TECHNICAL MANUAL

LSISASx12/LSISASx12A 3.0 Gbit/s Serial Attached SCSI / Serial ATA Expander

October 2005

Version 3.0

LSI LOGIC®

DB14-000277-04

This document contains proprietary information of LSI Logic Corporation. The information contained herein is not to be used by or disclosed to third parties without the express written permission of an officer of LSI Logic Corporation.

LSI Logic products are not intended for use in life-support appliances, devices, or systems. Use of any LSI Logic product in such applications without written consent of the appropriate LSI Logic officer is prohibited.

Purchase of I^2C components of LSI Logic Corporation, or one of its sublicensed Associated Companies, conveys a license under the Philips I^2C Patent Rights to use these components in an I^2C system, provided that the system conforms to the I^2C standard Specification as defined by Philips.

Document DB14-000277-04, Version 3.0, October 2005
This document describes LSI Logic Corporation's LSISASx12/LSISASx12A 3.0 Gbit/s Serial Attached SCSI expander and will remain the official reference source for all revisions/releases of this product until rescinded by an update.

LSI Logic Corporation reserves the right to make changes to any products herein at any time without notice. LSI Logic does not assume any responsibility or liability arising out of the application or use of any product described herein, except as expressly agreed to in writing by LSI Logic; nor does the purchase or use of a product from LSI Logic convey a license under any patent rights, copyrights, trademark rights, or any other of the intellectual property rights of LSI Logic or third parties.

TRADEMARK ACKNOWLEDGMENT
LSI Logic, the LSI Logic logo design, Fusion-MPT, Gflx, and GigaBlaze are trademarks or registered trademarks of LSI Logic Corporation. AMBA is a trademark of ARM Ltd., used under license. All other brand and product names may be trademarks of their respective companies.

JM

To receive product literature, visit us at http://www.lsilogic.com.

For a current list of our distributors, sales offices, and design resource centers, view our web page located at

http://www.lsilogic.com/contacts/index.html

Preface

This book is the primary reference and Technical Manual for the LSISASx12 and the LSISASx12A 3.0 Gbit/s Serial Attached SCSI (SAS) expander chips. It contains a complete functional description for the devices, as well as complete physical and electrical specifications.

Audience

This document assumes that you have some familiarity with computer chips and related support devices. The people who benefit from this book are:

- Engineers and managers who are evaluating the expander for possible use in a system
- Engineers who are designing the expander into a system

Organization

This document has the following chapters and appendixes:

- Chapter 1, **Introduction**, provides an overview of the LSISASx12/LSISASx12A expander.
- Chapter 2, **Functional Description**, provides a block diagram of the LSISASx12/LSISASx12A expander, and explains the operation of the various LSISASx12/LSISASx12A components.
- Chapter 3, **Signal Descriptions**, provides signal descriptions for the LSISASx12/LSISASx12A signals.
- Chapter 4, **Register Descriptions**, provides the LSISASx12/LSISASx12A register map, as well as bit-level descriptions of the register space.

- Chapter 5, **Specifications**, provides the environmental, electrical, mechanical specifications for the LSISASx12/LSISASx12A, as well as pinout diagrams for the LSISASx12/LSISASx12A.
- Appendix A, **Register Summary**, provides a summary of the registers in the LSISASx12/LSISASx12A.
- Appendix B, **Example Code for API2C Interface and Serial EEPROM**, provides example code for programming the API2C interface to access a Serial EEPROM.

Related Publications and Specifications

- **LSI Logic Documents**
  LSISAS1064 Serial Attached SCSI Controller Technical Manual
  DB14-000274-01

  Fusion MPT Device Manager user's Guide, Version 2.0
  DB15-000187-02
- **LSI Logic Word Wide Web Home Page**
  www.lsilogic.com
- **Serial Attached SCSI**
  Version 1.0
  T10 Technical Committee
  INCITS (International Committee for Information Technology Standards.)
  www.t10.org
- **Serial ATA: High Speed Serialized AT Attachment**
  Revision 1.0a,
  Serial ATA Workgroup,
  www.serialata.org
- **Serial ATA II: Port Selector**
  Revision 1.0,
  Serial ATA Workgroup,
  www.serialata.org
- **SFF-8485 Specification for Serial GPIO (SGPIO) Bus**
  http://www.sffcommittee.com/ns/
- **Philips I^2C Bus Specification**
  http://www.semiconductors.philips.com

Conventions Used in This Manual

The word *assert* means to drive a signal true or active. The word *deassert* means to drive a signal false or inactive. Signals that are active LOW end in a "/".

Hexadecimal numbers are indicated by the prefix "0x" —for example, 0x32CF. Binary numbers are indicated by the prefix "0b" —for example, 0b0011.0010.1100.1111.

Revision History

Revision	Date	Remarks
Final Version 3.0	7/2005	Added LSISASx12A info and API2C registers and state machines.
Final Version 2.2	3/2005	Updated "Related Publications and Specifications" section. Added power dissipation footnote to Table 5.2. Added Section 5.2, "AC Characteristics."
Final Version 2.1	1/2005	Corrected ESD typo in Table 5.1.
Final Version 2.0	12/2004	Final release version.
Advance Version 0.2	12/2003	Initial release of document.

Contents

Figures

Tables

Chapter 1
Introduction

This chapter describes the LSISASx12/LSISASx12A expander and consists of the following sections:

1.1 Overview

The LSISASx12/LSISASx12A expander is a 12-port, 3 Gbit/s Serial Attached SCSI (SAS)/Serial ATA (SATA) expander device. The LSISASx12/LSISASx12A expander provides the functionality for connecting targets and initiators. The expander provides phys for up to 12 SAS initiator, SAS target, SAS expander, or Serial ATA (SATA) target devices. The LSISASx12/LSISASx12A supports both wide and narrow port configurations. Narrow ports have one phy per port. Wide ports have multiple phys per port. The expander supports the SAS protocol as described in the *Serial Attached SCSI (SAS) Standard, version 1.0,* as well as the following features of the *SAS Standard, version 1.1*:

- BROADCAST(SES)
- BROADCAST(CHANGE) primitive for virtual phy resets, enables, and disables
- SATA II Port Selector
- Report Manufacturing Information in the format defined by the SAS version 1.1 specification
- BREAK handling clarifications

- ALIGN transmission from expanders

The LSISASx12/LSISASx12A expander manages connection requests between end devices and/or other expanders, and supports the Serial SCSI Protocol (SSP), the Serial ATA Tunneled Protocol (STP), the Serial Management Protocol (SMP), and the LSISASx12/LSISASx12A supports SATA as defined in the *Serial ATA: High Speed Serialized AT Attachment Specification, version 1.0a.* The expander includes an SMP target and an enclosure management bridge. The expander supports direct, table, and subtractive routing modes. LSI Logic produces the LSISASx12/LSISASx12A expander using the Gflx™ (0.11 µm) process.

Each of the 12 phys on the LSISASx12 expander supports SAS transfers of up to 3.0 Gbits/s and SATA transfers of up to 1.5 Gbits/s. Each of the 12 phys on the LSISASx12A expander supports SAS transfers of up to 3.0 Gbits/s and SATA transfers of up to 3.0 Gbits/s. The SASx12A is a pin-compatible upgrade from the LSISASx12 for those customers requiring 3.0 Gbits/s SATA data transfers and/or enhanced serial GPIO (SGPIO) support.

There are two major differences between the LSISASx12 and the LSISASx12A expanders.

- The LSISASx12A supports 3.0 Gbit/s SAS and SATA connections, whereas the LSISASx12 supports only 1.5 Gbit/s SATA connections.
- The LSISASx12A provides enhanced SGPIO support which enables additional data modulation control for blink rates and patterns as well as programmable on/off times for LED/GPIO activity.

The LSISASx12/LSISASx12A 12-port SAS expander offers high performance, high disk drive connectivity, scalability and flexibility in various storage environments as an alternative to today's expensive and complex offerings. The LSISASx12/LSISASx12A[1] SAS expander is ideal for high availability and scalable server clustering environments and front-end storage subsystems used in clusters, SANs, and NAS environments. LSI Logic SAS expanders are ideal for data centers and Storage Area Networks, leveraging existing SCSI infrastructure for investment protection and ease of migration and implementation.

1. In the rest of this document LSISASx12 refers to both the LSISASx12 expander and the LSISASx12A expander unless specifically noted.

1.2 Application Examples

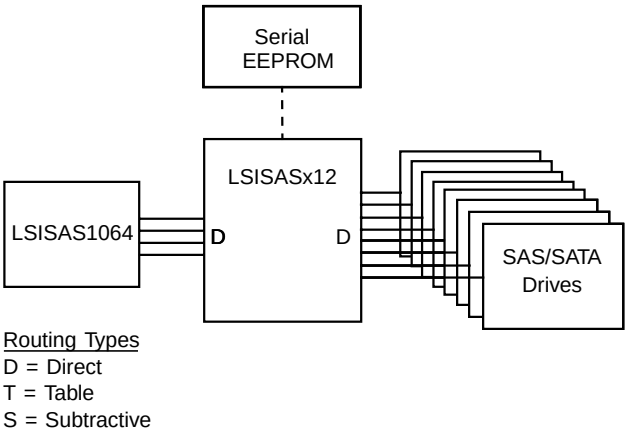
The LSISASx12 can be used in simple topologies to attach an initiator to SAS/SATA devices, in edge expander topologies to increase the number of accessible devices, or in fault-tolerant path-redundancy topologies to improve system reliability. By cascading multiple LSISASx12 expanders, users can attach up to 144 devices.

The LSISASx12 expander can be used to construct many different SAS topologies. A few examples are shown in the following subsections.

1.2.1 Single Expander Example

Simple topologies can utilize a single expander to access SAS or SATA drives as illustrated by Figure 1.1.

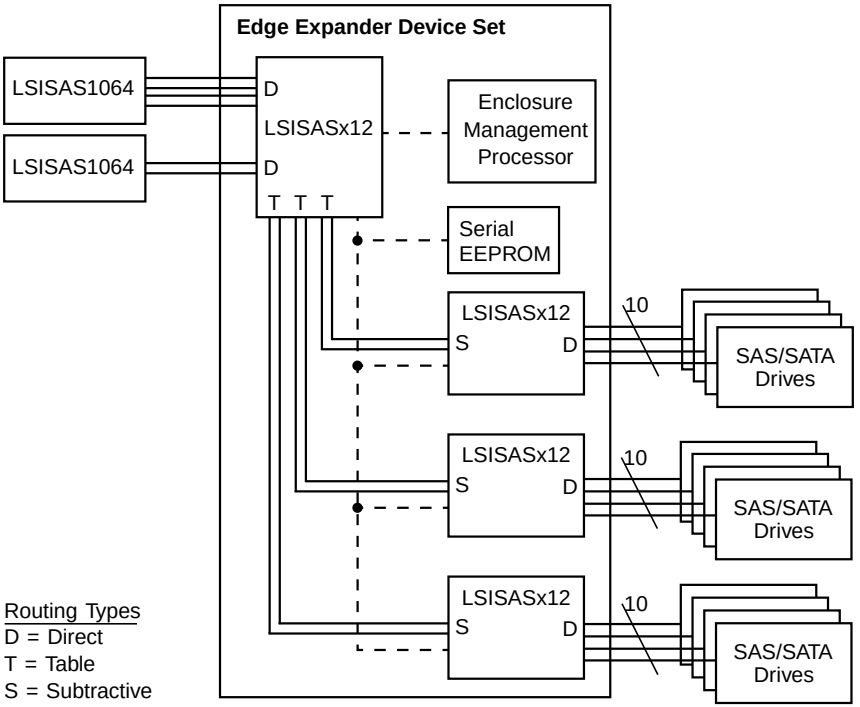
Figure 1.1 Single Expander Example



1.2.2 Multiple Expander Example

To increase the number of devices within a topology, multiple LSISASx12 expanders can be cascaded into edge expander device sets as illustrated in Figure 1.2. Notice that the expander requires minimal external support components including a reference clock and a Serial EEPROM device. A single serial EEPROM can provide unique configurations to up to four LSISASx12 expanders.

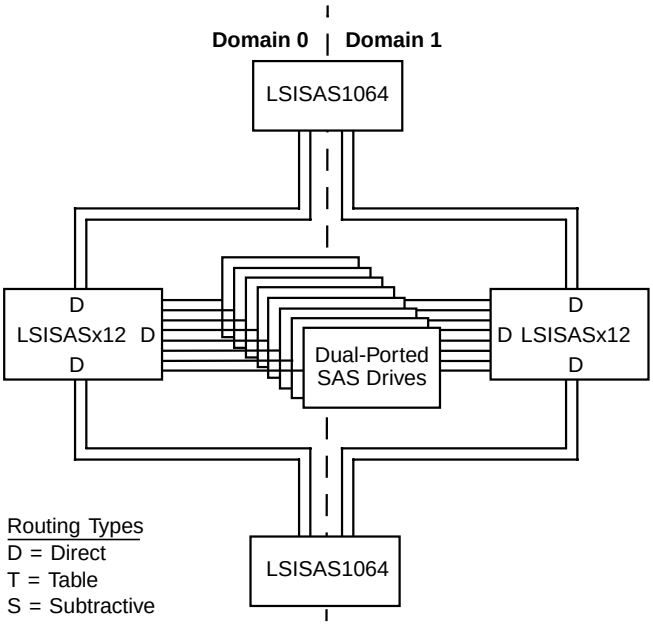
Figure 1.2 Multiple LSISASx12 Expanders



1.2.3 Path Redundancy Application

Path redundancy can be provided by utilizing dual-ported SAS drives. Figure 1.3 shows a path redundancy application.

Figure 1.3 LSISASx12 Path Redundancy Application



1.3 Expander Communication

The LSISASx12 supports Serial SCSI Protocol (SSP), Serial ATA Tunneled Protocol (STP), Serial Management Protocol (SMP), and Serial ATA (SATA) protocol. SSP provides mapping of SCSI commands to support initiators and targets, and enables the LSISASx12 to communicate with other SSP devices, such as SAS controllers and SAS drives. STP maps serial ATA commands to support multiple initiators and targets, and enables the LSISASx12 to communicate with serial ATA devices. The STP target enclosure management bridge communicates with an external enclosure management processor.

SMP provides management commands for SAS expanders. The LSISASx12 provides an SMP block that enables initiator devices to communicate directly to the LSISASx12. The SMP Target provides access to the standard SMP functions and extended functions of the LSISASx12. Initiators can use the SMP Target to perform discovery on the LSISASx12 expander.

1.4 Features

This section lists the features of the LSISASx12 expander.

1.4.1 Expander Features

This subsection describes the expander features.

- Supports SSP, STP, and SMP as defined in the *Serial Attached SCSI (SAS) Standard, version 1.0*
- Supports the following features of the *SAS Standard, version 1.1*
  - BROADCAST(SES)
  - BROADCAST(CHANGE) primitive for virtual phy resets, enables, and disables
  - SATA Port Selector
  - Report Manufacturing Information in the format defined by the SAS version 1.1 specification
  - BREAK handling clarifications
  - ALIGH transmission from expanders
- Supports SATA as defined in the *Serial ATA: High Speed Serialized AT Attachment Specification, version 1.0a*
- Supports multiple data rates and auto-negotiation between
  - 1.5 Gbits/s and 3.0 Gbits/s SAS
  - 1.5 Gbits/s and 3.0 Gbits/s SATA (on LSISASx12A only)
- Provides a low latency connection router to efficiently create and maintain connections
- Provides an Inter-IC (I^2C) bus to the STP target enclosure management bridge to streamline communication with an optional enclosure management processor
- Provides configurable drive spin-up sequencing on a per-phy basis
- Programmable TX and RX signal polarity for optimization of board routing
- Provides a scalable interface that supports up to 144 devices through multiple expanders

- Provides four configurable GPIO signals for each phy to indicate drive activity, data transfers, faults, and cable detection
- Offers an advanced GPIO interface that provides serial and parallel GPIO capabilities
- Provides a Serial I/O (SIO) interface for managing general-purpose data communication

1.4.2 STP/SATA Features

This subsection describes the STP features.

- The LSISASx12 supports SATA data transfers of 1.5 Gbits/s
- The LSISASx12A supports SATA data transfers of 1.5 Gbits/s or 3.0 Gbits/s
- Supports STP data transfers of 3.0 Gbits/s and 1.5 Gbits/s
- Allows addressing of multiple SATA targets
- Allows multiple initiators to address a single target

1.4.3 SMP Features

This subsection describes the SMP features.

- Implements SMP functions as defined in the SAS standard
- Uses the SMP Report Manufacturer Information Frame format that is specified in version 1.1 of the SAS standard
- Decodes SMP packets that are destined for the expander
- Performs CRC checking and generation on the request frames and response frames

1.4.4 Usability

This subsection describes the usability features.

- Simplifies cabling with a point-to-point, serial architecture
- Supports multiple routing methodologies
  - Supports direct, table, and subtractive routing
  - Allows per-phy configurable routing

1.4.5 Flexibility

This subsection describes the features that increase the flexibility of the LSISASx12 expander.

- Allows concurrent connections to SAS or SATA targets
- Offers configurable options
  - Stores configuration options in a serial EEPROM
  - A single serial EEPROM can provide unique configuration information for up to 4 LSISASx12 expanders
  - A serial EEPROM is optional if the enclosure management processor is implemented
- Allows flexible allocation of routing table entries to the LSISASx12 phys
- Allows reuse of routing table resources across all of the phys composing a wideport

1.4.6 Testing and Reliability

This subsection describes the testing and reliability features.

- Uses proven GigaBlaze® transceivers
- Provides 2 kV ESD protection
- Provides latch-up protection
- Has a high proportion of power and ground pins
- Uses the proven Gflx technology
- Provides a UART to support serial debugging
- Supports JTAG testing

Chapter 2
Functional Description

This chapter provides a subsystem level overview of the LSISASx12 expander chip. This chapter consists of the following sections.

- Section 2.1, “Block Diagram Description”
- Section 2.2, “Routing”
- Section 2.3, “Addressing”
- Section 2.4, “Vendor-Specific SMP Application Layer Commands”
- Section 2.5, “Boot Loader, Serial EEPROM, and API2C Interface”
- Section 2.6, “GPIO/LED and SIO Configuration”
- Section 2.7, “STP Enclosure Management Interface”

This chapter assumes that the reader understands the SAS standard.

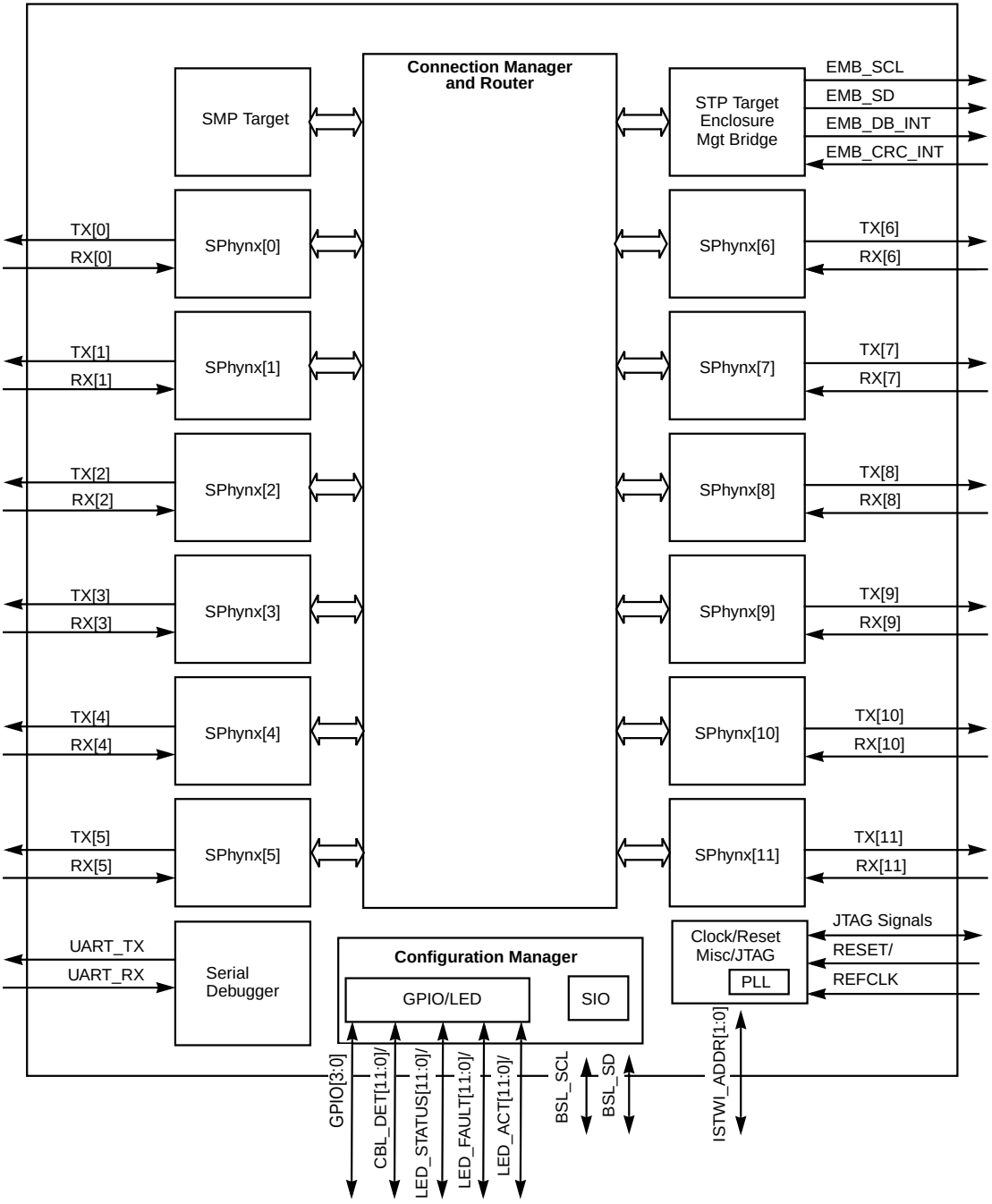
2.1 Block Diagram Description

The LSISASx12 consists the following blocks:

- Connection Manager and Router
- SMP Target
- STP Target Enclosure Management Bridge
- SPhynx [11:0]
- Serial Debugger
- Configuration Manager
- Clock/Reset/JTAG

Figure 2.1 provides the block diagram, which shows the relationships between these modules. The following subsections discuss the modules.

