executed while in this state. All port registers and functions are reset to their initial state, except as noted below. All commands are flushed from the port and all command execution parameters are set to an initialized state. Setting this bit to *1* causes the Port Ready bit in the Port Status register to be cleared to *0*. Upon setting this bit to *0* from an asserted state, the port generates the COMRESET primitive on the Serial ATA bus. When the out of band sequence and initialization procedure is complete, the Port Ready bit is set to *1*. This bit is set to *1* by the Global reset, which is set by a PCI reset, and remains set until cleared by the host by writing a *1* to bit 0 of the Port Control Clear register.
  The register bits that are not initialized by the Port Reset are:
  - OOB Bypass (bit 25) in Port Control (this register)
  - Port PHY Configuration register (all bits)

Port Status

Address Offset: 1000_H

Access Type: Read

Reset Value: 0x001F_0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Port Ready	Reserved					OOB Bypass	Reserved				Active Slot					LED On	Auto Interlock Accept	PM Enable	Interlock Accept	Interlock Reject	32-bit Activation	Scramble Disable	CONT Disable	Transmit BIST	Resume	Packet Length	LED Disable	Interrupt NCoR	Port Initialize	Device Reset	Port Reset

This register is used to determine the status of various port functions.

- **Bit [31]**: Port Ready (R). This bit reports the Port Ready status. The transition from *0* to *1* of this bit generates the Port Ready Interrupt Status (bit 18/2 of the Port Interrupt Status register).
- **Bit [30:26,24:21]**: Reserved (R). These bits are reserved.
- **Bit [20:16]**: Active Slot (R). This bit field contains the slot number of the command currently being executed. When a command error occurs, this bit field indicates the slot containing the command in error.
- **Bit [25,15:0]**: These bits reflect the current state of the corresponding bits in the Port Control Set register. Refer to that register for a complete description.

Port Control Clear

Address Offset: 1004_H

Access Type: Write 1 To Clear

Reset Value: N/A

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved						OOB Bypass	Reserved									LED On	Auto Interlock Accept	PM Enable	Reserved		32-bit Activation	Scramble Disable	CONT Disable	Transmit BIST	Resume	Packet Length	LED Disable	Interrupt NCoR	Reserved		Port Reset

This register is used to direct various port operations. A *1* written to a bit position clears that bit in the Port Control Set register.

- **Bit [31:26,24:16,12:11,2:1]**: Reserved (R). These bits are reserved.
- **Bit [25, 15:13,10:3,0]**: (W1C) Writing a *1* to these bits clears the associated bit position of the Port Control Set register. Refer to that register for bit descriptions.

Port Interrupt Status

Address Offset: 1008_H

Access Type: Read/Write 1 Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved				SDB Notify	Hshk Error Thresh	CRC Error Thresh	8b/10 Error Thresh	DevExchg	UnrecFIS	Comwake	PhyRdyChg	PM Change	Port Ready	Command Error	Cmd Completion	Reserved				SDB Notify	Hshk Error Thresh	CRC Error Thresh	8b/10 Error Thresh	DevExchg	UnrecFIS	Comwake	PhyRdyChg	PM Change	Port Ready	Command Error	Cmd Completion	

This register is used to report the interrupt status. The status bits in the upper half of the register report the described condition. The status bits in the lower half of the register are masked by the corresponding interrupt enable bits or by the setting in the corresponding threshold registers. Writing a *1* to either interrupt status bit clears it.

- **Bit [31:28,15:12]**: Reserved (R). These bits are reserved.
- **Bit [27/11]**: SDB Notify (W1C). This bit indicates that a Set Device Bits FIS was received with the N-bit (bit 15 of first Dword) set to *1*.
- **Bit [26/10]**: Handshake Error Threshold (W1C). This bit indicates that the Handshake error count is equal to or greater than the Handshake error threshold. Bit 10 is masked if the Port Handshake Error Counter Threshold register contains a zero threshold setting. When a *1* is written to this bit, both the status bit and the Handshake Error Counter are cleared.
- **Bit [25/9]**: CRC Error Threshold (W1C). This bit indicates that the CRC error count is equal to or greater than the CRC error threshold. Bit 9 is masked if the Port CRC Error Counter Threshold register contains a zero threshold setting. When a *1* is written to this bit, both the status bit and the CRC Error Counter are cleared.
- **Bit [24/8]**: 8b/10b Decode Error Threshold (W1C). This bit indicates that the 8b/10b Decode error count is equal to or greater than the 8b/10b Decode error threshold. Bit 8 is masked if the Port 8B/10B Decode Error Counter Threshold register contains a zero threshold setting. When a *1* is written to this bit, both the status bit and the 8b/10b Decode Error Counter are cleared.
- **Bit [23/7]**: DevExchg (Device Exchanged) (W1C) – This bit is the X bit in the DIAG field of the SError register. It may be cleared by writing a *1* to the corresponding bit of either register.
- **Bit [22/6]**: UnrecFIS (Unrecognized FIS Type) (W1C) – This bit is the F bit in the DIAG field of the SError register. It may be cleared by writing a corresponding one bit to either register.
- **Bit [21/5]**: ComWake (W1C) – This bit is the W bit in the DIAG field of the SError register. It may be cleared by writing a corresponding one bit to either register.
- **Bit [20/4]**: PhyRdyChg (W1C) – This bit is the N bit in the DIAG field of the SError register. It may be cleared by writing a corresponding one bit to either register.
- **Bit [19/3]**: PM Change (W1C). This bit indicates that a change has occurred in the power management state.
- **Bit [18/2]**: Port Ready (W1C). This bit indicates that the port has become ready to accept and execute commands. This status indicates that Port Ready (bit 31 in the Port Status register) has made a *0* to *1* transition. Clearing this status does not change the Port Ready bit in the Port Status register and this status is not set subsequently until the Port Ready bit changes state.
- **Bit [17/1]**: Command Error (W1C). This bit indicates that an error occurred during command execution. The error type can be determined via the Port Command Error register.
- **Bit [16/0]**: Command Completion (W1C). This bit indicates that one or more commands have completed execution.