Figure 2.1 LSISASx12 Block Diagram



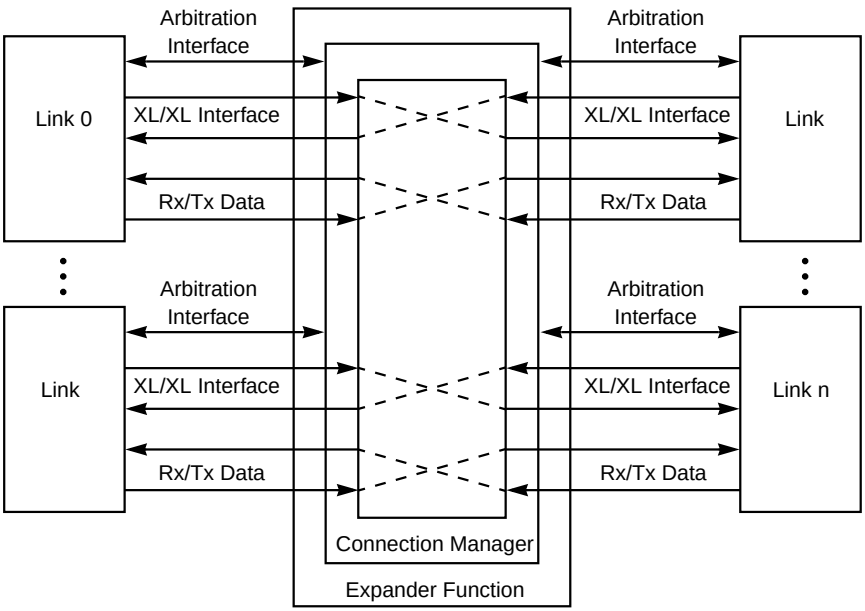
2.1.1 Connection Manager and Router

The connection manager responds to connection requests by scheduling and allocating the connection router path resources. When a link is established, the linked phys can communicate and transfer data. The connection router connects signals between pairs of phys, or between a phy and an internal port (SMP or STP) with minimal latency.

The connection manager establishes a connection when a link requests a connection. The connection manager looks at all connection requests and establishes as many connections as possible. The connection manager uses a distributed arbitration mechanism. Once a connection is established, the connected links communicate with each other and pass the receive/transmit data stream.

Registers within the connection manager are readable and writable through the register bus interface. Section 4.2, “Configuration Manager Registers,” describe the connection manager registers. Figure 2.2 illustrates the operation of the connection manager.

Figure 2.2 Connection Manager and Router Operation



2.1.2 SPhynx[n]

The SPhynx[n] modules contain SAS physical, phy, and link layer functionality. The SPhynx[n] modules interface with the connection manager and the connection router to establish connections between ports. Each SPhynx[n] module contains a GigaBlaze core that converts the serial receive data to a parallel format, and converts the parallel transmit data to a serial format. Each SPhynx module can also function as an STP to SATA bridge. The LSISASx12 expander contains 12 SPhynx[n] modules, one for each port. The modules are numbered SPhynx[0] to SPhynx[11]. These modules are also referred to as Phy[0] to Phy[11].

2.1.3 SMP Target

The SMP target supports SMP functions on the LSISASx12 expander. This block decodes SMP packets destined to the expander device and performs the requested operation. The SMP target also provides generation of SMP response frames.

The SMP target consists of:

- SMP link layer
- SMP register bus interface slave
- SMP port layer
- SMP transport layer
- SMP application layer

The SMP Target is responsible for the following tasks:

- Receiving SMP OPEN address frames (OAF)
- Transmitting OPEN_REJECT(PROTOCOL NOT SUPPORTED) frames in response to OAFs when the:
  - OAF protocol is not SMP
  - OAF initiator bit is cleared to 0b0
  - OAF Features bits [3:0] are not 0b0000
- Transmitting OPEN_REJECT(RETRY) in response to OAFs if the Application Layer is processing an SMP Request

- Transmitting OPEN_ACCEPT in response to OAFs which are not covered by the previous two bullet items
- Receiving SMP Request Frames
- CRC checking on SMP Request Frames
- Validating fields within SMP Request Frames
- Processing mandatory SMP functions
- Processing vendor-specific SMP functions
- Generating SMP response data
- Generating CRC for SMP Response Frames
- Transmitting SMP Response Frames

The SMP application layer is responsible for processing SMP commands. This layer implements the mandatory commands as the SAS standard requires, as well as several vendor-specific SMP commands. These commands are documented in Section 2.4, “Vendor-Specific SMP Application Layer Commands,” on page 2-12.

2.1.4 STP Target Enclosure Management Bridge

The STP target enclosure management bridge (STP EMB) is an STP to I^2C bridge that allows communication with an enclosure management processor. This block transports STP packets between the LSISASx12 expander and the optional external enclosure management processor. The STP EMB supports connection management to the external enclosure management processor, consisting of doorbell communication and data transfer functions.

The EMB_SCL and EMB_SD signals provide the clock and data interface. Two additional signals, EMB_DB_INT and EMB_CRC_INT provide the doorbell and CRC error handshaking to external enclosure processors. The STP also uses the ISTWI_ADDR[1:0] signals to communicate. Section 2.7, “STP Enclosure Management Interface,” on page 2-28 provides more information on the STP EMB interface.

2.1.5 Configuration Manager

The configuration manager performs device configuration, broadcast processing, startup sequencing, and spin-up control. The configuration manager also configures the GPIO/LED usages and the SIO interface.

The configuration manager has a 64 Kbyte register block. The first 16 Kbytes is for the Config Manager registers, the second 16 Kbytes is reserved for use by the APB Register Interface, the next 16 Kbytes is for the LED control block, and the last 16 Kbytes is for the SIO interface block. Section 4.2, “Configuration Manager Registers,” on page 4-3 describes these registers.

2.1.5.1 Boot Sequencer

The configuration manager reads a boot record of the configuration parameters through the boot loader. An external serial EEPROM stores the boot record. The boot record provides a fixed data field for common configuration information, and an optional data field for additional, vendor-specific configuration information. The use of the ISTWI_ADDR[1:0] signals permits a single serial EEPROM to provide boot records for up to four LSISASx12 expanders. Section 2.5, “Boot Loader, Serial EEPROM, and API2C Interface,” on page 2-17 provides more information on the boot sequencer, boot record, and serial EEPROM loader.

2.1.5.2 Broadcast Processing

The configuration manager monitors broadcast request inputs from each SPhynx module and requests the transmission of BROADCAST primitive sequences by SPhynx modules.

2.1.5.3 GPIO/LED Block

The GPIO/LED block controls the activity of the GPIO signals. There are 52 independent GPIO signals. The function of each GPIO and LED signal is set in the LED control registers, which are documented in Section 4.2, “Configuration Manager Registers,” Possible functions are:

- Activity LED – typically supported by the LED_ACT[11:0]/ signals
- Status LED – typically supported by the LED_STATUS[11:0]/ signals

- Fault LED – typically supported by the LED_FAULT[11:0]/ signals
- Cable Detection – typically supported by the CBL_DET[11:0]/ signals
- General GPIO – typically supported by the GPIO[3:0] signals

Section 2.6, “GPIO/LED and SIO Configuration,” on page 2-23 provides more information on the configuration of the GPIO and LED signals.

2.1.5.4 SIO Functional Description

The LSISASx12 transmits SGIO information on a serial interface to external logic. The SIO interface has BlinkClkin, BlinkClkout, SioClkin, Siolkout, SioDout, SioStart, SioEnd, and SioDin signals. SioStart and SioEnd control the input and output data streams, SioDout and SioDin. The SIO signals are mapped to the LSISASx12 GPIO/LED signals.

The LSISASx12A provides enhanced SGPIO support which is not available on the LSISASx12. The enhanced functionality enables additional data modulation for blink rates and blink patterns. The LSISASx12A also provides programmable LED/GPIO on/off times. Configuration of the enhanced SGPIO interface is accomplished through SMP register write commands.

Table 2.1 describes the interface signals and provides the SIO to GPIO/LED signal-pin mapping. More information on the SIO interface is available in Section 2.6.2, “SIO Configuration,” on page 2-26, Section 3.7, “Multiplexed SIO Interface,” on page 3-7, and in the SIO register descriptions on pages 4-55 to 4-66

Table 2.1 SIO Signal Description and Mapping

SIO Signal Name	LSISASx12 Signal	I/O	Description
SioEnd	LED_STATUS[6]/	O	This output signal indicates that this module is currently driving the last bit of serial data on the SioDout line.
SioStart	LED_STATUS[7]/	I	This signal indicates that serial data may be driven on the SioDout line during the next clock cycle, and that data can be received on the SioDin line. In Single originator mode, which is set by the SioMode bit in the SIO Control register, the SioEnd signal provides the start signal, and the LSISASx12 does not require an SioStart input.
SioDout	LED_STATUS[8]/	O	This signal provides the serial data output line.
SioDin	LED_STATUS[9]/	I	This signal provides the serial data input line.
SioClkin	LED_STATUS[10]/	I	An originator with ClkEnable set to 1 in its SIO_CFG register drives SioClkout as an output. The ClkDivide bits in the SIO_CFG register control the frequency of this clock. Originators with ClkEnable cleared to 0 receive this clock signal on SioClkin.
SioClkout	LED_STATUS[11]/	O	
BlinkClkin	LEDSYNCIN	I	An originator with ClkEnable set to 1 in its SIO_CFG register drives BlinkClkout as an output with the frequency fixed at 0.5 Hz. Originators with ClkEnable cleared to 0 receive this clock signal on BlinkClkin.
BlinkClkout	LEDSYNCOUT	O	

2.1.5.5 Spin-Up Control

This subblock coordinates when each SPhynx module allows an attached drive to spin up. The spin-up controller monitors spin-up requests from each SPhynx module and issues spin-up acknowledgements to the requesting SPhynx modules according to the following configurable parameters:

- Spin Number – maximum number of concurrent spin-ups
- Spin-up Delay – spin-up time interval
- Spin Mode – selects one of three modes for spin-up of SATA drives: Immediate, Host-notify, or self-timed

The module contains 12 timers, one for each of the 12 phys. If the number of timers currently running is less than Spin Number, the spin-up controller grants a spin-up acknowledgement and then starts the associated timer. If the number of timers currently running is greater than or equal to the Spin Number, the spin-up controller does not grant any more spin-up acknowledgements until one or more of the currently running timers reaches the Spin-up Delay. The spin-up controller uses round-robin arbitration.

The spin-up timers use a 1 ms clock, are 14 bits wide, and can time from 2 ms to 16.4 seconds, in 1 ms increments. The number of timers that are allowed to run simultaneously (Spin Number) is programmable from zero to 12. A setting of zero results in no spin-up acknowledgement being granted, and setting of 12 results in every spin-up request being immediately acknowledged. The Spin-up Control register configures the spin-up controller.

The LSISASx12 uses the Spin Mode parameter to provide special spin-up support for phys that are connected to SATA drives. The Immediate mode allows OOB to occur with the drive as soon as both phys are ready. The Host-Notify mode holds OOB until a host in the domain issues a link reset to the expander phy. The Self-Timed mode holds OOB until Spin Number and Spin Delay parameters permit the drive to spin-up.

2.1.6 Clock/Reset/Misc/JTAG

This block includes the required logic for testing, clock generation, and reset generation.

2.1.7 Serial Debugger

This block contains the serial debugger functionality. The serial debugger includes a UART interface that is used for debugging the chip. The UART supports 19.2 Kbaud transfers.

2.2 Routing

The LSISASx12 supports direct routing, table routing, and subtractive routing. The individual routing method is configurable on a per-phy basis.

2.2.1 Direct Routing

Direct routing is done for SAS initiators, SAS targets, or SATA targets that directly attach to one of the LSISASx12 phys. The LSISASx12 routes frames based on the SAS address in the frame. The LSISASx12 does not use routing tables or subtractive routing when performing direct routing.

2.2.2 Subtractive Routing

Subtractive routing occurs when the LSISASx12 routes frames with unresolved addresses to a specific port. This port can be one of the SPhynx modules or a group of SPhynx modules, as in the case of a wide port. Subtractive routing is enabled through the register bus interface.

The ECM Config register configures subtractive routing. The SAS standard does not allow end devices to use subtractive routing.

2.2.3 Table Routing

The LSISASx12 supports 144 routing table entries in 12 banks with 12 entries each. Each phy can be configured to use any or all of the 12 banks, provided that the banks are contiguous. All entries include the WWN and a disable bit. The WWN is valid when the disable bit is cleared.

The table routing attribute is blocked when the Device Type is set to end device. As such, the LSISASx12 uses direct routing. The SAS standard does not allow end devices to use table routing.

Routing entries are stored in 12 register banks, which are accessible through the register bus interface. Each bank supports up to 12 entries. All entries include the WWN and a disable bit. Each phy can select any, all, or none of these banks. The Bank Enable bits control the bank selection. All banks selected by a phy are searched for an address hit to that phy during the arbitration process. The remote entry disable bit must be cleared to enable an address hit to a remote WWN in the remote routing table.

Initiators obtain routing entry data and deliver it to the SMP target. The SMP target then delivers this data to the routing table through the primary internal registers.

The LSISASx12 employs hardware logic to compare requested path attributes with all valid entries in each selected bank. If a match is found, the LSISASx12 attempts to form a link between the devices.

2.3 Addressing

This section discusses the different address spaces for the LSISASx12 and SAS addressing.

2.3.1 Address Spaces

The LSISASx12 internal registers are divided into the primary internal register space and the EMB register space. The primary internal registers allow access to most registers in the device, including registers for the configuration manager, the connection manager, and the individual SPhynx modules. The primary internal registers may be accessed through the serial debugger, the SMP target, the EMB, or the serial EEPROM boot loader. The primary internal registers are described in Section 4.1, “Primary Internal Registers Address Map,” on page 4-2 through Section 4.6, “Expander Connection Manager Registers,” on page 4-124.

The EMB registers have their own register space, which the primary internal registers cannot access. However, the EMB can access the primary internal registers through its register space. Through the EMB, the SEP can access the primary internal registers for device configuration and/or LED control. Access to these registers is accomplished by placing the desired command and address into the RR_Command0 - RR_Command3 registers. For write operations the data to be written is placed in the RR_Data0 - RR_Data7 registers. For read operations data returned from the primary internal registers is placed into the RR_Data0 - RR_Data7 registers. Writing to the RR_Control register causes execution of the command to the primary internal registers. These registers are described in Section 4.7, “EMB Slave Registers,” on page 4-133.

2.3.2 SAS Addressing

Each LSISASx12 expander requires a block of 16 addresses starting at 0x000–0xFFFF. When performing STP-SATA Bridge addressing or

internal target addressing, address bits [63:4] are given by World Wide Name bits [63:4], and address bits [3:0] are given by the port number bits [3:0]. Table 2.2 provides the port number encodings for the LSISASx12.

Table 2.2 Port Number Encodings

Port	Definition
0x0	Phy 0 (when SATA device attached)
0x1	Phy 1 (when SATA device attached)
0x2	Phy 2 (when SATA device attached)
0x3	Phy 3 (when SATA device attached)
0x4	Phy 4 (when SATA device attached)
0x5	Phy 5 (when SATA device attached)
0x6	Phy 6 (when SATA device attached)
0x7	Phy 7 (when SATA device attached)
0x8	Phy 8 (when SATA device attached)
0x9	Phy 9 (when SATA device attached)
0xA	Phy 10 (when SATA device attached)
0xB	Phy 11 (when SATA device attached)
0xC	STP Target to EMB
0xD	Reserved
0xE	Reserved
0xF	SMP Target

2.4 Vendor-Specific SMP Application Layer Commands

The SMP Target implements the following the Vendor-Specific commands:

- WRITE N REGISTERS (0xC0)
- READ N REGISTERS (0x40)

The following sections describe the format for the request and response frames associated with these commands.

2.4.1 WRITE N REGISTERS (0xC0)

This Vendor-Unique function writes to N contiguous 8-byte registers, where $1 \leq N \leq 127$ and N specifies the number of 8-byte register write operations. Table 2.3 shows the format for the WRITE N REGISTERS request frame.

Table 2.3 SMP Write N Registers Request Frame

Byte	Bit							
–	7	6	5	4	3	2	1	0
0	SMP Frame Type (0x40)							
1	Function (0xC0)							
2	Reserved							
3	N							
4	StartAdr[18:11]							
5	StartAdr[10:3]							
6	StartByteEnables[7:0]							
7	EndByteEnables[7:0]							
8	(MSB) WriteData[3] (Register at StartAdr)							
9	WriteData[2]							
10	WriteData[1]							
11	WriteData[0] (LSB)							
12	(MSB) WriteData[7] (Register at StartAdr+4)							
13	WriteData[6]							
14	WriteData[5]							
15	WriteData[4] (LSB)							
(N-1) × 8 + 8	(MSB) WriteData[3] (N^{th} Register)							
(N-1) × 8 + 15	WriteData[4] (LSB)							
(N-1) × 8 + 16	(MSB)							
(N-1) × 8 + 19	CRC (LSB)							

Descriptions of these fields follow.

- **StartAdr[18:3]** specifies the starting 8-byte boundary to begin the register write operation.

- **StartByteEnable[7:0]** specifies the active HIGH byte enables for the first register write operation, which targets the registers located at address StartAdr and StartAdr+4. StartByteEnables[3:0] apply to the register at StartAdr. StartByteEnables[7:4] apply to the register at StartAdr+4. This allows for double dword, dword, and byte granularity on the first register write operation. All register writes in between the first and last writes assume a byte enable of 0xFF.
- **EndByteEnable[7:0]** specifies the active HIGH byte enables for the last register write operation. EndByteEnable[3:0] apply to the register located at EndAdr. EndByteEnables[7:4] apply to the register located at EndAdr+4, where EndAdr = StartAdr + [(N-1) $\times$ 8]. This allows for double dword and byte granularity on the last register write operation.
- **WriteData[3]** corresponds to bits [31:24] of the dword to be written to address StartAdr, while **WriteData[0]** corresponds to bits [7:0] of the dword to be written to address StartAdr. **WriteData[7]** corresponds to bits [31:24] of the dword to be written to address StartAdr+4, while **WriteData[4]** corresponds to bits [7:0] of the dword to be written to address StartAdr+4.

Table 2.4 shows the response frame format. The response conditions are:

- If N is not 0x00, and is less than or equal to 127, the Function Result is set to 0x00 to indicate SMP FUNCTION ACCEPTED.
- If N=0, the Function Result is set to 0x02 to indicate SMP FUNCTION FAILED.
- If N > 127, and the frame size is less than 259 dwords, the Function Result is set to SMP FUNCTION FAILED.
- If N > 127 and the frame size is 259 dwords or more, the link layer sends a BREAK sequence, and no SMP Response frame is sent.
- If 0 < N <128 and the size of the Request Frame is not (N $\times$ 2) + 3 dwords, the function result is set to 0xh03 (INVALID REQUEST FRAME LENGTH) and no registers are written.

All register writes in between the first and last addresses assume a byte enable of 0xFF.

Table 2.4 SMP Write N Registers Response Frame

Byte	Bit							
–	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (0x41)							
1	Function (0xC0)							
2	Function Result							
3	Reserved							
4	(MSB) CRC							
7	(LSB)							

2.4.2 READ N REGISTERS (0x40)

This Vendor-Unique function returns the contents of N contiguous 8-byte registers, where $1 \leq N \leq 127$ and N specifies the number of 8-byte register write operations. Table 2.5 shows the format for the READ N REGISTERS request frame.

Table 2.5 SMP Read N Registers Request Frame

Byte	Bit							
–	7	6	5	4	3	2	1	0
0	SMP Frame Type (0x40)							
1	Function (0x40)							
2	Reserved							
3	N							
4	StartAdr[18:11]							
5	StartAdr[10:3]							
6	Reserved							
7	Reserved							
8	(MSB) CRC							
11	(LSB)							

StartAdr[18:3] specifies the starting 8-byte boundary to begin the register read operation.

All register writes in between the first and last addresses assume a byte enable of 0xFF.

Table 2.6 provides the response frame format. The response conditions are:

- If N is not 0x00, and is less than or equal to 127, the Function Result is set to 0x00 to indicate SMP FUNCTION ACCEPTED.
- For all other cases, the Function Result is set to 0x02 to indicate SMP FUNCTION FAILED, and the Response frame is limited to the Response Header and the CRC.

Table 2.6 SMP Read N Registers Response Frame

Byte	Bit							
–	7	6	5	4	3	2	1	0
0	SMP Frame Type (0x41)							
1	Function (0x40)							
2	Function Result (0x00 or 0x02)							
3	Reserved							
4	(MSB) ReadData[3] (Register at StartAdr)							
5	ReadData[2]							
6	ReadData[1]							
7	ReadData[0] (LSB)							
8	(MSB) ReadData[7] (Register at StartAdr+4)							
9	ReadData[6]							
10	ReadData[5]							
11	ReadData[4] (LSB)							
(N-1) × 8 + 4	(MSB) Read Data[3] (N^{th} Register)							
(N-1) × 8 + 7	Read Data[4] (LSB)							
(N-1) × 8 + 8	(MSB)							
(N-1) × 8 +11	CRC (LSB)							

The following describes the Read Response frame fields.

- **ReadData[3]** corresponds to bits [31:24] of the dword read from the register located at address StartAdr, while **ReadData[0]** corresponds to bits [7:0] of the dword read from the register located at address StartAdr.
- **ReadData[7]** corresponds to bits [31:24] of the dword read from the register located at address StartAdr+4, while **ReadData[4]** corresponds to bits [7:0] of the dword read from the register located at address StartAdr+4.

If the size of the Request Frame is not 3 dwords, the Function Result is 0x03 to indicate INVALID REQUEST FRAME LENGTH, and no registers are read.

2.5 Boot Loader, Serial EEPROM, and API2C Interface

The Boot Loader reads a record of configuration from an external serial EEPROM and distributes that information to various functional blocks within the expander.

The boot record contains two sections. The first section is a fixed field of data that must be read to enable the part. The second section provides optional data where both the address and data to be written is specified. Both sections are covered by a checksum. The sum of all data and the checksum equals zero. The null pointer following the optional data field and the second checksum are required regardless of the inclusion of optional data commands.

Boots are retried up to seven times if errors occur. If the boot ultimately fails then the phys are disabled and the LSISASx12 generates an interrupt to the Enclosure Processor.

The optional data field can configure GPIO/LEDs, nonstandard time-outs, or any register that this document describes.

The configuration manager uses an AMBA™ Peripheral Inter-IC (API2C) interface to read the serial EEPROM. To allow up to four expanders to access unique configuration records from a single serial EEPROM, the API2C serial EEPROM loader uses ISTWI_ADDR[1:0] to specify the base vector of the configuration record. ISTWI_ADDR[1:0] provide the least significant bits of the base vector address.

The API2C interface operates as a master only with 100 kHz or 400 kHz memories that support 16-bit addressing. The base vector is always read at a 100 kHz rate. If the most significant bit is set, the remaining data is read at 400 kHz. If a failure occurs, the configuration manager retries the read. After three retries, all subsequent read attempts are at 100 kHz.

Figure 2.3 shows the format of the boot record. The data fields in the boot record are 8 bits wide.

Figure 2.3 Boot Record Format

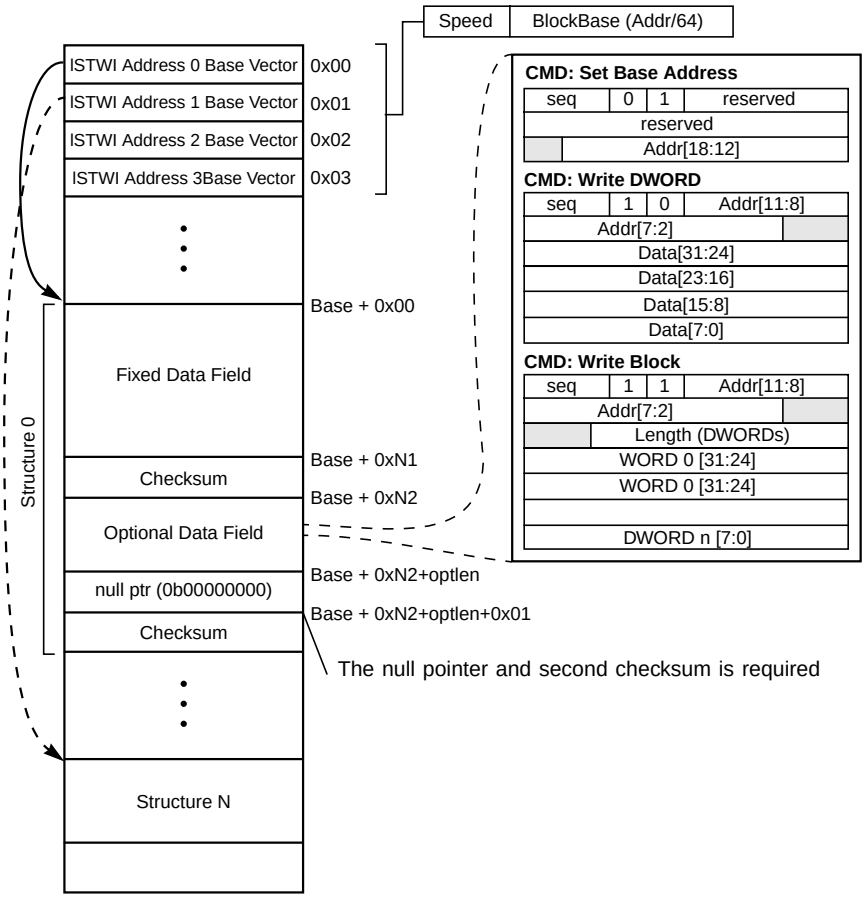


Table 2.7 shows the contents of the fixed data field within the boot record. The first byte of the fixed data record is a “Safety Byte.” This byte is the XOR of the 2-bit ISTWI_ADDR[1:0] replicated four times, and the value 0x51. The safety byte is not written to the register bus interface, but is included in the fixed data record checksum.

Table 2.7 Boot Record Format – Fixed Data Field

Destination	Address	SEEPROM Offset	Field	
–	–	0	Safety Byte	Safety Byte
CfgMgr	0x00000	1	WWN Base Address	WWN[63:56]
CfgMgr		2		WWN[55:48]
CfgMgr		3		WWN[47:40]
CfgMgr		4		WWN[39:32]
CfgMgr	0x00004	5		WWN[31:24]
CfgMgr		6		WWN[23:16]
CfgMgr		7		WWN[15:8]
CfgMgr		8		WWN[7:0]
CfgMgr	0x00008	9	Vendor ID	Vendor ID[63:56]
CfgMgr		10		Vendor ID[55:48]
CfgMgr		11		Vendor ID[47:40]
CfgMgr		12		Vendor ID[39:32]
CfgMgr	0x0000C	13		Vendor ID[31:24]
CfgMgr		14		Vendor ID[23:16]
CfgMgr		15		Vendor ID[15:8]
CfgMgr		16		Vendor ID[7:0]
CfgMgr	0x00010	17	Product ID	Product ID[127:120]
CfgMgr		18		Product ID[119:112]
CfgMgr		19		Product ID[111:104]
CfgMgr		20		Product ID[103:96]
CfgMgr	0x00014	21	Product ID (cont.)	Product ID[95:88]
CfgMgr		22		Product ID[87:80]
CfgMgr		23		Product ID[79:72]
CfgMgr		24		Product ID[71:64]

Table 2.7 Boot Record Format – Fixed Data Field (Cont.)

Destination	Address	SEEPROM Offset	Field	
CfgMgr	0x00018	25	Product ID (cont.)	Product ID[63:56]
CfgMgr		26		Product ID[55:48]
CfgMgr		27		Product ID[47:40]
CfgMgr		28		Product ID[39:32]
CfgMgr	0x0001C	29	Product ID (cont.)	Product ID[31:24]
CfgMgr		30		Product ID[23:16]
CfgMgr		31		Product ID[15:8]
CfgMgr		32		Product ID[7:0]
CfgMgr	0x00020	33	Product Revision	Product Revision[31:24]
CfgMgr		34		Product Revision[23:16]
CfgMgr		35		Product Revision[15:8]
CfgMgr	0x00040	36	Spin-up Control	Spin Delay[15:8]
CfgMgr		37		Spin Delay[7:0]
CfgMgr		38		Spin Number
CfgMgr		39		Spin Mode
CfgMgr	0x00044	40	Phy Config	Reserved
CfgMgr		41		Reserved
CfgMgr		42		SMP Target Enable, STP Target Enable, Phy Enable[11:8]
CfgMgr		43		Phy Enable[7:0]
CfgMgr	0x00048	44	Reserved Must write to 0x0000.	Reserved
CfgMgr		45		Reserved
CfgMgr		46		Reserved
CfgMgr		47		Reserved

Table 2.7 Boot Record Format – Fixed Data Field (Cont.)

Destination	Address	SEEPROM Offset	Field	
CfgMgr	0x0004c	48	TX/RX Polarity	Reserved
CfgMgr		49		Reserved
CfgMgr		50		TX Polarity[11:8]
CfgMgr		51		TX Polarity[7:0]
CfgMgr	0x00050	52		Reserved
CfgMgr		53		Reserved
CfgMgr		54		RX Polarity[11:8]
CfgMgr		55		RX Polarity[7:0]
Phy (all)	0x18304	56	GigaBlaze RX Config	1E
Phy (all)		57		1E
Phy (all)		58		1E
Phy (all)		59		1E
Phy (all)	0x18308	60	GigaBlaze TX Config (SAS)	28
Phy (all)		61		2C
Phy (all)		62		28
Phy (all)		63		2A
Phy (all)	0x1830C	64	GigaBlaze TX Config (SATA)	28
Phy (all)		65		25
Phy (all)		66		28
Phy (all)		67		14
ECM	0x60000	68	Expander Connection Manager Registers Configuration	ECM Config 0
ECM	0x60004	69		ECM Remote Bank Enable 0 [11:8]
ECM		70		ECM Remote Bank Enable 0 [7:0]
ECM	0x60010	71		ECM Config 1
ECM	0x60014	72		ECM Remote Bank Enable 1 [11:8]
ECM		73		ECM Remote Bank Enable 1 [7:0]
ECM	0x60020	74		ECM Config 2
ECM	0x60024	75		ECM Remote Bank Enable 2 [11:8]
ECM		76		ECM Remote Bank Enable 2 [7:0]

Table 2.7 Boot Record Format – Fixed Data Field (Cont.)

Destination	Address	SEEPROM Offset	Field	
ECM	0x60030	77	Expander Connection Manager Registers Configuration (cont)	ECM Config 3
ECM	0x60034	78		ECM Remote Bank Enable 3 [11:8]
ECM		79		ECM Remote Bank Enable 3 [7:0]
ECM	0x60040	80		ECM Config 4
ECM	0x60044	81		ECM Remote Bank Enable 4 [11:8]
ECM		82		ECM Remote Bank Enable 4 [7:0]
ECM	0x60050	83		ECM Config 5
ECM	0x60054	84		ECM Remote Bank Enable 5 [11:8]
ECM		85		ECM Remote Bank Enable 5 [7:0]
ECM	0x60060	86	Expander Connection Manager Registers Configuration (cont)	ECM Config 6
ECM	0x60064	87		ECM Remote Bank Enable 6 [11:8]
ECM		88		ECM Remote Bank Enable 6 [7:0]
ECM	0x60070	89		ECM Config 7
ECM	0x60074	90		ECM Remote Bank Enable 7 [11:08]
ECM		91		ECM Remote Bank Enable 7 [7:0]
ECM	0x60080	92		ECM Config 8
ECM	0x60084	93		ECM Remote Bank Enable 8 [11:8]
ECM		94		ECM Remote Bank Enable 8 [7:0]
ECM	0x60090	95	Expander Connection Manager Registers Configuration (cont)	ECM Config 9
ECM	0x60094	96		ECM Remote Bank Enable 9 [11:8]
ECM		97		ECM Remote Bank Enable 9 [7:0]
ECM	0x600A0	98		ECM Config 10
ECM	0x600A4	99		ECM Remote Bank Enable 10 [11:8]
ECM		100		ECM Remote Bank Enable 10 [7:0]
ECM	0x600B0	101		ECM Config 11
ECM	0x600B4	102		ECM Remote Bank Enable 11 [11:8]
ECM		103		ECM Remote Bank Enable 11 [7:0]

Table 2.7 Boot Record Format – Fixed Data Field (Cont.)

Destination	Address	SEEPROM Offset	Field	
SMP Target	0x10020	104	Route Table Configuration	Route Table Offset 0 [7:0]
SMP Target		105		Route Table Size 0 [7:0]
SMP Target	0x10024	106		Route Table Offset 1 [7:0]
SMP Target		107		Route Table Size 1 [7:0]
SMP Target	0x10028	108		Route Table Offset 2 [7:0]
SMP Target		109		Route Table Size 2 [7:0]
SMP Target	0x1002C	110	Route Table Configuration (cont)	Route Table Offset 3 [7:0]
SMP Target		111		Route Table Size 3 [7:0]
SMP Target	0x10030	112		Route Table Offset 4 [7:0]
SMP Target		113		Route Table Size 4 [7:0]
SMP Target	0x10034	114		Route Table Offset 5 [7:0]
SMP Target		115		Route Table Size 5 [7:0]
SMP Target	0x10038	116	Route Table Configuration (cont)	Route Table Offset 6 [7:0]
SMP Target		117		Route Table Size 6 [7:0]
SMP Target	0x1003C	118		Route Table Offset 7 [7:0]
SMP Target		119		Route Table Size 7 [7:0]
SMP Target	0x10040	120		Route Table Offset 8 [7:0]
SMP Target		121		Route Table Size 8 [7:0]
SMP Target	0x10044	122	Route Table Configuration (cont)	Route Table Offset 9 [7:0]
SMP Target		123		Route Table Size 9 [7:0]
SMP Target	0x10048	124		Route Table Offset 10 [7:0]
SMP Target		125		Route Table Size 10 [7:0]
SMP Target	0x1004C	126		Route Table Offset 11 [7:0]
SMP Target		127		Route Table Size 11 [7:0]
Fixed Data Checksum	0x10050	128	Checksum	Checksum

2.6 GPIO/LED and SIO Configuration

This section describes the use and configuration of the GPIO, LED, and SIO signals.

2.6.1 GPIO/LED Configuration

The LSISASx12 provides a total of 52 LED and GPIO signals. There are four GPIO signals (GPIO[3:0]) and 48 LED signals (LED_ACTIVITY[11:0]/, LED_FAULT[11:0]/, LED_STATUS[11:0]/, and CABLE_DET[11:0]/). The LED signals are associated with the SAS ports on the device, but can also be used as GPIO signals. The configuration manager LED control registers can configure the LED and GPIO signals for a variety of status functions. Figure 2.4 shows the control structure for the LSISASx12 GPIO/LED pins.

Each SAS port has four LED signals. Three of these signals are typically configured to drive LEDs that indicate device activity, fault, and status. The fourth signal can be configured to provide a cable detect input. The LSISASx12 also provides four GPIO pins, GPIO[3:0], which are not associated with a phy. For ease of reference, Figure 2.5 associates each functional GPIO and LED signal group with a group number.

Figure 2.4 LSISASx12 GPIO/LED Control Structure

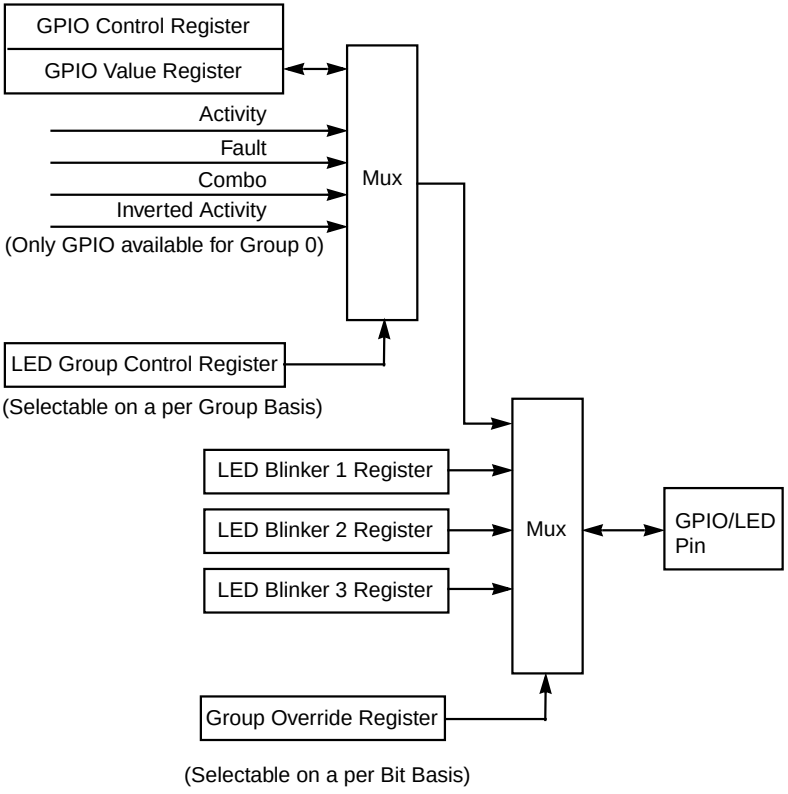
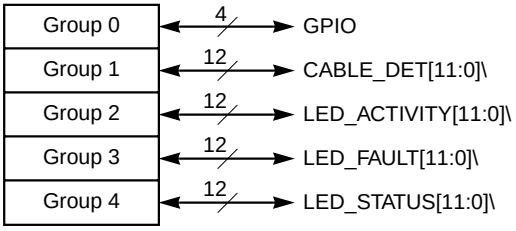


Figure 2.5 LSISASx12 GPIO/LED Pins



The user can configure all GPIO and LED signal groups as GPIO signals. The Group GPIO Control and Group GPIO Value register bits individually set the direction and value of each signal.

The LED signals support all the functionality listed in Table 4.3. GPIO[3:0] support only the GPIO functionality, and do not support the other functionalities listed in Table 4.3. The LED Group Control register configures the Group 1–4 signals for the Activity, Fault, Combo, or Inverted Activity setting.

Note: The Group 1–4 signals support all the functionalities listed in Table 4.3, despite their pin names.

The Group Override registers can assign three independent blinker definitions to any GPIO/LED signal, including GPIO[3:0]. The Group Override register blinker settings override other configuration settings for the signal. Each blinker definition consists of a 30-bit rotating shift register. The user can configure the blink pattern by programming the shift register, which can shift every 16 ms or 64 ms. Bit 0 of the shift register determines the state of the signal.

The LSISASx12 provides LEDSYNCIN and LEDSYNCOUT pins to synchronize the LED blink patterns between multiple LSISASx12 devices. The LEDSYNCOUT pin emits a 0.5 Hz square wave, which can be applied to the LEDSYNCIN pin on other LSISASx12 devices to synchronize blinker patterns on the devices. Each blinker definition has the ability to override the blink pattern and use the 0.5 Hz LEDSYNCOUT signal to drive the signals that the blinker definition controls.

2.6.2 SIO Configuration

This section describes the enhanced SGPIO interface that is available on the LSISASx12A expander. The enhanced SGPIO interface is not available on the LSISASx12 expander.

The SPGIO interface on the LSISASx12A can operate in a standard SGPIO mode or in an enhanced SGPIO mode. The standard SGPIO mode provides the same SGPIO interface that the LSISASx12 expander provides. The enhanced SPGIO mode provides three programmable output data control registers for each phy, and four pattern definition registers. Section 4.2, “Configuration Manager Registers,” on page 4-3 describes these registers.