Port Interrupt Enable Set / Port Interrupt Enable Clear

Address Offset: 1010_H / 1014_H

Access Type: Read/Write 1 Set/Write 1 Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Interrupt Steering																				SDB Notify	Reserved			DevExchg	UnrecFIS	Comwake	PhyRdyChg	PM Change	Port Ready	Command Error	Cmd Completion

The Interrupt Enable register is controlled by these registers. Writing to the `Port Interrupt Enable Set` register sets the Interrupt Enable bits. The enable bit is set for each corresponding bit to which 1 is written. Writing to the `Port Interrupt Enable Clear` register clears the Interrupt Enable bits; the enable bit is cleared for each corresponding bit to which a *1* is written. The Interrupt Enable register may be read at either address offset.

Bits 8, 9, and 10 do not have an enable bit; the corresponding interrupts are enabled by corresponding threshold registers.

- **Bit [31:30]**: Interrupt Steering (R/W). This bit field specifies which one of the four interrupts is to be used for interrupts from this port. INTA is selected by 00_B; INTB by 01_B; INTC by 10_B; and INTD by 11_B.
- **Bit [29:12,10:8]**: Reserved (R). These bits are reserved and return 0s on a read.
- **Bit [11,7:0]**: Interrupt Enables (R/W1S/W1C). These bits are the interrupt enables for the corresponding bits of the Interrupt Status register.

32-bit Activation Upper Address

Address Offset: $101C_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Upper Address																															

This register contains the 32-bit value written to the upper half of the Command Activation register when the lower half of that register is written and the 32-bit Activation control bit (bit 10) is set in the `Port Control Set` register.

Port Command Execution FIFO

Address Offset: 1020_H

Access Type: Read/Write

Reset Value: 0x0000_00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																											Execution Slot Number				

When written, this register causes the supplied slot number to be pushed into the tail of the command execution FIFO. A valid PRB must be populated in the associated slot in port LRAM. When read, this register supplies the entry at the head of the command execution FIFO. The FIFO is not popped as a result of a read operation.

Port Command Error

Address Offset: 1024_H

Access Type: Read

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Error Code																															

This register contains the error type resulting from a command error. Table 34 lists the error codes, error names, and error descriptions.

Table 34. Command Error Codes

Error Name	Code	Description
DEVICEERROR	1	The ERR bit was set in a register - device to host FIS received from the device. The task file registers are written back to PRB slot for host scrutiny.
SDBERROR	2	The ERR bit was set in a set device bits FIS received from the device.
DATAFISERROR	3	The controller detected an error during command execution that was not reported by the device upon command completion.
SENDFISERROR	4	The controller was unable to send the Initial command FIS for a command. This can occur if a low-level link error occurs during command transmission.
INCONSISTENTSTATE	5	The controller detected an inconsistency in protocol. Any departure from standard Serial ATA protocol that causes indecision in the internal sequencers will cause this error.
DIRECTIONERROR	6	A Data FIS was received when a write data protocol was specified or a DMA Activate FIS was received when a read data protocol was specified.
UNDERRUNERROR	7	While transferring data from the controller to a device, the end of the Scatter Gather list was encountered before the entire transfer was completed. The device is requesting additional data but there is no Scatter Gather Entry to define the source of data.
OVERRUNERROR	8	While transferring data from a device to the controller, the end of the Scatter Gather list was encountered before the entire transfer was completed. Data was received from the device but there is no Scatter Gather Entry to define where the data should be deposited.
LLOVERRUNERROR	9	Low level overrun error. While transferring data from a device to the controller, the transfer count received in a PIO Setup or DMA Setup FIS has been satisfied and there is additional data being received from the device.
PACKETPROTOCOLERROR	11	During the first PIO setup of Packet command, the data direction bit was invalid, indicating a transfer from device to host.
PLDSGTERRORBOUNDARY	16	A requested Scatter Gather Table not aligned on a quadword boundary. All addresses defining Scatter Gather Tables must be quadword aligned. Bits[2:0] must be zeroes.
PLDSGTERRORTARGETABORT	17	A PCI Target Abort occurred while the controller was fetching a Scatter Gather Table from host memory.
PLDSGTERRORMASTERABORT	18	A PCI Master Abort occurred while the controller was fetching a Scatter Gather Table from host memory.
PLDSGTERRORPCIPERR	19	A PCI Parity Error occurred while the controller was fetching a Scatter Gather Table from host memory.
PLDCMDERRORBOUNDARY	24	The address of a PRB written to a Command Activation register was not aligned on a quadword boundary. All PRB addresses must be quadword aligned. Bits[2:0] must be zeroes.
PLDCMDERRORTARGETABORT	25	A PCI Target Abort occurred while the controller was fetching a Port Request Block (PRB) from host memory.
PLDCMDERRORMASTERABORT	26	A PCI Master Abort occurred while the controller was fetching a Port Request Block (PRB) from host memory.
PLDCMDERRORPCIPERR	27	A PCI Parity Error occurred while the controller was fetching a Port Request Block (PRB) from host memory.
PSDERRORTARGETABORT	33	A PCI Target Abort occurred while data transfer was underway between the controller and host memory.
PSDERRORMASTERABORT	34	A PCI Master Abort occurred while data transfer was underway between the controller and host memory.
PSDERRORPCIPERR	35	A PCI Parity Error occurred while data transfer was underway between the controller and host memory.
SENDSERVICEERROR	36	A FIS was received while attempting to transmit a Service FIS. Following the receipt of a Set Device Bits FIS containing a service request, the device sent another FIS before allowing the host to send a Service FIS.

Port FIS Configuration

Address Offset: 1028_H

Access Type: Read/Write

Reset Value: 0x1000_1555

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved		FIS27cfg		FIS34cfg		FIS39cfg		FIS41cfg		FIS46cfg		FIS58cfg		FIS5Fcfg		FISA1cfg		FISA6cfg		FISB8cfg		FISBFcfg		FISC7cfg		FISD4cfg		FISD9cfg		FISOcfg	

This register contains bits for controlling Serial ATA FIS reception. For each possible FIS type, a 2-bit code defines the desired reception behavior as follows:

- 00 – Accept FIS without interlock.
- 01 – Reject FIS without interlock
- 10 – Interlock FIS. Receive FIS into slot reserved for interlocked FIS reception. If no slot has been reserved, reject the FIS.
- 11 – Reserved.

Bit[1:0] (FISOcfg) defines the 2-bit code for all other FIS types not defined by bits [29:2].

Table 35 defines the default behavior of FIS configuration.

Table 35. Default FIS Configurations

FIS Code	FIS Name	Configuration Bits		Default Action
		Signals	Default Value	
27h	Register (Host to Device)	fis27cfg[1:0]	01b	reject FIS without interlock
34h	Register (Device to Host)	fis34cfg[1:0]	00b	accept FIS without interlock
39h	DMA Activate	fis39cfg[1:0]	00b	accept FIS without interlock
41h	DMA Setup	fis41cfg[1:0]	00b	accept FIS without interlock
46h	Data	fis46cfg[1:0]	00b	accept FIS without interlock
58h	BIST Activate	fis58cfg[1:0]	00b	accept far-end retimed loopback, reject any other
5Fh	PIO Setup	fis5Fcfg[1:0]	00b	accept FIS without interlock
A1h	Set Device Bits	fisa1cfg[1:0]	00b	accept FIS without interlock
A6h	reserved	fisa6cfg[1:0]	01b	reject FIS without interlock
B8h	reserved	fisb8cfg[1:0]	01b	reject FIS without interlock
BFh	reserved	fisbFcfg[1:0]	01b	reject FIS without interlock
C7h	reserved	fisc7cfg[1:0]	01b	reject FIS without interlock
D4h	reserved	fisd4cfg[1:0]	01b	reject FIS without interlock
D9h	reserved	fisd9cfg[1:0]	01b	reject FIS without interlock
Others	reserved	fisocfg[1:0]	01b	reject FIS without interlock

Port PCI Express Request FIFO Threshold

Address Offset: $102C_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved					PCI Exp Write Request Threshold								Reserved								PCI Exp Read Request Threshold								Reserved		

This register contains threshold levels at which the PCI Express master state machine requests the PCI Express bus relative to the amount of data or free space in the data FIFO. The data FIFO capacity is 2 kB (256 Qwords). When writing to host memory (reading data from a device), the PCI Express Write Request Threshold is compared to the amount of data in the data FIFO. When the FIFO contents exceed the threshold value, a request is issued to write the data to host memory, emptying the contents of the data FIFO. Before the SiI3132 controller loads its FIFO with data from the host memory that it intends to write to a device, it compares the PCI Express Read Request Threshold register to the amount of free space in the data FIFO. When the free space in the FIFO exceeds the threshold value, the controller proceeds to fill the FIFO from host memory.

- **Bit [31:27]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [26:19]:** PCI Exp Write Request Threshold (R/W). This field defines the number of Qwords that must be in the data FIFO before issuing a PCI Express request. A value of 0 causes a request if the FIFO contains any amount of data.
- **Bit [18:16]**: Reserved (R). This bit field is reserved and returns zeros on a read. This field is defined so that the host may write a byte count value into the threshold register.
- **Bit [15:11]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [10:3]:** PCI Exp Read Request Threshold (R/W). This field defines the number of Qwords that must be available in the data FIFO before issuing a PCI Express request. A value of 0 causes a request if the FIFO contains any free space and the DMA is active.
- **Bit [2:0]**: Reserved (R). This bit field is reserved and returns zeros on a read. This field is defined so that the host may write a byte count value into the threshold register.

Port 8B/10B Decode Error Counter

Address Offset: 1040_H

Access Type: Read/Write/Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
8B/10B Decode Error Threshold																8B/10B Decode Error Counter															

This register counts the number of 8B/10B Decode Errors that have occurred since last cleared.

- **Bit [31:16]**: 8B/10B Decode Error Threshold (R/W). This bit field defines the count at which an interrupt is asserted. When the count in bits 15:0 is equal to this value, an 8B/10B interrupt will be latched. A threshold value of 0 disables interrupt assertion and masks the corresponding interrupt status bit in the `Port Interrupt Status` register.
- **Bit [15:0]**: 8B/10B Decode Error Count (R/WC). This bit field represents the count of 8B/10B errors that have occurred since this register was last written. Any write to this register field clears both the counter and the interrupt condition. Clearing the interrupt status bit also clears the counter. The count will not overflow. Once this register reaches its maximum count, it retains that count until cleared to 0 by a write operation.

Port CRC Error Counter

Address Offset: 1044_H

Access Type: Read/Write/Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
CRC Error Counter Threshold																CRC Error Counter															

This register counts the number of Serial ATA CRC Errors that have occurred since it was cleared the last time.

- **Bit [31:16]**: Serial ATA CRC Error Threshold (R/W). This bit field defines the count at which an interrupt will be asserted. When the count in bits 15:0 is equal to this value, a Serial ATA CRC interrupt is latched. A threshold value of 0 disables interrupt assertion and masks the corresponding interrupt status bit in the `Port Interrupt Status` register.

- **Bit [15:0]**: Serial ATA CRC Error Count (R/WC). This bit field represents the count of Serial ATA CRC errors that have occurred since this register was written the last time. Any write to this register will clear both the counter and the interrupt condition. Clearing the interrupt status bit will also clear the counter and the count will not overflow. Once this register reaches its maximum count, it will retain that count until cleared to *0* by a write operation.

Port Handshake Error Counter

Address Offset: 1048_H

Access Type: Read/Write/Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Serial ATA Handshake Error Counter Threshold																Serial ATA Handshake Error Counter															

This register counts the number of Serial ATA Handshake Errors that have occurred since it was cleared the last time.