The SIO Control Select bit in the SIO Adapter Control register enables the enhanced SPGIO interface. Clearing this bit places the LSISASx12A expander in the standard SGPIO mode. Setting this bit places the

LSISASx12A expander in the enhanced SGPIO mode. Table 2.8 defines the encoding of the SIO Control Select bit.

Table 2.8 SIO Control Select Bit Encoding

SIO Control Select	SIO Mode
0	Standard SGPIO Mode
1	Enhanced SGPIO Mode

2.6.3 SGPIO Pattern Generation

This section describes how to program the enhanced SGPIO interface on the LSISASx12A expander.

There are four programmable pattern definition registers that define the LED blink pattern (SIO Pattern Definition 3-0). Pages 4-67 through 4-69 define these registers. The pattern definition registers are rotating shift registers that shift to the right once per time interval. Each pattern generator uses a unique, programmable 20-bit pattern and a time base interval of either 25 ms or 100 ms. The Timebase bit in the pattern definition register determines the time base interval for the given pattern definition register. Table 2.9 provides the definition of the Timebase bit.

Table 2.9 Timebase Bit Definition

Timebase Bit Setting	Shift frequency
0	25 ms
1	100 ms

The Activity Stretch Control register and SIO Adapter Control registers provide control of the minimum assertion time, maximum assertion time, minimum deassertion time, and off time following a maximum assertion.

Several examples of programming the enhanced SGPIO interface follow.

Example 1 – This example uses the SIO Pattern Definition 0 register to create a repeating pattern of 125 ms ON and 125 ms OFF.

- Set the SIO Control Select bit in the SIO Adapter Control register to 0b1 to enable the enhanced SGPIO interface.
- Set the Timebase bit to 0b0 to program a 25 ms time base.

– Program the Pattern Definition 0 register to 0b00000111110000011111.

Example 2 – This example uses the SIO Pattern Definition 3 register create a repeating pattern of 1500 ms OFF and 500 ms ON.

– Set the SIO Control Select bit in the SIO Adapter Control register to 0b1 to enable the enhanced SGPIO interface.

– Set the Timebase bit to 0b1 to program a time base of 100 ms.

– Program Pattern Definition 3 to 0b11111000000000000000.

2.7 STP Enclosure Management Interface

This section describes the STP interface. This block contains the I^2C clock, data, and interrupt signals. The enclosure management bridge (EMB) provides data integrity checking on the I^2C bus.

The STP Target consists of:

- STP Link Layer
- STP Port Layer
- STP Transport Layer
- STP Application Layer

The STP Target is responsible for the following tasks:

- Accepting STP OAFs.
- Rejecting OAFs when the
  - OAF protocol is not STP
  - OAF initiator bit is cleared to 0b0
  - OAF Features bits [3:0] are not cleared to 0b0000
  - STP Target has an existing Initiator Affiliation, but not with the requesting STP Initiator.
- Managing STP Affiliations.
- Receiving SATA Frame Information Structures (FIS).
- Checking CRC on inbound SATA FIS.

- Positively decoding and notifying the SEP to process the following mandatory ATAPI commands or sequences:
  - IDENTIFY PACKET DEVICE (0xA1)
  - PACKET (0xA0)
  - DEVICE RESET (0x08)
  - EXECUTE DEVICE DIAGNOSTICS (0x90)
  - Soft Reset sequence
- Positively decoding the following mandatory ATAPI commands and returning the appropriate ATAPI status without SEP intervention:
  - SET FEATURES (0xEF)
  - IDENTIFY DEVICE (0xEC)
  - READ SECTOR (0x20)
- Negatively decoding and aborting the ATAPI NOP command and the following mandatory ATAPI Power Management Feature set commands:
  - NOP (0x00)
  - CHECK POWER MODE (0xE5)
  - SLEEP (0xE6)
  - STANDBY (0xE2)
  - STANDBY IMMEDIATE (0xE0)
  - IDLE (0xE3)
  - IDLE IMMEDIATE (0xE1)
- Negatively decoding and aborting all non-mandatory ATAPI commands.
- Closing a connection after the sending ATAPI command status through the D2H Register FIS.
- Closing a connection after the receipt of the SCSI CDB associated with PACKET commands.
- Generating interrupts to the external SEP when commands and data are available for the SEP to process.
- Generating and transmitting SATA FISes (D2H Register, PIO Setup, and Data FISes).

- Generating CRC for outbound SATA FISes.
- Generating and transmitting OAFs when a SATA FIS is ready to be sent to the STP Initiator.
- Responding to PMREQ_S and PMREQ_P primitives with PMNAK.

2.7.1 Doorbell Interface

The STP EMB doorbell messaging interface provides a communication channel between the STP target and the SEP. The LSISASx12 STP sets the various bits in the STP-SEP Doorbell registers to send commands to the SEP. The STP-SEP Doorbells are the interrupt sources.

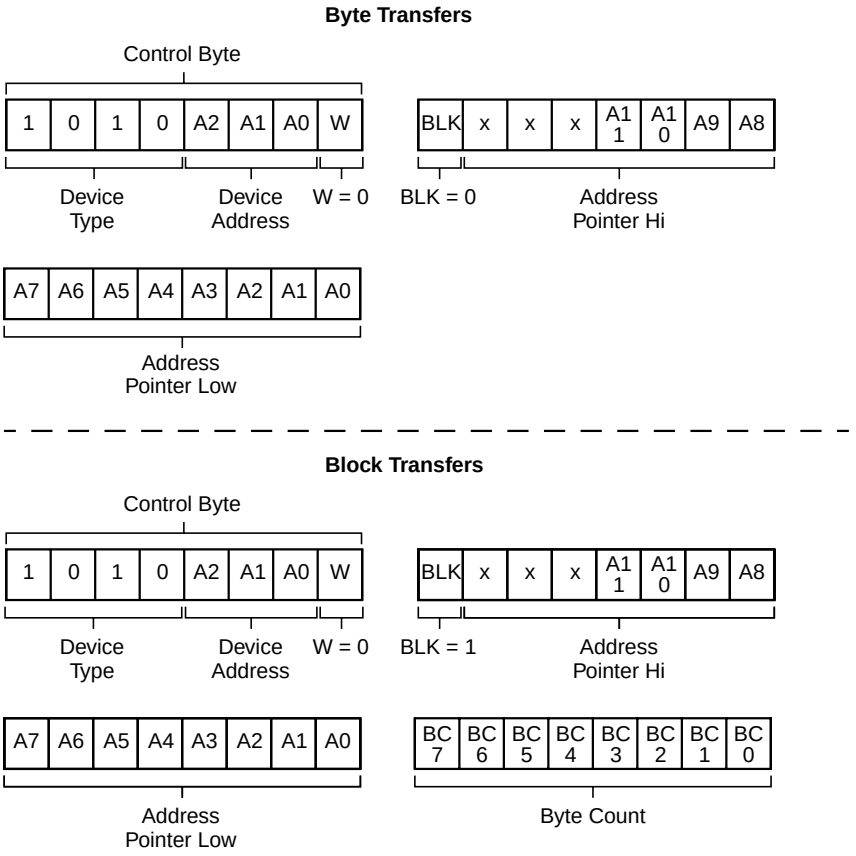
The SEP sets bits in the SEP-STP Doorbell register, which are inputs to the LSISASx12 STP.

2.7.2 I^2C Addressing and Transfer Modes

The STP is accessed through the I^2C. The device addressing for the I^2C slave behaves in a similar manner to industry standard serial EEPROM devices. The first byte on the bus following a start condition provides the control byte. The control byte consists of a 4-bit device type code that is set to 0b1010. Following the device type code is a 3-bit device address (A2, A1, A0). Bit A2 is hard-coded to a value of 0, and ISTWI_ADDR[1:0] provide bit A1 and bit A0 respectively. The last bit of the control byte indicates if the operation is a read or write. When set to one (1), the operation is a read. When cleared to zero (0), the operation is a write.

Figure 2.6 provides an illustration of the slave addressing and transfer modes. The I^2C can transfer data to/from the I^2C slave using byte transfers or block transfers. The most-significant bit of the Address Pointer Hi byte (BLK) selects the transfer mode of the current operation. Clearing the BLK bit selects the byte transfer method. Setting the BLK bit selects the block transfer method. For block transfers, the third data byte provides the byte count. A byte count of zero indicates a 256 byte transfer.

Figure 2.6 I²C Slave Addressing and Transfer Modes



2.7.3 I²C Error Detection

Because the I²C bus is susceptible to errors in noisy environments, the EMB transmits CRC bytes with each data transfer. The algorithm uses an 8-bit CRC that is calculated using each byte in the data transfer, including the control byte. The polynomial used for the CRC is: $x^8 + x^2 + x + 1$.

Device users can enable or disable CRC checking by writing to the Error Detection Control register. If CRC checking is disabled, then the LSISASx12 still transmits CRC bytes during data transfers. CRC checking is enabled by default after reset. When CRC checking is

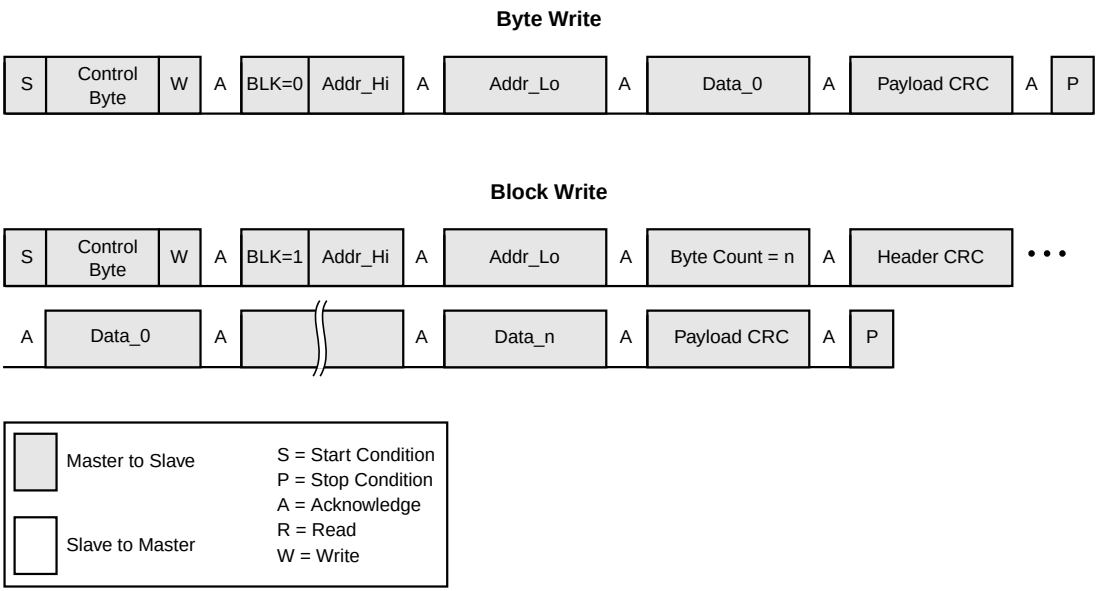
enabled and the LSISASx12 detects a bad CRC value, the LSISASx12 asserts the EMB_CRC_INT signal to notify the SEP of the error.

2.7.4 I²C Slave Write Operations

Figure 2.7 illustrates an I²C slave write operation. For byte write transfers, the fourth data byte of the transfer contains the CRC. The CRC is calculated using the control byte (0b1010_0$A_1A_0$0), the Address_Hi byte, Address_Lo byte, and the data byte. The LSISASx12 does not commit the data until it validates the payload CRC.

Block transfers use two CRC bytes: the header CRC and the payload CRC. The header CRC covers the address and the byte count; the payload CRC covers the data. The fourth byte of the transfer contains the header CRC. If the LSISASx12 detects an error in the address or byte count, it asserts the EMB_CRC_INT signal. The data transfer can continue, but the LSISASx12 discards the data. The LSISASx12 resets the CRC calculation for the data transfer portion of the block write operation. The payload CRC byte is the last data byte before the I²C master signals a stop condition.

Figure 2.7 Slave Write Operation

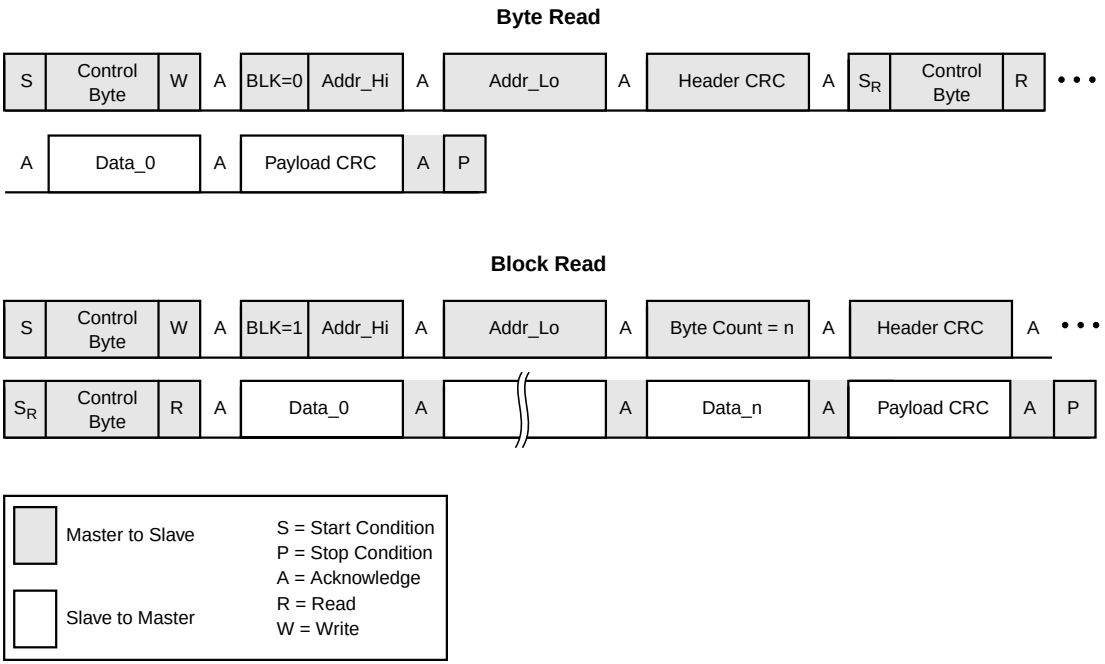


2.7.5 I²C Slave Read Operations

Figure 2.8 illustrates I²C read operations. For byte read transfers, the header CRC byte follows the Address_Hi and Address_Lo bytes. The header CRC is calculated using the control byte and the two address bytes. The payload CRC byte follows the read data byte and is calculated using only the read data byte.

For block read transfers, the header CRC byte follows the Address_Hi, Address_Lo, and Byte_Count bytes. The header CRC is calculated using the control byte and the two address bytes. The payload CRC byte follows the last byte of read data, and is calculated using the read data bytes only.

Figure 2.8 Slave Read Operation



2.7.6 ATAPI Command Status

Table 2.10 provides the expected results for common ATAPI commands. These commands allow the EMB to access a bridge and to operate with an ATAPI host.

Table 2.10 Expected Doorbells and SATA FISes

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
NOP (00) (This command is always aborted.)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1
SET FEATURES (0xEF) (Features = 0x03 and SectorCnt ≤ 0x0F)	None	None	None	Error[ABRT] = 0 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 0 I = 1
SET FEATURES (0xEF) (Features != 0x03) or (SectorCnt > 0x0F)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1
READ SECTOR (0x20)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
IDENTIFY DEVICE (0xEC)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
CHECK POWER MODE (0xE5) SLEEP (0xE6) STANDBY (0xE2) STANDBY IMM (0xE0) IDLE (0xE3) IDLE IMM (0xE1)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1
DEVICE RESET (0x08) (No SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 0x01 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0x01 Status = 0x00 I = 0 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
DEVICE RESET (0x08) (SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 8'h00 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0x01 Status = 0x00 I = 0 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
EXEC DEV DIAGNOSTICS (0x90) (No SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 0x01 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0xh01 Status = 0x00 I = 1 SectorCnt = 0xh01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
EXEC DEV DIAGNOSTICS (0x90) (SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 0x00 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0x00 Status = 0x00 I = 1 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
Software Reset (SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 0x00 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0x00 Status = 0x00 I = 0 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
Software Reset (No SEP Diag. Error)	StpDbell[2] = 1 (RST_CMD_RCVD) SepDbell[5] = 0	StpDbell[2] = 0 CmdStatus = 0x01 SepDbell[5] = 1 (STATUS_RDY)	None	Error = 0x01 Status = 0x00 I = 0 SectorCnt = 0x01 SectorNum = 0x01 CylinderLo = 0x14 CylinderHi = 0xEB
IDENTIFY PACKET DEVICE (0xA1)	StpDbell[1] = 1 (ID_PCKT_DEV_RCVD) SepDbell[4] = 0	StpDbell[1] = 0 SepDbell[4] = 1 (RD_DATA_RDY)	D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 0 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd512	None
PACKET (A0h) (w/ Features[DMA] = 1)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1
PACKET (A0h) (w/ Features[OVL] = 1)	None	None	None	Error[ABRT] = 1 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 1 I = 1

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (A0h) CDB = Test Unit Ready (00h) (MaxByteCnt > 16'd1024) Note: MaxByteCnt[15:0] = {CylinderHi, CylinderLo}	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[5] = 0	StpDbell[0] = 0 CmdStatus[7:0] = {Sense Key[3:0], 4'b0} SepDbell[5] = 1 (STATUS_RDY)	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12	(Status Transfer) I = 1 Error[7:0] = CmdStatus[7:0] Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = (SenseKey !=0)

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Send Diagnostic (0x1D) Xfer Length = 1 Kbyte (MaxByteCnt > 16'd1024)	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[0] = 0 StpDbell[3] = 1 (WRITE_DATA_RCVD) SepDbell[3] = 0 SepDbell[5] = 0	StpDbell[0] = 0 SepXferLen = 8'h00 (256 dwords) XferAttr[1:0] = 0b10 (LastXfer / Out) SepDbell[0] = 1 (PCKT_CMD_ACK) StpDbell[3] = 0 SepDbell[3] = 1 (WRITE_DATA_ACK) CmdStatus[7:0] = {Sense Key[3:0], 0b0000} SepDbell[5] = 1 (STATUS_RDY)	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12 (Data Transfer) D = 0 (Data-Out) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd1024	(Status Transfer) I = 1 Error[7:0] = CmdStatus[7:0] Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = (SenseKey !=0)

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Send Diagnostic (0x1D) Xfer Length = 1.5 Kbytes (MaxByteCnt > 16'd1024)	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD)	 StpDbell[0] = 0 SepXferLen = 0x00 (256 dwords) XferAttr[1:0] = 0b00 (Not LastXfer / Out) SepDbell[0] = 1 (PCKT_CMD_ACK)	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12	
	SepDbell[0] = 0 StpDbell[3] = 1 (WRITE_DATA_RCVD)	 StpDbell[3] = 0 SepXferLen = 0x80 (128 dwords) XferAttr[1:0] = 0b10 (LastXfer / Out) SepDbell[3] = 1 (WRITE_DATA_ACK)	(Data Transfer) D = 0 (Data-Out) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd1024	
	SepDbell[3] = 0 StpDbell[3] = 1 (WRITE_DATA_RCVD) SepDbell[3] = 0	 StpDbell[3] = 0 SepDbell[3] = 1 (WRITE_DATA_ACK)	(Data Transfer) D = 0 (Data-Out) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd512	(Status Transfer) I = 1 Error[7:0] = CmdStatus[7:0] Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = (SenseKey !=0)
	SepDbell[5] = 0	CmdStatus[7:0] = {Sense Key[3:0], 4'b0} SepDbell[5] = 1 (STATUS_RDY)		

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Rcv Diagnostic Results (0x1C) Xfer Length = 1 Kbyte (MaxByteCnt > 16'd1024)	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[0] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED)	 StpDbell[0] = 0 SepXferLen = 0x00 (256 dwords) XferAttr[1:0] = 0b11 (LastXfer / In) SepDbell[0] = 1 (PCKT_CMD_ACK) StpDbell[4] = 0	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd1024	 (Status Transfer) I = 1 Error[7:0] = 0x00 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 0

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Rcv Diagnostic Results (0x1C) Xfer Length = 1.5 Kbytes (MaxByteCnt > 16’d1024)	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[0] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED) SepDbell[4] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED)	StpDbell[0] = 0 SepXferLen = 0x00 (256 dwords) XferAttr[1:0] = 0b01 (Not LastXfer / In) SepDbell[0] = 1 (PCKT_CMD_ACK) StpDbell[4] = 0 SepXferLen = 0x80 (128 dwords) XferAttr[1:0] = 0b11 (LastXfer / In) SepDbell[4] = 1 (RD_DATA_RDY) StpDbell[4] = 0	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16’d12 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16’d1024 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16’d512	(Status Transfer) I = 1 Error[7:0] = 0x00 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 0

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Request Sense (0x03) Xfer Length = 20 bytes (MaxByteCnt > 16'd1024)	StpXferLen = 0x00 (256 dwords) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[0] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED)	 StpDbell[0] = 0 SepXferLen = 0x05 (5 dwords / 20 bytes) XferAttr[1:0] = 0b11 (LastXfer / In) SepDbell[0] = 1 (PCKT_CMD_ACK) StpDbell[4] = 0	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd20	 (Status Transfer) I = 1 Error[7:0] = 0x00 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 0

Table 2.10 Expected Doorbells and SATA FISes (Cont.)