- **Bit [31:16]**: Serial ATA Handshake Error Threshold (R/W). This bit field defines the count at which an interrupt is asserted. When the count in bits 15:0 is equal to this value, a Serial ATA Handshake interrupt will be latched. A threshold value of zero disables interrupt assertion and masks the corresponding interrupt status bit in the `Port Interrupt Status` register.
- **Bit [15:0]**: Serial ATA Handshake Error Count (R/WC). This bit field represents the count of Serial ATA Handshake errors that have occurred since it was written the last time.. Any write to this register will clear both the counter and the interrupt condition. Clearing the interrupt status bit will also clear the counter. The count will not overflow. Once this register reaches its maximum count, it will retain that count until cleared to 0 by a write operation.

Port PHY Configuration

Address Offset: 1050_H

Access Type: Read/Write

Reset Value: 0x0000_020C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
PHY Status																PHY Config															

The `Port PHY Configuration` register is reset by the Global Reset, not by the Port Reset. The reset value is 0x0000020C.

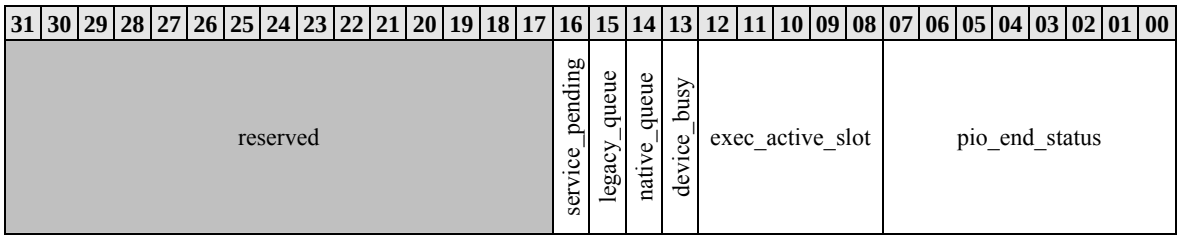
- **Bit[31:16]**: PHY Status (R). These bits report status of the PHY (currently always 0).
- **Bit[15:5]**: PHY Config (R/W). These bits configure the PHY. The value should not be changed because erratic operation may result.
- **Bit[4:0]**: Tx Amplitude (R/W) These bits set the nominal output swing for the Transmitter. The amplitude is increased by 50 mV by an increment of the value.

Port Device Status

Table 36 gives the offset to the Port Device Status registers for each of the 16 possible port multiplier ports.

Table 36. Address Offsets to Port Device Status Registers

Address Offset	PM Port	Address Offset	PM Port	Address Offset	PM Port	Address Offset	PM Port
$F80_H$	0	$FA0_H$	4	$FC0_H$	8	$FE0_H$	12
$F88_H$	1	$FA8_H$	5	$FC8_H$	9	$FE8_H$	13
$F90_H$	2	$FB0_H$	6	$FD0_H$	10	$FF0_H$	14
$F98_H$	3	$FB8_H$	7	$FD8_H$	11	$FF8_H$	15



These 16 registers contain information useful for diagnosing behavior of the execution unit. These 16 registers contain Port Multiplier device specific information. Address Offset bits 6 to 3 are the Port Multiplier Port number for the device to which the status bits apply. There is one register for each of 16 possible port multiplier ports. These registers are part of the LRAM.

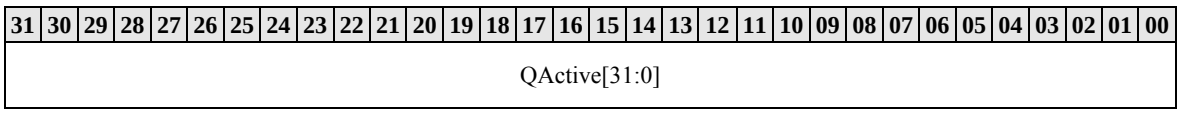
- **Bit [31:17]**: Reserved
- **Bit [16]**: service_pending (R/W). Indicates that a service request has been received from this device and a SERVICE command has not yet been acknowledged.
- **Bit [15]**: legacy_queue (R/W). Indicates that one or more legacy queued commands are outstanding to this device.
- **Bit [14]**: native_queue (R/W). Indicates that one or more native queued commands are outstanding to this device.
- **Bit [13]**: device_busy (R/W). Virtual BSY bit indicating that a command has been issued to the device without receipt of a final register FIS or that a data transfer is in progress.
- **Bit [12:08]**: exec_active_slot (R/W). Contains the slot number of the last command active on this device.
- **Bit [07:00]**: pio_end_status (R/W). Contains the PIO ending status of the last PIO setup command received from this device.

Port Device QActive

Table 37 gives the offset to the Port Device QActive registers for each of the 16 possible port multiplier ports.

Table 37. Address Offsets to Port Device QActive Registers

Address Offset	PM Port	Address Offset	PM Port	Address Offset	PM Port	Address Offset	PM Port
$F84_H$	0	$FA4_H$	4	$FC4_H$	8	$FE4_H$	12
$F8C_H$	1	FAC_H	5	FCC_H	9	FEC_H	13
$F94_H$	2	$FB4_H$	6	$FD4_H$	10	$FF4_H$	14
$F9C_H$	3	FBC_H	7	FDC_H	11	FFC_H	15



These 16 registers contain Port Multiplier device specific status indicating outstanding queued commands in the device. For each bit set to 1, a queued command, legacy or native, is outstanding associated with the slot number corresponding to the bit position. There is one register for each of 16 possible port multiplier ports. Address Offset bits 6 to 3 are the Port Multiplier Port number for the device to which the status bits apply.

- **Bit [31:00]**: Each bit corresponds to a slot number that contains an active outstanding legacy or native queued command.

Port Context

Address Offset: $1E04_H$

Access Type: Read

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																							PM Port				Slot				

- **Bit [31:09]**: Reserved
- **Bit [08:05]**: PM Port (R). This field contains the Port Multiplier port number corresponding to the last FIS transferred (transmit or receive). After a device specific error causes a processing halt, this field contains the PM port corresponding to the device that returned error status.
- **Bit [04:00]**: Slot (R). This field contains the slot number of the last command processed by the execution unit. This slot number does not necessarily correspond to the command in error during error halt conditions. For native queue error recovery, the command slot in error must be determined by issuing a READ LOG EXTENDED to the device to determine the tag number of the command in error.

SControl

Address Offset: $1F00_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved												PMP				SPM				IPM				SPD				DET			

This register is the `SControl` register as defined by the Serial ATA specification.

- **Bit [31:20]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [19:16]**: PMP (R/W). This field identifies the currently selected Port Multiplier port for accessing the SActive register and some bit fields of the Diagnostic registers.
- **Bit [15:12]**: SPM (R/W). This field selects a power management state. A non-zero value written to this field causes initiation of the select power management state. This field self-resets to 0 as soon as action begins to initiate the power management state transition.

Value	Definition
0000	No power management state transition requested
0001	Transition to the Partial power management state initiated
0010	Transition to the Slumber power management state initiated
0100	Transition from a power management state initiated (ComWake asserted)
others	Reserved

- **Bit [11:08]**: IPM (R/W) – This field identifies the interface power management states that may be invoked via the Serial ATA interface power management capabilities.

Value	Definition
0000	No interface power management restrictions (Partial and Slumber modes enabled)
0001	Transitions to the Partial power management state are disabled
0010	Transitions to the Slumber power management state are disabled
0011	Transitions to both the Partial and Slumber power management states are disabled
others	Reserved

- **Bit [07:04]**: SPD (R/W) – This field identifies the highest allowed communication speed the interface is allowed to negotiate.

Value	Definition
0000	No restrictions (default value)
0001	Limit to Generation 1 (1.5 Gbit/s)
0010	Limit to Generation 2 (3.0 Gbit/s)
others	Reserved

- **Bit [03:00]**: DET (R/W) – This field controls host adapter device detection and interface initialization.

Value	Action
0000	No action
0001	COMRESET is periodically generated until another value is written to the field
0100	No action
Others	Reserved, no action

SStatus

Address Offset: $1F04_H$

Access Type: Read

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																				IPM				SPD				DET			

This register is the `SStatus` register as defined by the Serial ATA specification.

- **Bit [31:12]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [11:08]**: IPM (R) – This field identifies the current interface power management state.

Value	Definition
0000	Device not present or communication not established
0001	Interface in active state
0010	Interface in Partial power management state
0110	Interface in Slumber power management state
Others	Reserved

- **Bit [07:04]**: SPD (R) – This field identifies the negotiated interface communication speed.

Value	Definition
0000	No negotiated speed
0001	Generation 1 communication rate (1.5 Gbit/s)
0010	Generation 2 communication rate (3.0 Gbit/s)
Others	Reserved

- **Bit [03:00]**: DET (R) – This field indicates the interface device detection and PHY state.

Value	Action
0000	No device detected and PHY communication not established
0001	Device presence detected but PHY communication not established
0011	Device presence detected and PHY communication established
0100	PHY in offline mode as a result of the interface being disabled or running in a BIST loopback mode
Others	Reserved, no action

SError

Address Offset: $1F08_H$

Access Type: Read/Write 1 Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
R	R	R	R	R	**X**	**F**	**T**	**S**	**H**	**C**	**D**	**B**	**W**	**I**	**N**	R	R	R	R	**E**	**P**	**C**	**T**	R	R	R	R	R	R	**M**	**I**
DIAG																ERR															

This register is the `SError` register as defined by the Serial ATA specification.

- **Bit [31:16]**: DIAG (R/W1C) – This field contains bits defined as shown in the following table. Writing *1* to the register bit clears the B, C, F, N, H, W, and X bits. Writing *1* to the corresponding bits in the `Port Interrupt Status` register also clears the F, N, W, and X bits. The B, C, and H bits operate independently of the corresponding error counter registers; if the error counters are used, these bits should be ignored.

Table 38. SError Register Bits (DIAG Field)

Bit	Definition	Description
B	10b to 8b decode error	Latched decode error or disparity error from the Serial ATA PHY
C	CRC error	Latched CRC error from the Serial ATA PHY
D	Disparity error	N/A, always 0; this error condition is combined with the decode error and reported as B error
F	Unrecognized FIS type	Latched Unrecognized FIS error from the Serial ATA Link
I	PHY Internal error	N/A, always 0
N	PHYRDY change	Indicates a change in the status of the Serial ATA PHY
H	Handshake error	Latched Handshake error from the Serial ATA PHY
R	Reserved	Always 0
S	Link Sequence error	N/A, always 0
T	Transport state transition error	N/A, always 0
W	ComWake	Latched ComWake status from the Serial ATA PHY
X	Device Exchanged	Latched ComInit status from the Serial ATA PHY

- **Bit [15:00]**: ERR – This field is not implemented; all bits are always 0.

SActive

Address Offset: $1F0C_H$

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Active bits																															

This register provides indirect access of the `Port Device QActive` registers. It contains the Active bits used to determine the activity of native queued commands for the selected Port Multiplier port (selection in SControl). A *1* in any bit position indicates that the corresponding command is still active in the device.

SNotification

Address Offset: $1F10_H$

Access Type: Read/Write 1 to Clear

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																Notify bits															

This register reports the devices that have sent a Set Device Bits FIS with the Notification bit set.

- **Bit [31:16]**: Reserved (R). This bit field is reserved and returns zeros on a read.
- **Bit [15:00]**: Notify bits (R/W1C) – These 16 bits correspond to the 16 possible devices connected to a Port Multiplier on this port.

Internal Register Space – Base Address 2

These registers are 32-bits wide and provide Indirect Register Access to the registers of the SiI3132 controller. Access to this register space is through the PCI I/O space.

Table 39. SiI3132 Internal Register Space – Base Address 2

Address Offset	Register Name
00_H	Global Register Offset
04_H	Global Register Data
08_H	Port Register Offset
$0C_H$	Port Register Data

Global Register Offset

Address Offset: 00_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																									Dword Offset					00	

This register provides indirect addressing of a Global Register that is otherwise accessible directly through the Base Address 0 Register Space. The Dword address offset for an indirect access is in bits 6 to 2; bits 31 to 7, 1, and 0 are reserved and should always be 0.