ATAPI Command	Expected STP Target Doorbell Writes	Expected SEP Doorbell Writes	Expected SATA PIO Setup FISes	Expected SATA D2H Reg. FIS
PACKET (0xA0) CDB = Rcv Diagnostic Results (0x1C) Xfer Length = 800 bytes (MaxByteCnt = 16'd512)	StpXferLen = 0x80 (128 dwords / 512 bytes) StpDbell[0] = 1 (PCKT_CMD_RCVD) SepDbell[0] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED) SepDbell[4] = 0 StpDbell[4] = 1 (RD_DATA_XMITTED)	StpDbell[0] = 0 SepXferLen = 0x80 (128 dwords) XferAttr[1:0] = 0b01 (Not LastXfer / In) SepDbell[0] = 1 (PCKT_CMD_ACK) StpDbell[4] = 0 SepXferLen =0x80 (128 dwords) XferAttr[1:0] = 0b11 (LastXfer / In) SepDbell[4] = 1 (RD_DATA_RDY) StpDbell[4] = 0	(CDB Transfer) D = 0 (Data-Out) I = 0 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd12 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd512 (Data Transfer) D = 1 (Data-In) I = 1 IStatus[BSY] = 0 IStatus[DRDY] = 1 IStatus[DRQ] = 1 EStatus[BSY] = 1 EStatus[DRDY] = 1 EStatus[DRQ] = 0 XferCnt = 16'd288	(Status Transfer) I = 1 Error[7:0] = 0x00 Status[DRDY] = 1 Status[BSY] = 0 Status[DRQ] = 0 Status[ERR] = 0

Chapter 3
Signal Descriptions

This chapter describes the LSISASx12 expander signals and consists of the following sections.

A slash (/) at the end of a signal indicates that the signal is active LOW. When the slash is absent, the signal is active HIGH. NC designates a No Connection signal. NC signals do not connect inside of the LSISASx12 package.

3.1 Signal Types

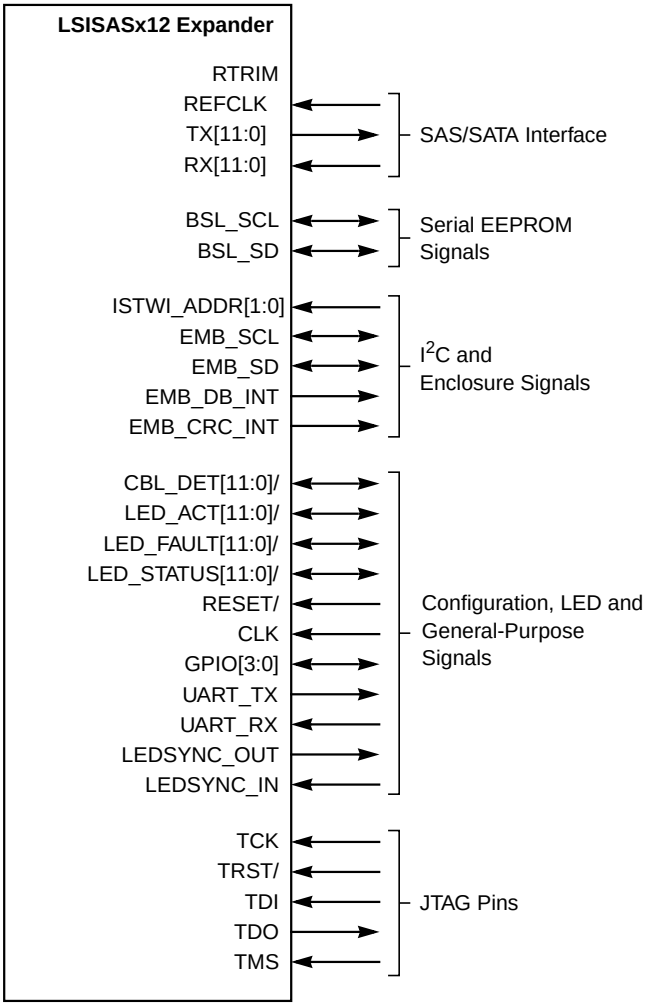
There are five signal types:

I	Input, a standard input-only signal
O	Output, a standard output driver
I/O	Input and output (bidirectional)
P	Power
G	Ground

3.2 Functional Signal Grouping

Figure 3.1 contains the functional signal groupings of the LSISASx12.

Figure 3.1 LSISASx12 Signal Functional Grouping[1]



1. The SIO signals are multiplexed and do not appear in Figure 3.1. Refer to Section 3.7, “Multiplexed SIO Interface,” on page 3-7 for the SIO signal definitions.

3.3 SAS/SATA Signals

Table 3.1 describes the SAS/SATA signals.

Table 3.1 SAS/SATA Signal Description

Signal Name	BGA Position	I/O	Description
REFCLK	AC13, AB13	I	The **Reference Clock** signal provides the serial differential clock to the LSISASx12 expander. Connect a 75 MHz oscillator having an accuracy of at least ±50 ppm to these pins. To use a single-ended oscillator, REFCLK+ and REFCLK– must be driven through a single-ended to differential level-shifting network. Refer to SEN S11054: LSISASx12 Design Considerations (Document Number: DB05-000116-xx) for information concerning this signal.
RTRIM	AA6	–	The **Resistor Reference** provides the analog resistor reference for the GigaBlaze termination logic.
RX[11:0]+[1]	AC2, W3, P1, L2, F2, C2, C25, E25, J25, N25, V24, AB24	I	The **Receive Differential Data** signals provide the differential data receiver for the respective phy[n].
RX[11:0]–[1]	AC1, W2, N1, L3, F3, B2, C24, E24, J24, N24, V25, AB25		
TX[11:0]+[1]	AC3, AA1, U1, P2, G1, C1, B26, G26, L26, T25, AA26, AC25	O	The **Transmit Differential Data** signals provide the differential data transmit signal for the respective phy[n].
TX[11:0]–[1]	AC4, AB1, V1, P3, H1, D1, C26, H26, M26, T24, AB26, AC24		

1. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy through the Phy Transmit Polarity and Phy Receiver Polarity registers.

3.4 Serial EEPROM Signals

Table 3.2 describes the serial EEPROM bus signals.

Table 3.2 Serial EEPROM Interface Signal Description

Signal Name	BGA Position	I/O	Description
BSL_SCL	AE3	I/O	The **Serial EEPROM Interface Clock** signal provides the clock signal for the serial EEPROM.
BSL_SD	AD4	I/O	The **Serial EEPROM Interface Data** signal provides the data signal for the serial EEPROM.

3.5 I^2C and EMB Signals

Table 3.3 describes the I^2C and EMB signals.

Table 3.3 I^2C and EMB Signal Description

Signal Name	BGA Position	I/O	Description
EMB_SCL	AE4	I/O	The **EMB Clock** pin provides the clock signal for the I^2C bus.
EMB_SD	AF3	I/O	The **EMB Data** pin provides the data signal for the I^2C bus.
EMB_DB_INT	AF4	O	The **Enclosure Management Doorbell Interrupt** pin provides the active HIGH doorbell interrupt from the enclosure management bridge to the external enclosure services processor.
EMB_CRC_INT	AD5	O	The **Enclosure Management CRC Error Interrupt** pin provides the active HIGH CRC error interrupt from the enclosure management bridge to the external enclosure services processor.
ISTWI_ADDR[1:0]	AA8, AE5	I	The **I^2C Addresses** This bus serves the following functions: • Provides the lower two bits of the address to which the enclosure management bridge responds. The additional address bits are hard-coded. This structure enables a single enclosure services processor to address multiple expanders. • Provides the block offset of the configuration data in the serial EEPROM memory to the boot strap loader. This structure enables a single serial EEPROM to contain separate configuration data for up to four LSISASx12 expanders.

3.6 Configuration and General-Purpose Signals

Table 3.4 describes a possible configuration of the general-purpose signals.

Table 3.4 Configuration and General-Purpose Signal Description

Signal Name	BGA Position	I/O	Description
CBL_DET[11:0]/	A5, A6, A7, A8, C8, B8, A16, D14, B16, C16, D15, E16	I/O	The **Cable Detection** pins can be configured to detect the presence of a cable on the associated phy. These pins can also be configured as activity LEDs, fault LEDs, status LEDs, or independent GPIO signals.
LED_ACT[11:0]/	AD6, AC7, AE6, AF5, AF6, AF7, AF21, AA18, AF22, AE22, AD22, AF23	I/O	The **Activity LED** pins can be configured to indicate serial port activity by driving an LED. These pins can also be configured as cable detection LEDs, fault LEDs, status LEDs, or independent GPIO signals.
LED_FAULT[11:0]/	AC9, AD9, AE9, AB11, AD10, AC12, AE23, AF24, AA20, AF25, AB21, AD23	I/O	The **Fault LED** pins can be configured to indicate a serial port fault by driving an LED. These pins can also be configured as cable detection LEDs, activity LEDs, status LEDs, or independent GPIO signals.
LED_STATUS[11:0]/	AF9, AF16, AF17, AF18, AB15, AE18, AD18, AE19, AB17, AF19, AF20, AB18	I/O	The **Status LED** pins can be configured to indicate disk drive activity by driving an LED. These pins can also be configured as cable detection LEDs, activity LEDs, fault LEDs, or independent GPIO signals. LED_STATUS[11:6]/ are multiplexed to the SIO interface. Refer to Section 3.7, "Multiplexed SIO Interface," for the SIO interface signal definitions.
RESET/	AD14	I	Asserting the **Reset** pin (which is an active LOW signal) forces the chip into a Power-On-Reset (POR) state.
CLK	AE11	I	The **Clock** pin provides 75 MHz clock for the internal control logic. Refer to SEN S11054: LSISASx12 Design Considerations (DB05-000116-xx) for information concerning this signal.
GPIO[3:0]	AC23, AE24, AA21, AD24	I/O	The **General Purpose Input/Output (GPIO)** pins provide general purpose inputs and/or outputs.
UART_TX	F18	O	The **UART Output** pin provides the UART output from the serial debugger.

Table 3.4 Configuration and General-Purpose Signal Description (Cont.)

Signal Name	BGA Position	I/O	Description
UART_RX	E18	I	The **UART Input** pin provides the UART input to the serial debugger.
LEDSYNCOUT	AE17	O	The **LED Synchronization Output** pin provides a 0.5 Hz output clock that can drive the LEDSYNCIN pin on other expanders. This signal is multiplexed to the SIO interface. Refer to Section 3.7, "Multiplexed SIO Interface," for the SIO interface signal definitions.
LEDSYNCIN	AF15	I	The **LED Synchronization Input** pin provides an asynchronous input to the blink pattern generators. All blink pattern generators restart their blink pattern on the rising edge of this signal. This signal is multiplexed to the SIO interface. Refer to Section 3.7, "Multiplexed SIO Interface," for the SIO interface signal definitions.

3.7 Multiplexed SIO Interface

The SIO interface is multiplexed across the LED_STATUS[11:6]/, LEDSYNCIN, and LEDSYNCOUT signals. Table 3.5 provides the multiplexed signal definitions.

Table 3.5 Multiplexed SIO Signals

LSISASx12 Signal	SIO Signal	BGA	I/O	Description
LED_STATUS[6]/	SioEnd	AE18	O	This output signal indicates that this module is currently driving the last bit of serial data on the SioDout line.
LED_STATUS[7]/	SioStart	AB15	I	This signal indicates that serial data may be driven on the SioDout line during the next clock cycle, and that data can be received on the SioDin line. In Single originator mode, which is set by the SioMode bit in the SIO Control register, the SioEnd signal provides the start signal, and the LSISASx12 does not require an SioStart input.
LED_STATUS[8]/	SioDout	AF18	O	This signal provides the serial data output line.
LED_STATUS[9]/	SioDin	AF17	I	This signal provides the serial data input line.

Table 3.5 Multiplexed SIO Signals (Cont.)

LSISASx12 Signal	SIO Signal	BGA	I/O	Description
LED_STATUS[10]/	SioClkin	AF16	I	An originator with ClkEnable set to 1 in its SIO_CFG register drives SioClkout as an output. The ClkDivide bits in the SIO_CFG register control the frequency of this clock. Originators with ClkEnable cleared to 0 receive this clock signal on SioClkin.
LED_STATUS[11]/	SioClkout	AF9	O	
LEDSYNCOUT	BlinkClkin	AE17	I	An originator with ClkEnable set to 1 in its SIO_CFG register drives BlinkClkout as an output with the frequency fixed at 0.5 Hz. Originators with ClkEnable cleared to 0 receive this clock signal on BlinkClkin.
LEDSYNCIN	BlinkClkout	AF15	O	

3.8 JTAG Pins

Table 3.6 describes the JTAG signal pins.

Table 3.6 JTAG and Test Pins Signal Description

Signal Name	BGA Position	I/O	Description
TCK	E20	I	JTAG Debug **clock**.
TRST/	E19	I	JTAG Debug **reset** (active low).
TDI	D19	I	JTAG Debug **test data in**.
TDO	D18	O	JTAG Debug **test data out**.
TMS	E17	I	JTAG Debug **test mode select**.

3.9 Factory Test Pins

Table 3.7 describes the factory test signal pins. These signals are reserved for LSI Logic test purposes.

Table 3.7 Factory Test Pins

Signal Pin	Ball	Type	Description
IDDT	A2	I	Active HIGH **IDDQ Test Mode Enable** is for LSI Logic production test only. Tie this pin to 0 for normal operation.
TN/	E6	I	Used for LSI Logic production test only. Tie this signal to 1 for normal operation.
SCANMODE	F19	I	Used for LSI Logic production test only.

Table 3.7 Factory Test Pins (Cont.)

Signal Pin	Ball	Type	Description
SCANEN	D21	I	Used for LSI Logic production test only.
SCANCLK1	D4	I	Used for LSI Logic production test only.
SCANCLK2	F6	I	Used for LSI Logic production test only.
SCANCLK3	C3	I	Used for LSI Logic production test only.
PROCMON	D20	O	Used for LSI Logic production test only.
TDiode_P	AE2	I	The **Thermal Diode Anode** pin provide Anode connection of the thermal diode.
TDiode_N	Y7	O	The **Thermal Diode Cathode** pin provides Cathode connection of the thermal diode.
MODE[3:0]	B5, C5, A4, B4	I	The **Mode Select** pins are reserved for LSI Logic test purposes. Externally pull these signals LOW for normal operation.

3.10 Power and Ground Pins

Table 3.8 describes the power signals.

Table 3.8 Power and Ground Pins

Signal Pin	Ball	Type	Description
VDDIO33	C11, C12, AD12, AD13, AC17, AD15, AD16, AB22, AD20, AD21, C19, C20, D10, D22, C6, C7, E5, AC5, AD7, AD8, AC11	Power	**VDDIO33 I/O power.** These pins provide 3.3 V I/O power.
VDD2	M13, M15, U4, AA3, AB5, Y3, K23, L24, M24, N12, N14, P13, P15, R12, R14, R3, T3	Power	**VDD2 Core power.** These pins provide 1.2 V core power.

Table 3.8 Power and Ground Pins (Cont.)

Signal Pin	Ball	Type	Description
VSS2	A25, B1, B6, B7, B11, B12, B14, B15, B19, B20, D5, D11, D17, E21, E23, F5, F25, G2, G8, G25, H2, H20, K4, L23, L25, M2, M12, M14, M25, N2, N13, N15, P12, AC10, P14, P25, R2, R13, R15, R25, T2, T4, U23, W7, W25, Y2, Y19, Y25, AA2, AA22, AB4, AB6, AC16, AC22, AE7, AE8, AE12, AE13, AE15, AE16, AE20, AE21, AE26, AF2	Ground	**VSS2** Ground. These pins provide Core and I/O ground.
PLL_VDD	B10	Power	**PLL_VDD Power**. These pins provide 1.2 V PLL power.
PLL_VSS	C10	Ground	**PLL_VSS Ground**. These pins provide PLL ground.
RXB_VDD[11:0]	Y4, V4, R4, K3, H5, H7, G22, J22, K25, T26, W23, AA23	Power	**RXB_VDD** supplies power to the input stage of the receiver analog section and the Receive Buffer of each GigaBlaze core.
RXB_VSS[11:0]	W6, U5, P4, J2, G5, G7, G21, J21, K24, R26, V23, Y23	Ground	**RXB_VSS** supplies ground to the input stage of the receiver analog section and the Receive Buffer of each GigaBlaze core.
RX_VDD[11:0]	AD1, W1, R1, N5, E1, D3, D24, F26, K26, P22, Y26, AD26	Power	**RX_VDD** supplies power to the receiver analog section of each GigaBlaze core.
RX_VSS[11:0]	AA4, W4, R5, K2, J6, G6, F23, H23, M22, U26, V22, W21	Ground	**RX_VSS** supplies ground to the receiver analog section of each GigaBlaze core.
TXB_VDD[11:0]	Y5, V5, U2, J1, J5, H6, G23, J23, M23, V26, V21, Y21	Power	**TXB_VDD** supplies power to the output stage of the transmitter analog section and to the Transmit Buffer of each GigaBlaze core.
TXB_VSS[11:0]	Y6, V6, U3, K1, H4, F4, H21, K22, N23, U25, W22, W20	Ground	**TXB_VSS** supplies ground to the output stage of the transmitter analog section and to the Transmit Buffer of each GigaBlaze core.
TX_VDD[11:0]	AD2, AB2, V2, P5, J3, E3, D26, H25, P26, R23, AA24, AE25	Power	**TX_VDD** supplies power for transmitter analog section of each GigaBlaze core.

Table 3.8 Power and Ground Pins (Cont.)

Signal Pin	Ball	Type	Description
TX_VSS[11:0]	AA05, W05, T5, M1, J4, G4, H22, L22, N22, U24, Y22, Y20	Ground	**TX_VSS** supplies ground for transmitter analog section of each GigaBlaze core.
NC	A3, A9, A20, A10, A11, A12, A13, A14, A15, A17, A18, A19, A21, A22, A23, A24, B3, B9, B13, B17, B18, B21, B22, B23, B24, B25, C4, C9, C13, C14, C15, C17, C18, C21, C22, C23, D2, D7, D8, D9, D12, D13, D16, D23, D25, E1, E2, E4, E6, E7, E8, E9, E10, E11, E12, E13, E14, E15, E22, E26, F1, F7, F8, F9, F19, F20, F21, F22, F24, G3, G20, G24, H3, H24, J26, K5, L1, L4, L5, M3, M4, M5, N3, N4, N26, P23, P24, R22, R24, T1, T22, T23, U22, V3, W24, W26, Y1, Y8, Y19, Y24, AA7, AA9, AA19, AA25, AB3, AB8, AB7, AB9, AB10, AB12, AB14, AB16, AB19, AB20, AB23, AC6, AC8, AC14, AC15, AC18, AC19, AC20, AC21, AC26, AD3, AD11, AD17, AD25, AE10, AE14, AF8, AF10, AF11, AF12, AF13, AF14		No connect. These signals do not connect inside of the LSISASx12 package.

Chapter 4
Register Descriptions

This chapter provides the address map and register descriptions for the LSISASx12 expander chip. This chapter contains the following sections:

Do not access reserved address spaces and reserved bits.

Section 4.1, “Primary Internal Registers Address Map” through Section 4.6, “Expander Connection Manager Registers,” on page 4-124 document the primary internal registers of the LSISASx12. Section 4.7, “EMB Slave Registers,” on page 4-133 documents the EMB registers. The EMB register space uses a separate address space than the primary internal registers, and can only be accessed through the EMB I^2C port.

4.1 Primary Internal Registers Address Map

Table 4.1 provides a top-level address map for the LSISASx12. Each address space is further explained in subsequent sections.

Table 4.1 Top-Level Address Map

Region	Size	Address Range	Page
Configuration Manager Registers	64 Kbytes	0x00000–0x0FFFF	4-3
SMP Target Registers	32 Kbytes	0x10000–0x17FFF	4-72
SPhynx Registers[1]	32 Kbytes	0x18000–0x1FFFF	4-75
Reserved	128 Kbytes	0x20000–0x3FFFF	–
SPhynx[00] Link Registers SPhynx[00] Phy Registers	1 Kbyte	0x40000–0x402FF 0x40300–0x403FF	4-75 4-93
SPhynx[01] Link Registers SPhynx[01] Phy Registers	1 Kbyte	0x40400–0x406FF 0x40700–0x407FF	4-75 4-93
SPhynx[02] Link Registers SPhynx[02] Phy Registers	1 Kbyte	0x40800–0x40AFF 0x40B00–0x40BFF	4-75 4-93
SPhynx[03] Link Registers SPhynx[03] Phy Registers	1 Kbyte	0x40C00–0x40EFF 0x40F00–0x40FFF	4-75 4-93
SPhynx[04] Link Registers SPhynx[04] Phy Registers	1 Kbyte	0x41000–0x412FF 0x41300–0x413FF	4-75 4-93
SPhynx[05] Link Registers SPhynx[05] Phy Registers	1 Kbyte	0x41400–0x416FF 0x41700–0x417FF	4-75 4-93
SPhynx[06] Link Registers SPhynx[06] Phy Registers	1 Kbyte	0x41800–0x41AFF 0x41B00–0x41BFF	4-75 4-93
SPhynx[07] Link Registers SPhynx[07] Phy Registers	1 Kbyte	0x41C00–0x41DFF 0x41F00–0x41FFF	4-75 4-93
SPhynx[08] Link Registers SPhynx[08] Phy Registers	1 Kbyte	0x42000–0x422FF 0x42300–0x423FF	4-75 4-93
SPhynx[09] Link Registers SPhynx[09] Phy Registers	1 Kbyte	0x42400–0x426FF 0x42700–0x427FF	4-75 4-93
SPhynx[10] Link Registers SPhynx[10] Phy Registers	1 Kbyte	0x42800–0x42AFF 0x42B00–0x42BFF	4-75 4-93
SPhynx[11] Link Registers SPhynx[11] Phy Registers	1 Kbyte	0x42C00–0x42EFF 0x42F00–0x42FFF	4-75 4-93
STP Target Registers	1 Kbyte	0x43000–0x433FF	4-99

Table 4.1 Top-Level Address Map (Cont.)

Region	Size	Address Range	Page
Reserved	115 Kbytes	0x43400–0x5FFFF	–
Phy Configuration Registers	64 Kbytes	0x60000–0x6FFFF	4-124
Remote Bank Configuration Registers	64 Kbytes	0x70000–0x7FFFF	4-129

1. All SPhynx modules accept writes from this region.

4.2 Configuration Manager Registers

Table 4.2 provides the register map for the Configuration Manager Config registers. Detailed register descriptions follow the address map.

Table 4.2 Configuration Manager Config Register Map

Offset	Register Name
0x00000	LSISASx12 Expander SAS Address High
0x00004	LSISASx12 Expander SAS Address Low
0x00008	Vendor Identifier High
0x0000C	Vendor Identifier Low
0x00010	Product Identifier 3
0x00014	Product Identifier 2
0x00018	Product Identifier 1
0x0001C	Product Identifier 0
0x00020	Product Revision
0x00024	Component Vendor ID High
0x00028	Component Vendor ID Low
0x0002C	Component ID and Revision
0x00030	Vendor Specific Dword 1
0x00034	Vendor Specific Dword 0
0x00038	Report General 1
0x0003C	Report General 2
0x00040	Spin-up Control
0x00044	Phy Configuration
0x00048	Reserved
0x0004C	Phy Transmit Polarity

Table 4.2 Configuration Manager Config Register Map (Cont.)

Offset	Register Name
0x00050	Phy Receiver Polarity
0x00054–0x0008F	Reserved
0x00090	Boot Control and Status
0x00094–0x03FFF	Reserved
0x4000	API2C Global Control
0x4008	API2C Interrupt Status
0x4010	API2C Interrupt Enable
0x4018	API2C Wait Timer Control
0x4020	API2C $t_{TIMEOUT}$ Control
0x4028	API2C t_{LOW} Control
0x4030	Reserved
0x4038	API2C Timer Clock Divider Control
0x4040	API2C Monitor
0x4048	API2C Soft Reset
0x4050	API2C Master Command
0x4058	API2C Receive Transfer Length
0x4060	API2C Transmit Transfer Length
0x4068	API2C Address Register 1
0x4070	API2C Address Register 2
0x4078	API2C Data
0x4080	API2C Transmit FIFO Status
0x4088	API2C Receive FIFO Status
0x4090	API2C Master Interrupt Enable
0x4098	API2C Master Interrupt Status
0x40A0	API2C Transmit Bytes Transferred
0x40A8	API2C Receive Bytes Transferred
0x040B0–0x40FF	Reserved
0x4100	API2C SCL High Period
0x4108	API2C SCL Low Period
0x4110	API2C Spike Filter Control
0x4118	API2C SDA Setup Time
0x4120	API2C SDA Hold Time

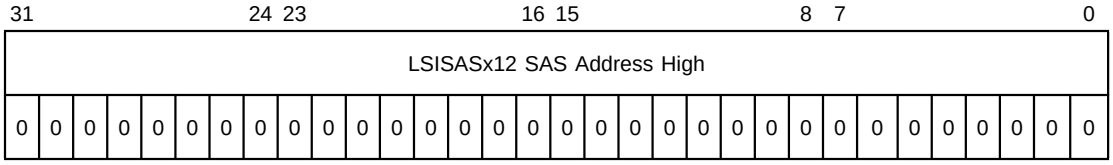
Table 4.2 Configuration Manager Config Register Map (Cont.)

Offset	Register Name
0x04130–0x07FFF	Reserved
0x08000	LED Group Control
0x08004	Group 0 GPIO Control
0x08008	Group 1 GPIO Control
0x0800C	Group 2 GPIO Control
0x08010	Group 3 GPIO Control
0x08014	Group 4 GPIO Control
0x08018	Group 0 GPIO Value
0x0801C	Group 1 GPIO Value
0x08020	Group 2 GPIO Value
0x08024	Group 3 GPIO Value
0x08028	Group 4 GPIO Value
0x0802C	Reserved
0x08030	LED Blinker Definition 1
0x08034	LED Blinker Definition 2
0x08038	LED Blinker Definition 3
0x0803C	Reserved
0x08040	Group 0 Override
0x08044	Reserved
0x08048	Group 1 Override
0x0804C	Reserved
0x08050	Group 2 Override
0x08054	Reserved
0x08058	Group 3 Override
0x0805C	Reserved
0x08060	Group 4 Override
0x08064–0x0BFFF	Reserved
0x0C000	SIO Configuration
0x0C004	SIO Receive 0
0x0C008	SIO Receive 1
0x0C00C	SIO General Purpose Receive 0
0x0C010	SIO General Purpose Receive 1

Table 4.2 Configuration Manager Config Register Map (Cont.)

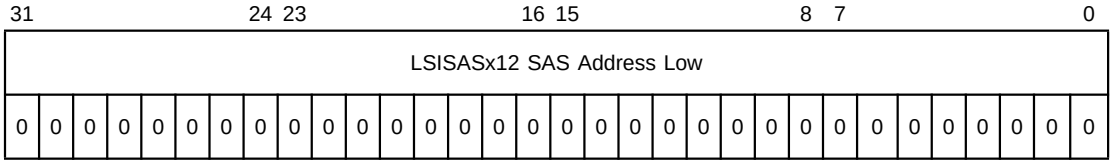
Offset	Register Name
0x0C014	SIO Output Data Control 0
0x0C018	SIO Output Data Control 1
0x0C01C	SIO Output Data Control 2
0x0C020	SIO General Purpose Output Data 0
0x0C024	SIO General Purpose Output Data 1
0x0C028–0x0C03F	Reserved
0x0C040	SIO Pattern Definition 0
0x0C044	SIO Pattern Definition 1
0x0C048	SIO Pattern Definition 2
0x0C04C	SIO Pattern Definition 3
0x0C050	Activity Stretch Control
0x0C054	SIO Adapter Control

Register: 0x0000
LSISASx12 Expander SAS Address High
Read/Write



WWN High [31:0]
This field provides World Wide Name (WWN) bits [63:32]. The boot loader writes this value during initialization. The LSISASx12 copies this value to the SMP target, STP target, and each SPhynx module.

Register: 0x0004
LSISASx12 Expander SAS Address Low
Read/Write



WWN Low **[31:0]**

This field provides WWN bits [31:0]. The boot loader writes this value during initialization. The LSISASx12 copies this value to the SMP target, STP target, and each SPhynx module. Set the least significant nibble of this register to 0xF.

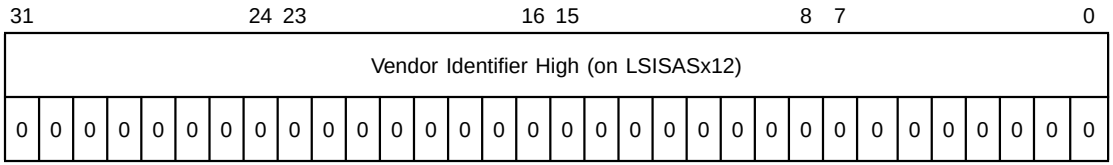
Any SPhynx modules attached to SATA devices and the STP target use {WWN High, WWN Low[31:4], PhyNum} as their source SAS address.

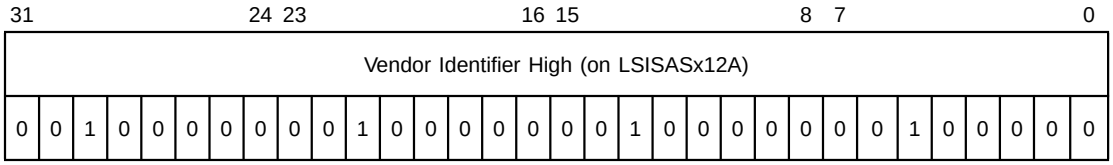
The SMP target and any Sphynx modules attached to SAS devices use {WWN High, WWN Low} as their SAS address.

Register: 0x0008

Vendor Identifier High

Read/Write

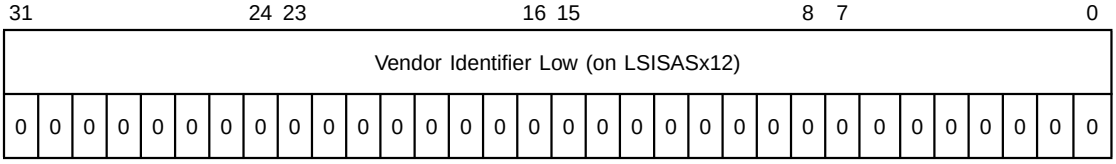


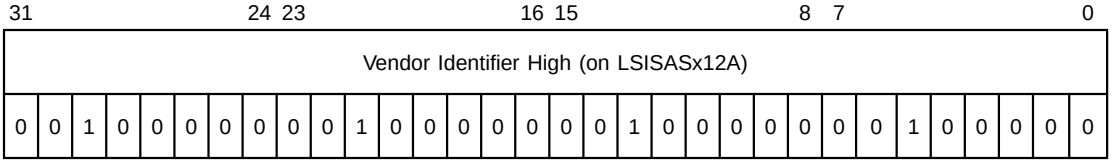


Vendor ID High **[31:0]**

This field provides Vendor ID bits [63:32]. Bits [31:24] of this register correspond to the first byte of the vendor ID data which is typically an ASCII text string.The boot loader writes this field during initialization.

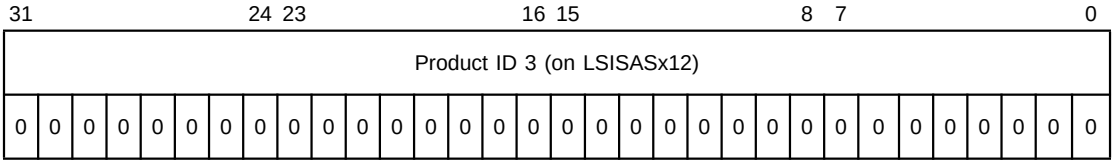
Register: 0x000C
Vendor Identifier Low
Read/Write

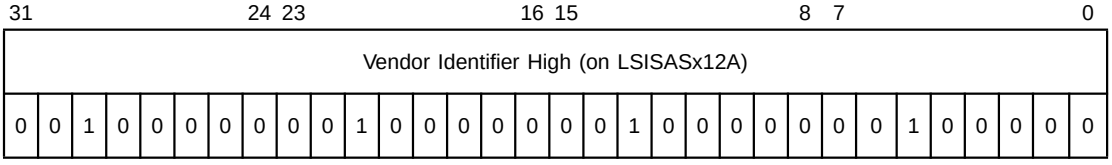




Vendor ID Low **[31:0]**
This field provides Vendor ID bits [31:0]. Bits [7:0] of this register correspond to the last byte of the vendor ID data which is typically an ASCII text string. The boot loader writes this field during initialization. Set the lower nibble of this register to 0xF.

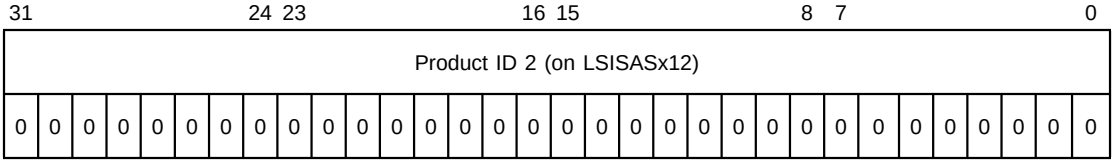
Register: 0x0010
Product Identifier 3
Read/Write

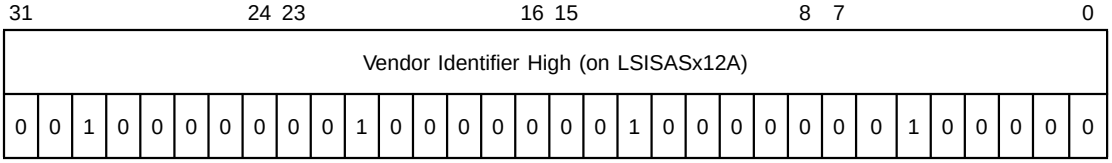




Product ID 3 **[31:0]**
This field provides Product Identifier bits [127:96]. The boot loader writes this field during initialization.

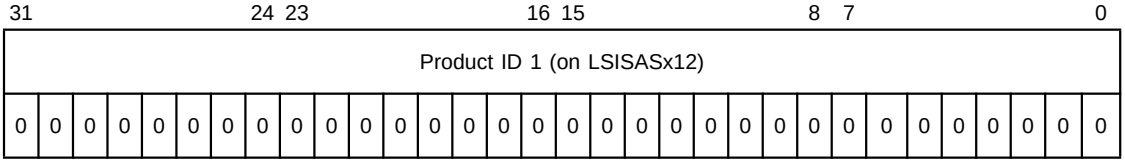
Register: 0x0014
Product Identifier 2
Read/Write

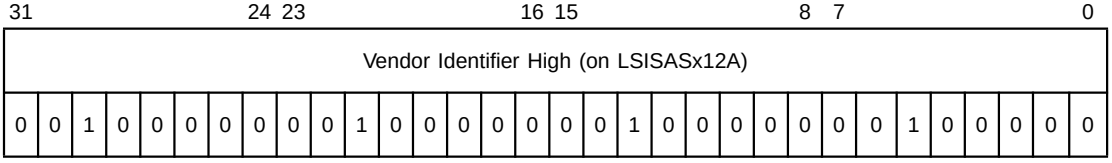




Product ID 2 **[31:0]**
This field contains Product Identifier bits [95:64]. The boot loader writes this field during initialization.

Register: 0x0018
Product Identifier 1
Read/Write

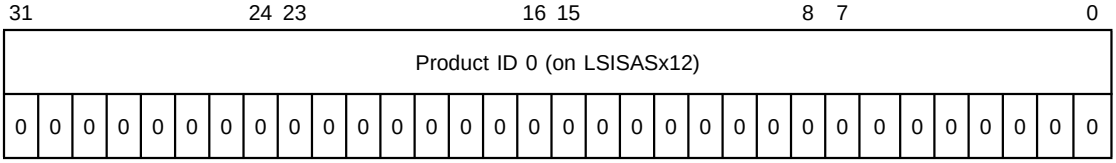


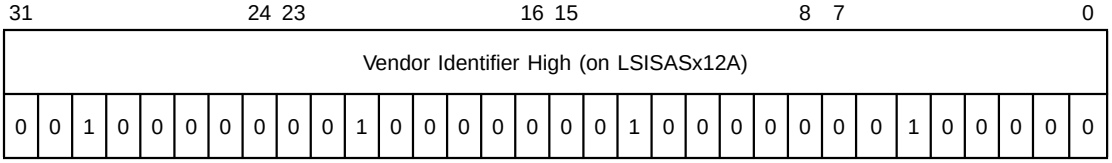


Product ID 1 **[31:0]**
This field contains Product Identifier [63:32]. The boot loader writes this field during initialization.

Register: 0x001C

Product Identifier 0
Read/Write

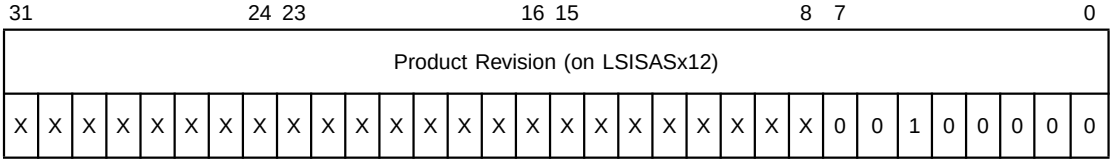


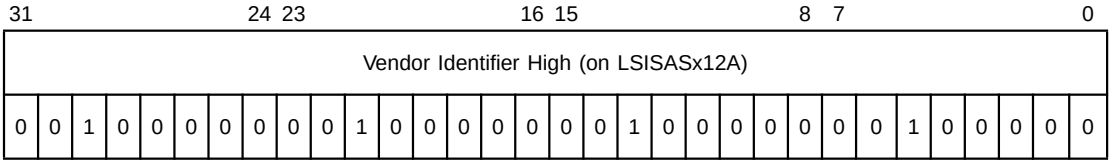


Product ID 0 **[31:0]**
This field contains Product Identifier bits [31:0]. The boot loader writes this field during initialization.

Register: 0x0020

Product Revision
Read/Write

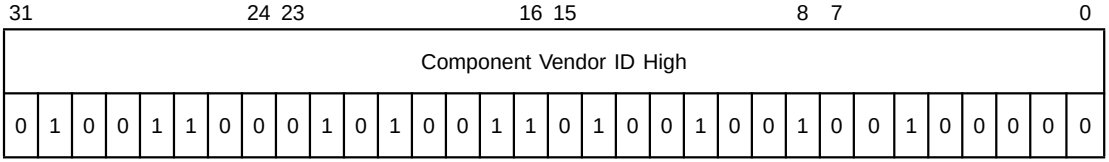




Product Revision **[31:0]**
Load this field with the ASCII bytes representing the revision level of the subsystem (board or enclosure) containing this expander device. The upper three bytes are loaded by the Boot Sequencer as part of the fixed record. On the LSISASx12 the lowest bytes defaults to 0x20 to

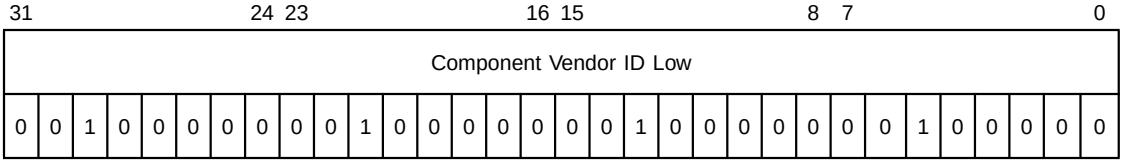
indicate an ASCII space, but may also be loaded by the Boot Sequencer with an optional write. Left-align data within this field. On the LSISASx12A all bytes default to 0x20202020.

Register: 0x0024
Component Vendor ID High
Read Only



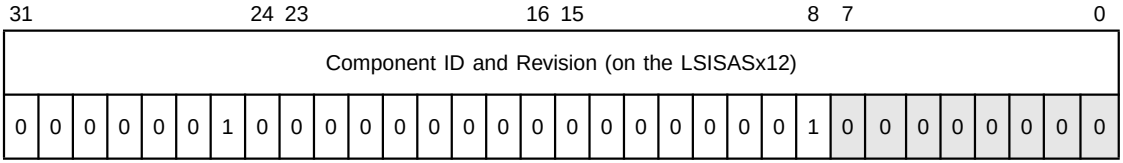
Component Vendor ID High **[31:0]**
This read-only field contains the Component Vendor ID High data. This value is hardwired to indicate the ASCII for “LSI Logic”.