Global Register Data

Address Offset: 04_H

Access Type: Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
As defined for indirectly accessed register																															

This register provides the indirect access that is addressed by the `Global Register Offset` register.

Port Register Offset

Address Offset: 08_H

Access Type: Read/Write

Reset Value: 0x0000_0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																		Dword Offset												00	

This register provides indirect addressing of a Port Register otherwise accessible directly through the Base Address 1 Register Space. The Dword address offset for an indirect access is in bits 13 to 2; bits 31 to 14, 1, and 0 are reserved and should always be 0.

Port Register Data

Address Offset: $0C_H$

Access Type: Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
As defined for indirectly accessed register																															

This register provides the indirect access addressed by the `Port Register Offset` register.

Power Management

The register bits described in Table 40 control Power Management in a SiI3132 controller port.

Table 40. Power Management Register Bits

Register	Bits	Description
Port Interrupt Status	PM Change (bit 3)	This bit reports a change in the Power Management mode. It corresponds to the interrupt enabled by bit 3 of the Port Interrupt Enable register.
SError	W (bit 18)	This bit reports a ComWake received from the Serial ATA bus. It corresponds to the interrupt enabled by bit 5 of the Port Interrupt Enable register.
Port Interrupt Status	ComWake (bit 5)	
SControl	SPM (bits 15–12)	This bit field initiates transitions to/from Partial or Slumber power management states; bit 14 corresponds to ComWake (exit power management); bit 13 corresponds to Slumber mode; bit 12 corresponds to Partial mode.
SControl	IPM (bits 11–8)	This bit field disables transitions to Partial or Slumber power management states; bit 9 corresponds to Slumber mode; bit 8 corresponds to Partial mode.
SStatus	IPM (bits 11–8)	This bit field reports the power management state. 0110 – Slumber mode 0010 – Partial mode.

There are two power management modes: Partial and Slumber. These power management modes may be software initiated through the SControl register or device initiated from the Serial ATA device.

Transitions to and from either power management mode generate an interrupt, the Power Management Mode Change Interrupt, which may be masked in the Port Interrupt Enable register (bit 3).

Partial/Slumber mode may be initiated by software through the SControl register. By setting the SPM field to either 0b0001 (Partial) or 0b0010 (Slumber), software causes a PMREQ to the Serial ATA device, which responds with either a PMACK or PMNAK. If a PMACK is received the Partial/Slumber mode is entered. If a PMNAK is ignored; the request remains asserted.

The Serial ATA device may initiate Partial/Slumber mode. Software enables the acknowledgement of this request by setting the IPM field in the SControl register to 0b0001 (Partial), 0b0010 (Slumber), or 0b0011 (Partial or Slumber). If enabled, a PMACK is sent to the device; if not enabled, a PMNAK is sent. When the request is received and its acknowledgement is enabled, Partial/Slumber mode is entered.

Partial/Slumber mode status is reported in the SStatus register (0b0010/0b0110 in the IPM field).

Partial/Slumber mode is cleared by ComWake (asserted when the SPM field is set to 0b0100).

Flash, GPIO, EEPROM, and I²C Programming

Flash Memory Access

The SiI3132 controller supports an external flash memory device of up to 4 Mbits (512 kB) capacity. Access to the flash memory is available using either PCI Direct Access or Register Access.

PCI Direct Access

Access to the Expansion ROM is enabled by setting bit 0 in the Expansion ROM Base Address register at Offset 30h of the PCI Configuration Space. When this bit is set, bits [31:19] of the same register are programmable by the system to set the base address for all flash memory accesses. Read and write operations with the flash memory are initiated by `Memory Read` and `Memory Write` commands on the PCI bus, which can be accessed as bytes, words, or Dwords.

Register Access

This type of flash memory access is carried out through a sequence of internal register read and write operations. The proper programming sequences are as follows:

Flash Write Operation

1. Verify that `Flash Address` register bit 25 (Mem Access Start) is *0*. The bit is *1* when a memory access is in progress. It is *0* when the memory access is complete and ready for another operation.
2. Program the write address for the flash memory access. The address field is bits [18:0] in the `Flash Address` register.
3. Program the write data for the flash memory access. The data field is bits [7:0] in the `Flash Memory Data / GPIO Control` register.
4. Program `Flash Address` register bit 24 (Mem Access Type) to *0* for a memory write.
5. Initiate the flash memory access by setting bit 25 in the `Flash Address` register.

Flash Read Operation

1. Verify that `Flash Address` register bit 25 (Mem Access Start) is *0*. The bit is *1* when a memory access is in progress. It is *0* when the memory access is complete and ready for another operation.
2. Program the read address for the flash memory access. The address field is bits [18:0] in the `Flash Address` register.
3. Program `Flash Address` register bit 24 (Mem Access Type) to *1* for a memory read.
4. Initiate the flash memory access by setting bit 25 in the `Flash Address` register.
5. Verify that `Flash Address` register bit 25 (Mem Access Start) is clear. The bit is *1* when a memory access is in progress. It is *0* when the memory access is complete.
6. Read the data from bits [7:0] in the `Flash Memory Data / GPIO Control` register.

I²C Operation

The SiI3132 controller provides a Multimaster I²C interface. For Auto-initialization of some PCI Configuration registers an external 256-byte EEPROM memory device may be connected to this I²C interface (refer to the Auto-Initialization on page 34). Four registers are provided for programmed read/write access to the interface: the `I2C Slave Address` register, `I2C Control` register, `I2C Data Buffer` register, and `I2C Status` register.