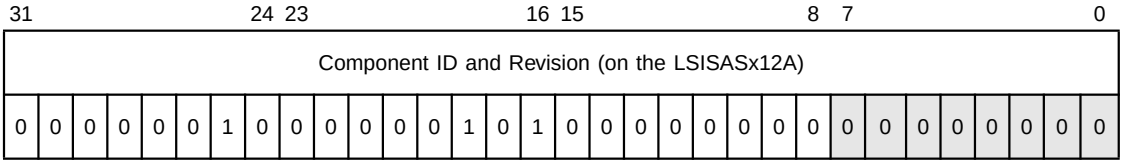
Register: 0x0028
Component Vendor ID Low
Read/Write



Component Vendor ID Low **[31:0]**
This read-only field contains the Component Vendor ID Low data. This value is hardwired to 0x20202020, which is 4 ASCII spaces.

Register: 0x002C
Component ID and Revision
Read/Write



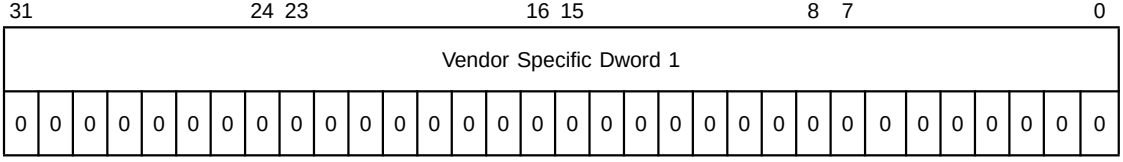


Component ID **[31:16]**
This field contains Component ID. This field is hardwired to 0x0200 on the LSISASx12 and hardwired to 0x0205 on the LSISASx12A.

Revision ID **[15:8]**
This field contains the revision ID. This field is hardwired to 0x01 on the LSISASx12 and to 0x00 on the LSISASx12A.

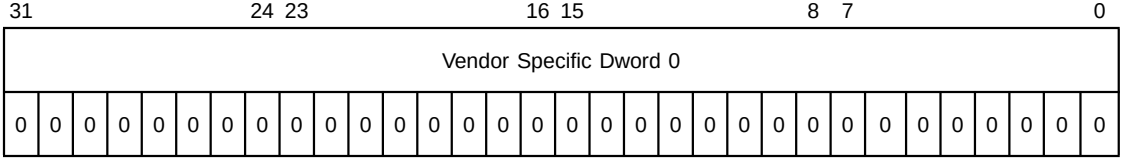
Reserved **[7:0]**

Register: 0x0030
Vendor Specific Dword 1
Read/Write



Vendor Specific Dword 1 **[31:0]**
This field contains Vendor Specific Dword 1. The boot loader optionally writes this field during initialization.

Register: 0x0034
Vendor Specific Dword 0
Read/Write

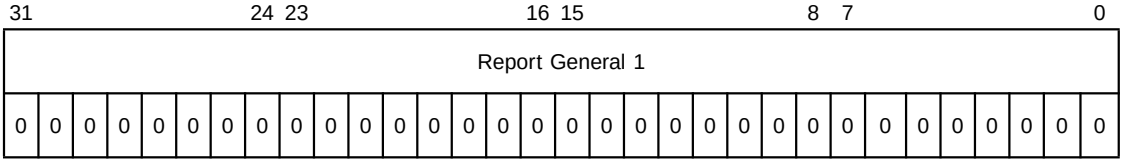


Vendor Specific Dword 0 **[31:0]**

This field contains Vendor Specific Dword 0. The boot loader optionally writes this field during initialization.

Register: 0x0038

Report General 1
Read/Write

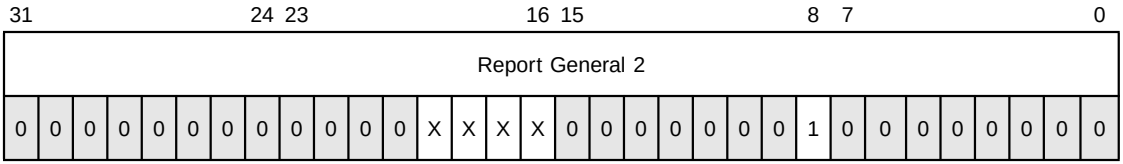


Expander Change Count **[31:16]**
This read-only field provides the count of the BROADCAST(CHANGE) primitive sequences that this device initiated. The value in this field equals the sum of the change counts in each of the links.

Expander Route Indexes **[15:0]**
The field provides the number of route indexes for this expander.

Register: 0x003C

Report General 2
Read/Write



Reserved **[31:20]**

NPhys **[19:16]**
This read-only field provides the number of phys for this expander. When the STP target is enabled, this field is 13 (0b1101). When the STP target is disabled, this field is 12 (0b1100)

Reserved **[15:9]**

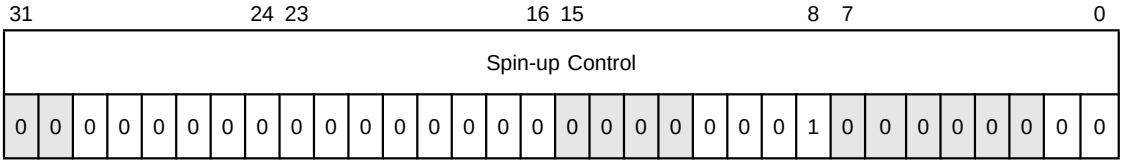
Configuration Table Support **8**

Setting this bit indicates that the expander supports a configurable routing table. Clearing this bit indicates that the expander does not support a configurable routing table.

Reserved **[7:0]**

Register: 0x0040

Spin-up Control
Read/Write



Reserved **[31:30]**

Spin-up Delay **[29:16]**
This field indicates the spin-up delay in milliseconds.

Reserved **[15:12]**

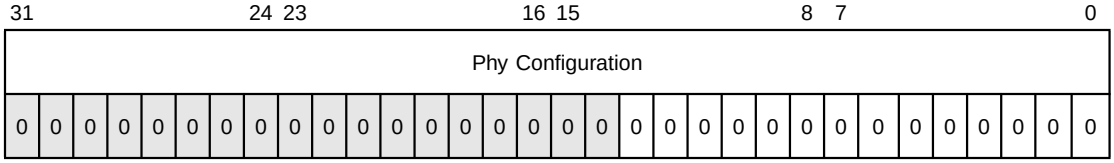
Spin Number **[12:8]**
This field specifies the maximum number of phys that the LSISASx12 is permitted to concurrently spin up.

Reserved **[7:2]**

Spin Mode **[1:0]**
This field specifies the spin-up mode for phys that are attached to SATA devices. The encoding of this field is:

Bit Value	Definition
0b00	SATA spin mode immediate
0b01	Reserved.
0b10	SATA spin mode host-notify
0b11	SATA spin mode self-timed

Register: 0x0044
Phy Configuration
Read/Write



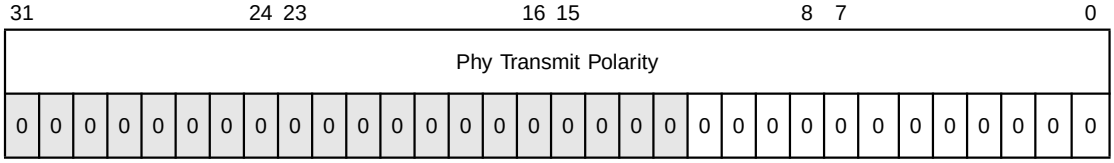
Reserved **[31:14]**

SMP Target Enable **13**
Setting this bit enables the SMP Target within the expander. The boot loader writes this field during initialization. The value of this field is passed to the SMP target after the boot checksum is validated.

STP Target Enable **12**
Setting this bit enables the STP Target and SEMB functions within the expander. The boot loader writes this field during initialization. The value of this field is passed to the STP target after the boot checksum is validated.

Phy Enables **[11:0]**
This field contains the enable bits for each phy. The boot loader writes this field during initialization. The value of this field is passed to each phy after the boot checksum is validated. Bit [0] corresponds to Phy [0], while bit [11] corresponds to Phy [11].

Register: 0x004C
Phy Transmit Polarity
Read/Write



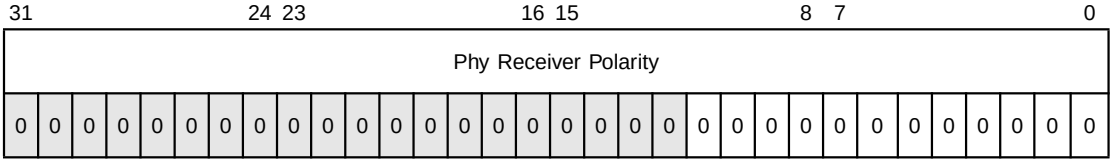
Reserved **[31:12]**

Phy Transmit Polarity **[11:0]**

This field contains the transmit polarity bits for each phy. Setting a bit inverts transmitter polarity for the corresponding phy from the default polarity of the phy. This is done to improve routing of board designs. Bit [0] corresponds to Phy[0], while bit [11] corresponds to Phy [11].

Register: 0x0050

Phy Receiver Polarity
Read/Write

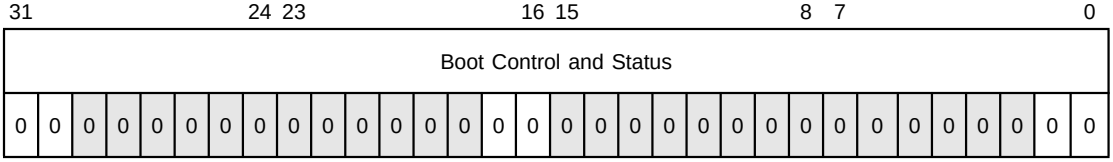


Reserved **[31:12]**

Phy Receiver Polarity **[11:0]**
This field contains receiver polarity bit for each phy. Setting a bit inverts receiver polarity for the corresponding phy from the default polarity of the phy. This is done to improve routing of board designs. Bit [0] corresponds to Phy [0], while bit [11] corresponds to Phy [11].

Register: 0x0090

Boot Control and Status
Read/Write



Boot Sequencer Active **31**
This read-only bit provides boot sequencer activity status. This bit asserts when the boot sequencer is active.

Boot OK **30**
This read-only bit provides the boot sequencer completion status. This bit asserts if the boot operation ends normally.

Reserved **[29:18]**

ISTWI Address **[17:16]**
This read-only field provides the value of the ISTWI address input pins. This is also referred to as the I^2C address.

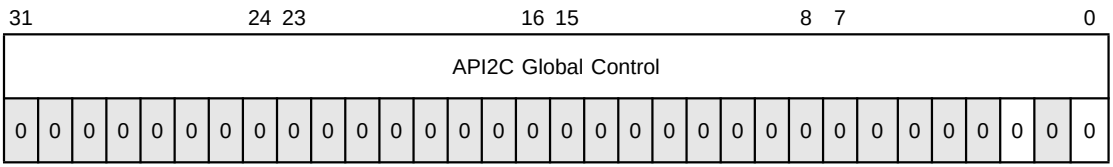
Reserved **[15:3]**

Phy Enable Bit Enable **2**
This bit enables the Phy Enable bits in the Phy Configuration register.

Boot Start **1**
Setting this write-only bit initiates a boot download operation. This bit is self-clearing.

Boot Stop **0**
Setting this write-only bit aborts the boot download operation. This bit is self-clearing.

Register: 0x4000
API2C Global Control
Read/Write



This register independently enables the master unit and globally enables and disables the IBML time-out timers.

R **Reserved** **[31:3]**
These bits are reserved.

IE **Timer Enable** **2**
This bit enables the API2Ctimers.

IE	IBML Timers
0	Disable
1	Enable

R **Reserved** **1**
This bit is reserved and is hardwired to 0x0.

ME **Master Enable** **0**

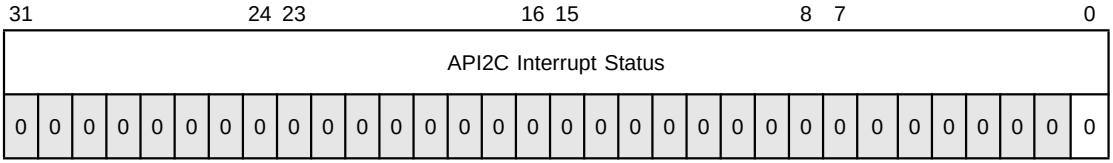
This bit enables the API2C master state machine.

ME	API2C Master State Machine
0	Disable
1	Enable

Register: 0x4008

API2C Interrupt Status

Read/Write



This register reports the status of the master state machine interrupt sources.

R **Reserved** **[31:1]**

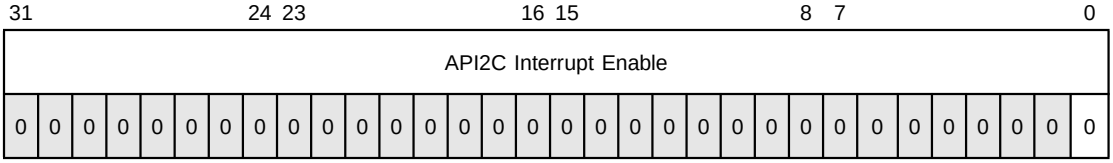
These bits are reserved.

MI **Master Interrupt** **0**

This bit reflects the logical OR of the master interrupt sources in the API2C Master Interrupt Status register, and determines the master contribution to the external IRQ signal.

MI	Master Interrupt
0	Either disabled or none of the master interrupt sources are set.
1	Enabled and one or more of the master interrupt sources are set.

Register: 0x4010
API2C Interrupt Enable
Read/Write



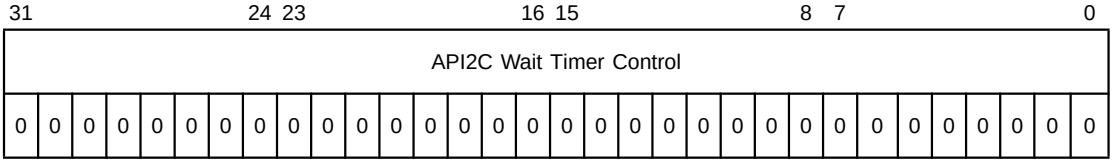
This register enables the reporting of interrupt status for the master interrupt sources. The status is reported through the corresponding API2C Interrupt Status register bit and through the external interrupt signal.

R **Reserved** **[31:1]**
These bits are reserved.

MIE **Master Interrupt Enable** **0**
This bit enables the master interrupt status.

MIE	Function
0	The reporting of the master interrupt is disabled.
1	The master interrupt is enabled. A master interrupt sets the master interrupt status bit and asserts the master contribution to the external IRQ signal if one or more of the master interrupt sources are true.

Register: 0x4018
API2C Wait Timer Control
Read/Write



This register controls the wait timer. The wait timer times API2C transactions and causes the appropriate state machine to take action when a time-out occurs.

R **Reserved** **[31:16]**
These bits are reserved.

TE **Timer Enable** **15**

This bit enables the wait timer.

TE	Function
0	Disables the timer and forces time-out flag to 0.
1	Enables the timer.

When the timer is enabled and the master state machine-extends the SCL LOW time by asserting an internal hold signal, the Timer Load Value loads into the timer. The timer then counts down at the divided clock rate.

When the timer reaches 0, the time-out flag is set and sent to the master state machine. The flag clears when the master hold signal switches to false or when the timer is disabled.

TLV **Timer Load Value** **[14:0]**

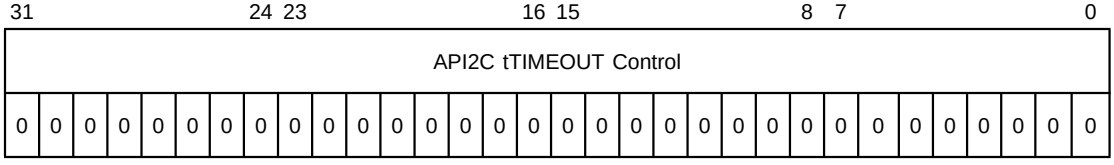
This 15-bit value loads into the timer if the timer is enabled, and if the master state machine extends the SCL LOW time. The timer counts down to 0 at the divided clock rate. If the timer reaches 0, the time-out flag is set. Following a time-out, the timer reloads the Timer Load Value when the timer is enabled and the master causes the SCL LOW time to extend.

The software programs this value for a certain time-out period, depending on the APB clock frequency according to:

```
Load value = Desired Time-Out/Divided Clock Period
```

Register: 0x4020

API2C $t_{TIMEOUT}$ Control
Read/Write



This register controls the API2C $t_{TIMEOUT}$ timer. This timer measures the time that the SCL signal is detected LOW.

R **Reserved** **[31:16]**
These bits are reserved.

TE **Timer Enable** **15**
This bit enables the API2C $t_{TIMEOUT}$ timer.

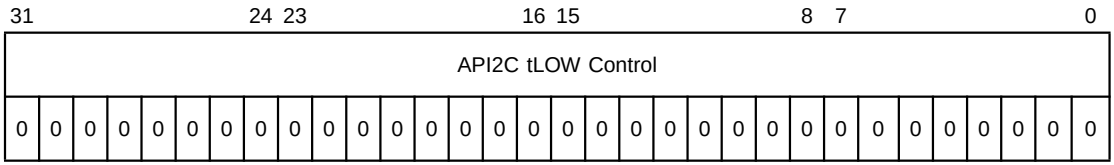
TE	Function
0	Disables timer and forces API2C $t_{TIMEOUT}$ flag to 0.
1	Enables the $t_{TIMEOUT}$ timer.

TLV **Timer Load Value** **[14:0]**
This value loads into the timer if the timer is enabled, and if the SCL signal is detected LOW following a HIGH value. The timer counts down from this value at the divided clock rate. If the timer reaches 0, the device sets the time-out flag. Following a time-out, the timer reloads the Timer Load Value when the SCL signal is detected LOW again following a HIGH state.

The software programs this value for a certain time-out period, depending on the APB clock frequency, and according to the following formula:

```
Load value = Desired Time-Out/Divided Clock Period
```

Register: 0x4028
API2C t_{LOW} Control
Read/Write



This register controls the API2C t_{LOW} timer. This timer measures the time that the master controller extends the SCL signal LOW during the transmission or reception of a data byte.

R **Reserved** **[31:16]**
These bits are reserved.

TE **Timer Enable** **15**

This bit enables the API2C t_{LOW} timer.

TE	Function
0	The timer is disabled and the API2C t_{LOW} flag is forced to 0.
1	The t_{LOW} timer is enabled.

TLV **API2C Timer Load Value** **[14:0]**

This value loads into the timer at the beginning of every data byte transfer to or from the API2C interface (Start or ACK of previous byte). When the timer is enabled and the master controller extends the SCL signal by asserting the SCL hold signal, the timer counts down from the timer value at the divided clock rate. If the timer reaches 0, the device sets the time-out flag. Following a time-out, the timer reloads the Timer Load Value at the beginning of the next data byte transfer.

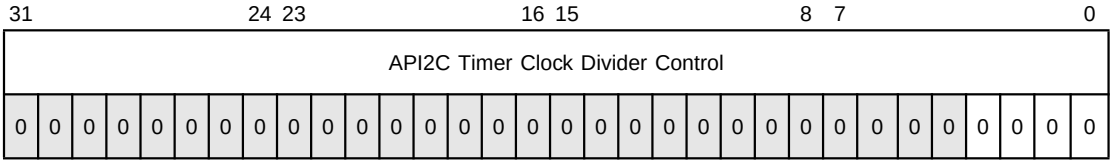
The software programs this value for a certain time-out period, depending on the APB clock frequency, and according to the following formula:

```
Load value = Desired Time-Out/Divided Clock Period
```

Register: 0x4038

API2C Timer Clock Divider Control

Read/Write



The following register divides the APB clock frequency to use for all of the timers. The value in the register picks a power of two value from a free running 16-bit counter that serves as an enable for the timers.

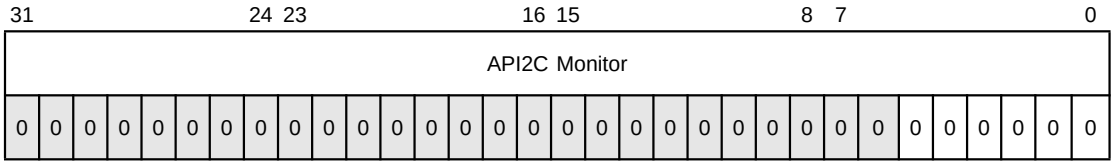
R **Reserved** **[31:4]**

These bits are reserved.

DS **Divider Select** **[3:0]**
This field selects the bit position from a 16-bit counter to use as an enable for the timers. Bit 3 is the most significant bit (MSB).

Register: 0x4040

API2C Monitor
Read/Write



This register allows the host to manually read and control the SCL and SDA API2C interface signals.