I²C Master Write Operation

1. Verify that I2C Status register bits 02 (Unit Busy) and 03 (Bus Busy) are *0*. These bits are *1* when an access is in progress. They are *0* when the access is complete and another operation can be started.
2. Write *1* to clear bits [10:4] in the I2C Status register. These bits are set if an error occurred during a previous access.
3. Program the I2C Data Buffer register (bits 7:0) with the first byte to be sent. In most applications, the first byte should consist of the I²C Slave address and a *0* value for the RW bit (bit 7).
4. Initiate the I²C write by setting bits of the I2C Control register. At a minimum, set the following bits:
   a. Assert the START bit (Bit[0] is 1) for the first byte in a series.
   b. Leave the STOP bit clear (Bit[1] is 0) for bytes other than the last byte in a series.
   c. Leave the ACK/NACK Ctrl bit clear (Bit[2] is 0) to allow standard ACK/NACK support.
   d. Assert Transfer Byte (Bit[3] is 1).
   e. Clear the Master Abort bit (Bit[4] is 0).
   f. Assert bits [8:5].
   g. Clear bits [15:14].
   h. Set bits [13:9] to subscribe to interrupts as needed.
5. Poll I2C Status register bit 02 (I2C Unit Busy). The bit is *1* while an access is in progress. It becomes *0* when the access is complete. (Alternatively, the appropriate I2C Interrupt may be enabled. See the Global Control register and Global Interrupt Status register descriptions on pages 52 and 53, respectively.)
6. Check the I2C Status register for a value of *1* in bit 6 (Tx MT), bit 1 (ACK/NACK Status), and error bit 10. If the error bits are set or the slave returned a NACK, an error occurred and the write should be aborted.
7. To abort the write, write 0x2FFA to the I2C Control register to issue a Master Abort. In case I2C Status bits 02 (I2C Unit Busy) and 03 (Bus Busy) are set after issuing a Master Abort, reset the SiI3132 controller by setting I2C Control bit 14 (Unit Reset).

Note: Leave the START bit clear (bit 0 of I2C Control register is *0*) for bytes sent other than the first byte and leave the STOP bit clear (bit 1 of I2C Control register is *0*) for bytes other than the last byte in a series.

I²C Master Read Operation

Setup for a Read Operation

1. Verify that I2C Status register bits 02 (Unit Busy) and 03 (Bus Busy) are *0*.
2. Write *1* to clear bits [10:4] in the I2C Status register.
3. Program the I2C Data Buffer register (bits 7:0) with the first byte to be sent. In most applications, the first byte should consist of the I²C Slave address and a *1* value for the RW bit.
4. Initiate the I²C write by setting bits of the I2C Control register. At a minimum, set the following bits:
   a. Assert the START bit (Bit[0] is 1) for the first byte in a series.
   b. Leave the STOP bit clear (Bit[1] is 0) for bytes other than the last byte in a series.
   c. Leave the ACK/NACK Ctrl bit clear (Bit[2] is 0) to allow standard ACK/NACK support.
   d. Assert Transfer Byte (Bit[3] is 1).
   e. Clear the Master Abort bit (Bit[4] is 0).
   f. Assert bits [8:5].
   g. Clear bits [15:14].
   h. Set bits [13:9] to subscribe to interrupts as needed.
5. Poll I2C Status register bit 02 (I2C Unit Busy). The bit is *1* while an access is in progress. It becomes *0* when the access is complete. (Alternatively, the appropriate I²C Interrupt may be enabled. See the Global Control register and Global Interrupt Status register descriptions on pages 52 and 53, respectively.)
6. Check the I2C Status register for a value of *1* in bit 6 (Tx MT), bit 1 (ACK/NACK Status), and error bit 10. If the error bits are set or the slave returned a NACK, an error occurred and the read should be aborted.

Read the Data

If an error did not occur and the I²C Slave returned an ACK to the first byte sent, setup the SiI3132 controller for a read by continuing:

1. Verify that I2C Status register bits 02 (Unit Busy) and 03 (Bus Busy) are 0.
2. Write *1* to clear bits [10:4] in the I2C Status register.
3. Initiate the I²C write by setting bits of the I2C Control register. At a minimum, set the following bits:
   a. Assert the START bit (Bit[0] is 1) for the first byte in a series.
   b. Leave the STOP bit clear (Bit[1] is 0) for bytes other than the last byte in a series.
   c. Leave the ACK/NACK Ctrl bit clear (Bit[2] is *0*) to allow standard ACK/NACK support.
   d. Leave Transfer Byte clear (Bit[3] is *0*).
   e. Clear the Master Abort bit (Bit[4] is *0*).
   f. Assert bits [8:5].
   g. Clear bits [15:14].
   h. Set bits [13:9] to subscribe to interrupts as needed.
4. Poll I2C Status register bit 02 (I2C Unit Busy). The bit is *1* while an access is in progress. It becomes *0* when the access is complete. (Alternatively, the appropriate I2C Interrupt may be enabled. See the Global Control register and Global Interrupt Status register descriptions on pages 52 and 53, respectively.)
5. Check the I2C Status register for a *1* value in bit 7 (Rx Full) and error bit 10. If the error bits are set an error occurred and the read should be aborted.
6. To abort the write, write 0x2FFA to the I2C Control register to issue a Master Abort. In case I2C Status bits 02 (I2C Unit Busy) and 03 (Bus Busy) are set after issuing a Master Abort, reset the SiI3132 controller by setting I2C Control bit 14 (Unit Reset).

Note: Leave the START bit clear (0 value for bit 0 of I2C Control register) for bytes sent other than the first byte and leave the STOP bit clear (0 value for bit 1 of I2C Control register) for bytes other than the last byte in a series.

I²C Slave Read Operations

To set SiI3132 controller to respond as an I²C slave, do the following.

1. Set the I²C Slave Address register with the static address assigned to the SiI3132-enabled I²C Slave
2. Poll I2C Status register bit 09 (Slave Addr Det). (Alternatively, the appropriate I2C Interrupt may be enabled.)
3. Verify that I2C Status register bits 02 (Unit Busy) and 03 (Bus Busy) are 0.
4. Write *1* to clear bits [10:4] in the I2C Status register.
5. Program the I2C Data Buffer register (bits 7:0) with the byte to be sent.
6. Initiate the I²C write by setting bits of the I2C Control register. At a minimum, set the following bits:
   a. Clear the START bit (Bit[0] is 0).
   b. Clear the STOP bit clear (Bit[1] is 0).
   c. Leave the ACK/NACK Ctrl bit clear (Bit[2] is 0) to allow standard ACK/NACK support.
   d. Assert Transfer Byte (Bit[3] is 1).
   e. Clear the Master Abort bit (Bit[4] is 0).
   f. Clear the SCL Enable bit (Bit[5] is 0).
   g. Assert bits [8:6].
   h. Clear bits [15:14].
   i. Set bits [13:9] to subscribe to interrupts as needed.
7. Poll I2C Status register bit 02 (I2C Unit Busy). The bit is *1* while an access is in progress. It becomes *0* when the access is complete. (Alternatively, the appropriate I2C Interrupt may be enabled. See the Global Control register and Global Interrupt Status register descriptions on pages 52 and 53, respectively.)
8. Check the I2C Status register for a value of *1* in bit 6 (Tx MT), bit 1 (ACK/NACK Status), and error bit 10. If the error bits are set or the master returned a NACK, an error occurred and the read should be aborted.
9. To abort the write, write 0x2FFA to the I2C Control register to issue a Master Abort. In case I2C Status bits 02 (I2C Unit Busy) and 03 (Bus Busy) are set after issuing a Master Abort, reset the SiI3132 controller by setting I2C Control bit 14 (Unit Reset).

Standards Documents

Table 41 lists the abbreviations of the standards mentioned in this document. For more information on these specifications contact the responsible standards groups listed in Table 42.

Table 41. Referenced Documents

Abbreviation	Standards publication, organization, and date
SATA-IO	*Serial ATA / High Speed AT Attachment Specification*, Revision 1.0
PCI-E	*PCI Express Base Specification*, Revision 1.0a

Table 42. Standards Groups Contact Information

Standards Group	Web URL	e-mail	phone
SATA-IO	http://www.sata-io.org	administration@sata-io.org	(503) 619-0572
PCI-SIG	http://www.pcisig.com	administration@pcisig.com	(503) 619-0569

Disclaimers

These materials are provided on an "AS IS" basis. Silicon Image, Inc. and its affiliates disclaim all representations and warranties (express, implied, statutory or otherwise), including but not limited to: (i) all implied warranties of merchantability, fitness for a particular purpose, and/or non-infringement of third party rights; (ii) all warranties arising out of course-of-dealing, usage, and/or trade; and (iii) all warranties that the information or results provided in, or that may be obtained from use of, the materials are accurate, reliable, complete, up-to-date, or produce specific outcomes. Silicon Image, Inc. and its affiliates assume no liability or responsibility for any errors or omissions in these materials, makes no commitment or warranty to correct any such errors or omissions or update or keep current the information contained in these materials, and expressly disclaims all direct, indirect, special, incidental, consequential, reliance and punitive damages, including WITHOUT LIMITATION any loss of profits arising out of your access to, use or interpretation of, or actions taken or not taken based on the content of these materials.

Silicon Image, Inc. and its affiliates reserve the right, without notice, to periodically modify the information in these materials, and to add to, delete, and/or change any of this information.

Notwithstanding the foregoing, these materials shall not, in the absence of authorization under U.S. and local law and regulations, as required, be used by or exported or re-exported to (i) any U.S. sanctioned or embargoed country, or to nationals or residents of such countries; or (ii) any person, entity, organization or other party identified on the U.S. Department of Commerce's Denied Persons or Entity List, the U.S. Department of Treasury's Specially Designated Nationals or Blocked Persons List, or the Department of State's Debarred Parties List, as published and revised from time to time; (iii) any party engaged in nuclear, chemical/biological weapons or missile proliferation activities; or (iv) any party for use in the design, development, or production of rocket systems or unmanned air vehicles.

Products and Services

The products and services described in these materials, and any other information, services, designs, know-how and/or products provided by Silicon Image, Inc. and/or its affiliates are provided on as "AS IS" basis, except to the extent that Silicon Image, Inc. and/or its affiliates provides an applicable written limited warranty in its standard form license agreements, standard Terms and Conditions of Sale and Service or its other applicable standard form agreements, in which case such limited warranty shall apply and shall govern in lieu of all other warranties (express, statutory, or implied). EXCEPT FOR SUCH LIMITED WARRANTY, SILICON IMAGE, INC. AND ITS AFFILIATES DISCLAIM ALL REPRESENTATIONS AND WARRANTIES (EXPRESS, IMPLIED, STATUTORY OR OTHERWISE), REGARDING THE INFORMATION, SERVICES, DESIGNS, KNOW-HOW AND PRODUCTS PROVIDED BY SILICON IMAGE, INC. AND/OR ITS AFFILIATES, INCLUDING BUT NOT LIMITED TO, ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND/OR NON-INFRINGEMENT OF THIRD PARTY RIGHTS. YOU ACKNOWLEDGE AND AGREE THAT SUCH INFORMATION, SERVICES, DESIGNS, KNOW-HOW AND PRODUCTS HAVE NOT BEEN DESIGNED, TESTED, OR MANUFACTURED FOR USE OR RESALE IN SYSTEMS WHERE THE FAILURE, MALFUNCTION, OR ANY INACCURACY OF THESE ITEMS CARRIES A RISK OF DEATH OR SERIOUS BODILY INJURY, INCLUDING, BUT NOT LIMITED TO, USE IN NUCLEAR FACILITIES, AIRCRAFT NAVIGATION OR COMMUNICATION, EMERGENCY SYSTEMS, OR OTHER SYSTEMS WITH A SIMILAR DEGREE OF POTENTIAL HAZARD. NO PERSON IS AUTHORIZED TO MAKE ANY OTHER WARRANTY OR REPRESENTATION CONCERNING THE PERFORMANCE OF THE INFORMATION, PRODUCTS, KNOW-HOW, DESIGNS OR SERVICES OTHER THAN AS PROVIDED IN THESE TERMS AND CONDITIONS.

1060 E. Arques Avenue
Sunnyvale, CA 94085
T 408.616.4000 F 408.830.9530
www.siliconimage.com

1060 E. Arques Avenue
Sunnyvale, CA 94085
T 408.616.4000 F 408.830.9530
www.siliconimage.com