R **Reserved** **[31:6]**
These bits are reserved.

SDAE **SDA Falling Edge Detect** **5**
A 1 in this bit indicates that at least one HIGH-to-LOW transition has occurred on the SDA status line. Writing a 1 to this bit clears it.

SCLE **SCL Falling Edge Detect** **4**
A 1 in this bit indicates that at least one HIGH-to-LOW transition has occurred on the SCL status line. Writing a 1 to this bit clears it.

SDAC **SDA Control** **3**
This bit controls the SDA signal.

SDAC	Function
0	SDA is not forced LOW.
1	SDA is forced LOW on the API2C interface.

SCLC **SCL Control** **2**

This bit controls the SCL signal.

SCLC	Function
0	SCL is not forced LOW.
1	SCL is forced LOW on the API2C interface.

SDAS **SDA Status** **1**

This bit reflects the current synchronized and filtered value of the SDA signal on the API2C interface. This bit is read only.

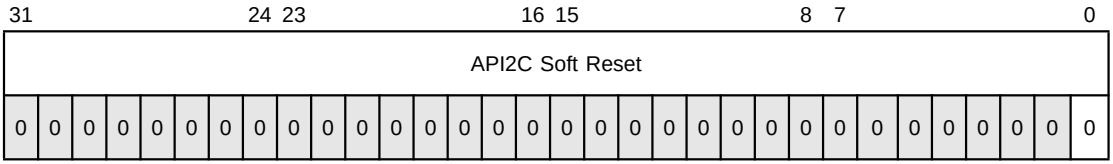
SCLS **SCL Status** **0**

This bit reflects the current synchronized and filtered value of the SCL signal on the API2C interface. This bit is read only.

Register: 0x4048

API2C Soft Reset

Read/Write



This register allows the host to independently reset the API2C interface.

R **Reserved** **[31:1]**

These bits are reserved.

I2CR **API2C Reset** **0**

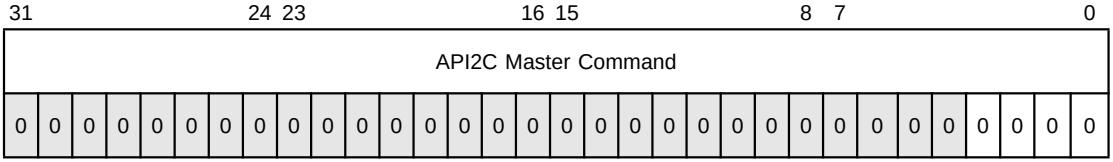
Setting this bit resets the API2C module. This bit clears automatically when the reset completes. The reset:

- Forces all master state machines to their IDLE states
- Insures that the API2C core is not driving the API2C interface
- Clears the Master Enable bit in the API2C Global Control register
- Clears master status bits

- Flushes (reset pointers) the Transmit and Receive FIFOs

Register: 0x4050

API2C Master Command
Read/Write



This register issues commands to the master state machine.

R **Reserved** **[31:4]**
These bits are reserved.

SC **Send Command** **3**
When software sets this bit, the master state machine issues the command that the command type field specifies. After the bit is set, it remains set until the command completes, and automatically clears when the command completes.

After a command is in progress, the API2C core ignores any write attempts to the command bit or to the command type field. If the software attempts to set this bit when the command type field contains a reserved command type, this bit does not set.

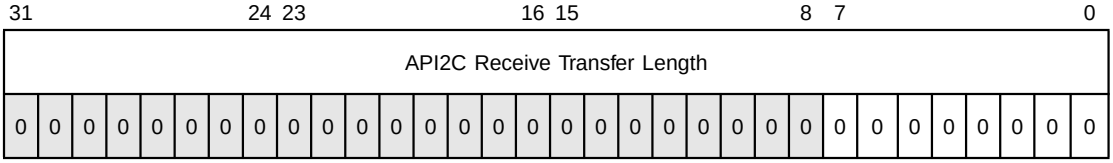
CT **Command Type** **[2:0]**
There are five commands the master state machine can execute.

CT	Function
000	Manual Mode Transfer
001	Automatic Mode Transfer
010	Sequence Mode Transfer
011	Issue Master Stop
100 – 110	Reserved
111	Flush Master Transmit FIFO

Register: 0x4058

API2C Receive Transfer Length

Read/Write



This register determines the number of bytes transferred during receive transfers. The API2C core ignores writes to this register during the execution of a data transfer command.

R **Reserved** **[31:8]**
These bits are reserved.

RXTL **Receive Transfer Length** **[7:0]**
This field determines the number of bytes received during API2C read transfers.

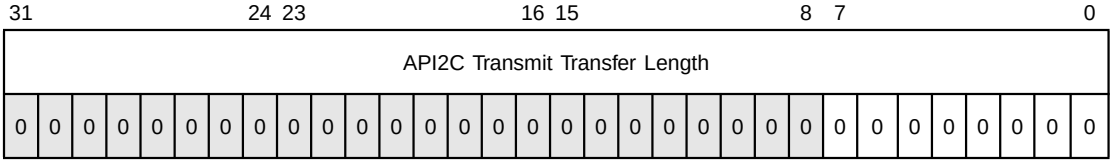
The API2C receives this number of data bytes from a slave (the Serial EEPROM) and writes them to the Receive FIFO. When the transfer length is reached, the master state machine either issues a Stop command, or holds the SCL signal LOW and awaits another command. If the Receive FIFO becomes full before the receive transfer length is reached, the master state machine inserts wait cycles by holding the SCL signal LOW.

A value of 0 for this field is illegal for the Automatic Transfer (receive) and Sequence Transfer commands. If this field is set to 0 when these commands are issued, the API2C core reports an error in the API2C Master Interrupt Status register.

For a manual command, a value of 0 causes the master state machine to send only the address phase information, and then to hold the SCL signal LOW while awaiting another command.

Register: 0x4060

API2C Transmit Transfer Length
Read/Write



This register determines the number of data bytes sent during transmit transfers. The API2C core ignores writes to this register during the execution of a data transfer command.

R **Reserved** **[31:8]**
These bits are reserved.

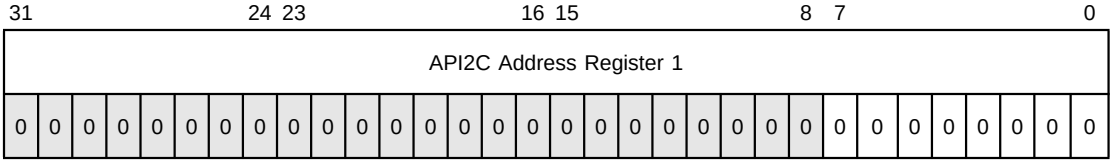
TXTL **Transmit Transfer Length** **[7:0]**
This field determines the number of bytes transmitted by the API2C during write transfers.

The API2C interface writes this number of data bytes to a slave (the Serial EEPROM) from the Transmit FIFO. When the transfer length is reached, the master state machine either issues a Stop command, or holds the SCL signal LOW and awaits another command. If the Transmit FIFO becomes empty and the transmit transfer length is not reached, the master state machine inserts wait cycles by holding the SCL signal LOW.

A value of 0 for this field is illegal for the Automatic Transfer (receive) and Sequence Transfer commands. If this field is set to 0 when these commands are issued, the API2C core reports an error in the API2C Master Interrupt Status register.

For a manual command, a value of 0 causes the master state machine to send only the address phase information, and then to hold the SCL signal LOW while awaiting another command.

Register: 0x4068
API2C Address Register 1
Read/Write



This Address Register 1 (AR1) is sent as the first byte in the address phase of an API2C transaction.

R **Reserved** **[31:8]**
These bits are reserved.

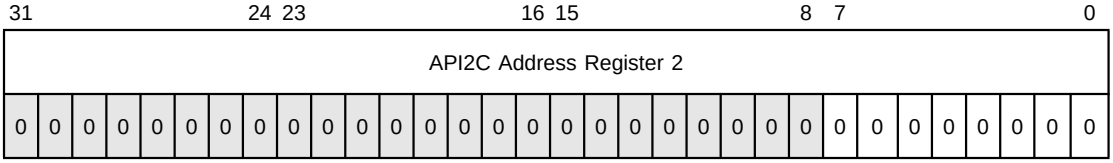
I2CA1 **API2C Address 1** **[7:1]**
These bits are sent as the first seven bits of the address phase in an API2C transaction. If bits [7:3] equal 11110, then 10-bit addressing is selected and bits [2:1] become the two most significant bits of a 10-bit address. With 10-bit addressing, an additional address byte is sent after the access direction bit and the first address byte ACK. The second byte is the lower eight bits of a 10-bit address.

AD **Access Direction** **0**
This bit is the API2C data direction bit. It is the eighth bit sent after a start or repeated start condition.

AD	Function
0	Write Access. The data phase after the address phase is taken from the Transmit FIFO and written to the addressed slave.
1	Read Access. The data after the address phase is returned by the addressed slave during the data phase. The master state machine places the data in the Receive FIFO.

This bit is ignored when a Sequence Transfer command is issued with the access direction automatically generated.

Register: 0x4070
API2C Address Register 2
Read/Write

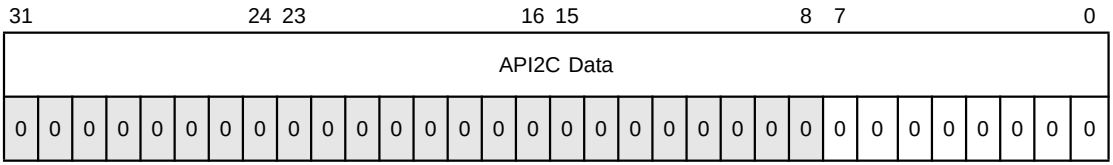


If 10-bit addressing is enabled, then this field in the Address Register 2 (AR2) is sent as the second byte in the address phase of an API2C transaction.

R **Reserved** **[31:8]**
These bits are reserved.

I2CA2 **API2C Address 2** **[7:0]**
This field contains API2C address bits [7:0] of a 10-bit API2C address. If 10-bit addressing is enabled, this field is sent as the second byte in the address phase of an API2C transaction. 10-bit addressing is enabled when bits [7:3] of the API2C Address Register 1 equal 11110.

Register: 0x4078
API2C Data
Read/Write



A write to this register places data into the Transmit FIFO. A read to this register retrieves data from the top of the Receive FIFO.

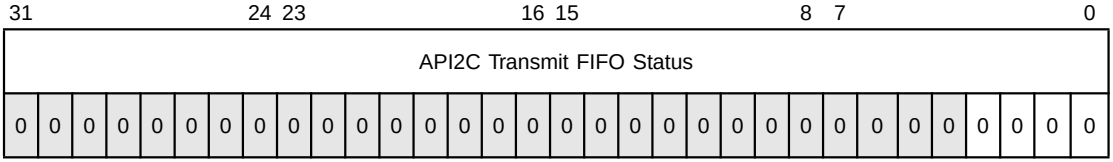
R **Reserved** **[31:8]**
These bits are reserved.

MDP **Master Data Port** **[7:0]**
A write to this location writes the data into the Transmit FIFO at the write pointer location and then increments the write pointer. The API2C core ignores the write if the Transmit FIFO is full.

A read to this location returns the data from the Receive FIFO at the read pointer location and then increments the read pointer. If the Receive FIFO is empty, the data pointed to by the read pointer is returned, but the read pointer is not incriminated.

Register: 0x4080

API2C Transmit FIFO Status
Read Only



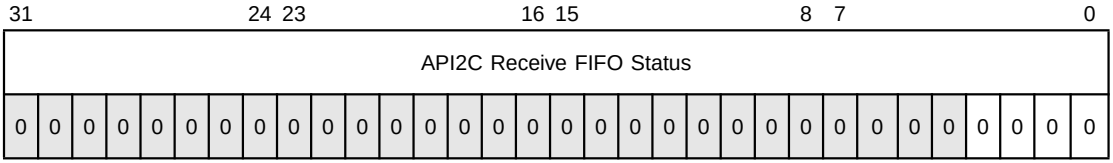
This register indicates the number of data bytes currently in the Transmit FIFO.

R **Reserved** **[31:4]**
These bits are reserved.

TFD **Transmit FIFO Depth** **[3:0]**
This field indicates the current number of data bytes in the Transmit FIFO. A value of 0 indicates that the Transmit FIFO is empty, while a value of 8 indicates that the Transmit FIFO is full. The values of 9-15 are reserved.

Register: 0x4088

API2C Receive FIFO Status
Read Only



This register indicates the number of data bytes currently in the Receive FIFO.

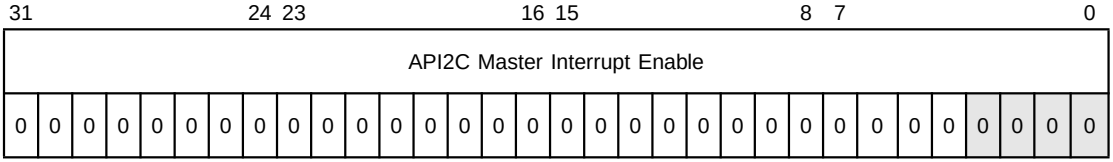
R **Reserved** **[31:4]**
These bits are reserved.

RFD **Receive FIFO Depth** **[3:0]**
This field indicates the current number of data bytes in the Receive FIFO. A value of 0 means the Receive FIFO is empty, while a value of 8 indicates the Receive FIFO is full. The values of 9-15 are reserved.

Register: 0x4090

API2C Master Interrupt Enable

Read/Write



This register enables the individual master state machine interrupts. The bits in this register correspond to the interrupt status bits in the API2C Master Interrupt Status register. When corresponding bits in both registers are set to 1, the interrupt status is reflected on the external interrupt signal and in the Master Interrupt bit in the API2C Master Interrupt Status register.

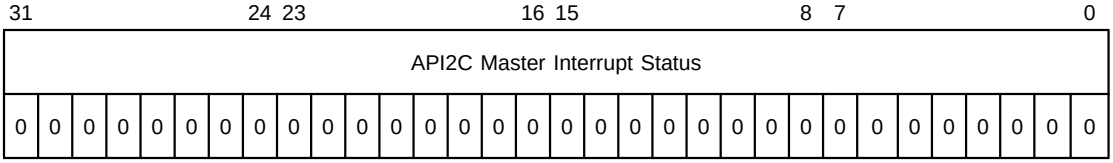
R **Reserved** **[31:16]**
These bits are reserved.

MIE **Master Interrupt Enables** **[15:4]**
These bits enable the master interrupt status for the corresponding individual interrupt sources in the API2C Master Interrupt Status register.

MIE	Function
0	Individual interrupt disabled.
1	Individual interrupt enabled.

R **Reserved** **[3:0]**
These bits are reserved and unused, but are writable.

Register: 0x4098
API2C Master Interrupt Status
RC/SC



This register records the master interrupt status of the most recently issued command. The master interrupt status bits are cleared either when read or when a new command is issued.

Status bits [15:4] are interrupt sources for the master interrupt. If interrupts are enabled, the API2C signals an interrupt when one or more of these bits are set.

R **Reserved** **[31:16]**
These bits are reserved.

TT **API2C $T_{TIMEOUT}$** **15**
This bit sets to 1 if the $T_{TIMEOUT}$ timer expires when the master controller is in control of the API2C interface.

This bit clears to 0 when this register is read.

TL **API2C T_{LOW}** **14**
This bit sets to 1 if $T_{LOW:MEXT}$ timer expires when the master controller is in control of the API2C interface.

This bit clears to 0 when this register is read.

RFL **Receive FIFO High Threshold Reached** **13**
This bit sets to 1 during a Manual, Automatic, or Sequence receive transfer when the Receive FIFO contains four bytes or more and the number of data bytes remaining in the transfer is greater than the number of free locations in the FIFO.

This bit clears to 0 when the above condition is no longer true or when the command completes abnormally (Receive FIFO is flushed).

TFL **Transmit FIFO Low Threshold Reached** **12**

During a Manual or Automatic transmit transfer, this bit sets to 1 when the Transmit FIFO contains three bytes or less, and when the number of bytes remaining in the transfer is greater than the number of data bytes in the FIFO.

This bit clears to 0 when the above condition is no longer true or when the command completes abnormally (Transmit FIFO is flushed).

SNS **Transfer Completed — Stop Not Sent** **11**

This bit sets to 1 after a manual transfer command completes successfully (all data bytes sent/received). This bit clears to 0 when it is read or when a new command is issued.

SS **Transfer Completed — Stop Sent** **10**

This bit sets to 1 when an Automatic Transfer or Sequence Transfer command completes successfully. This bit clears to 0 when it is read or when a new command is issued.

SCC **Stop Command Completed** **9**

This bit sets to 1 after the master controller completes a Stop command. A delay occurs between the assertion of this interrupt and the completion of the STOP on the API2C interface. This bit clears to 0 when it is read or when a new command is issued.

IP **Illegal Parameter** **8**

This bit sets to 1 if an Automatic Transfer or Sequence Transfer command is issued when the Master Transfer Length is set to 0. This bit clears to 0 when it is read or when a new command is issued.

TSS **Time-Out Occurred — Stop Sent** **7**

This bit is set if a transfer command times out. A time-out occurs when a wait timer expires while

- awaiting the next command
- the Transmit FIFO is empty and transmit transfer length has not been met
- the Receive FIFO is full and receive transfer length has not been met

This bit clears when it is read or when a new command is issued.

AL **Arbitration Lost** **6**
This bit sets to 1 if any data transfer command loses API2C interface arbitration. This bit clears to 0 when it is read or when a new command is issued.

ND **NAK Received During Transmit Data Phase** **5**
This bit sets to 1 if any data transfer command receives a NAK during the transmit data phase of an API2C interface transaction. This bit clears to 0 when it is read or when a new command is issued.

NA **NAK Received During Address Phase** **4**
This bit sets to 1 if any data transfer command receives a NAK during the address phase of an API2C interface transaction. This bit clears to 0 when it is read or when a new command is issued.

TS **Transfer Stopped** **3**
This bit indicates that a Stop command terminated the transfer. This bit is the logical OR of bits 15, 14, and [10:4] of this register and is read only.

STP **Sequence Transfer in Process** **2**
This bit indicates that a Sequence Transfer command data sequence (Write/Read) is in process.

This bit sets to 1 at the start of the transmit portion of a sequence command. This bit clears to 0 when the receive portion of a sequence command completes and this register is read.

TTP **Transmit Transfer in Process** **1**
This bit indicates that the master state machine is processing the transmit (write) part of a data transfer.

This bit sets to 1 at the start of the transmit transfer. The bit clears to 0 when either the transmit transfer completes and this register is read, or when a Sequence Transfer command switches to the receive part of the transfer.

Note: After an API2C timer time-out during a master read operation, this bit sets when the Stop condition appears on the interface.

RTP **Receive Transfer in Process** **0**

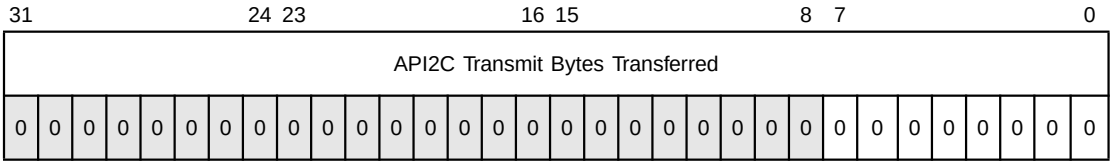
When this bit is set, it indicates that the master state machine is processing the receive (read) part of a data transfer.

This bit sets to 1 at the start of the receive transfer. The bit clears to 0 when the receive transfer completes and this register is read.

Register: 0x40A0

API2C Transmit Bytes Transferred

Read Only



This register tracks the number of successfully transmitted (acknowledged) data bytes.

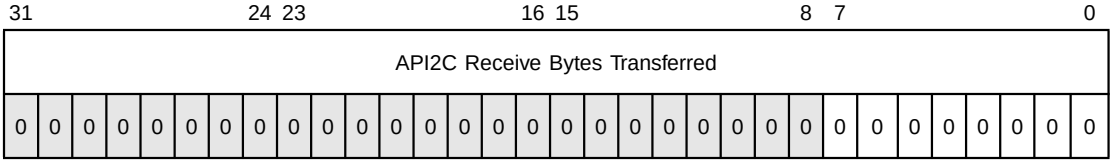
R **Reserved** **[31:8]**

These bits are reserved.

NBT **Number of Bytes Transmitted** **[7:0]**

This field counts the number of acknowledged data bytes that are transferred in the transmit direction. The register clears each time a start or repeated start occurs, and counts the data bytes that are transferred after the address phase of an API2C transaction. If a data transfer stops due to an abnormal termination, the software can use the value in this register to determine how to complete the transfer.

Register: 0x40A8
API2C Receive Bytes Transferred
Read Only

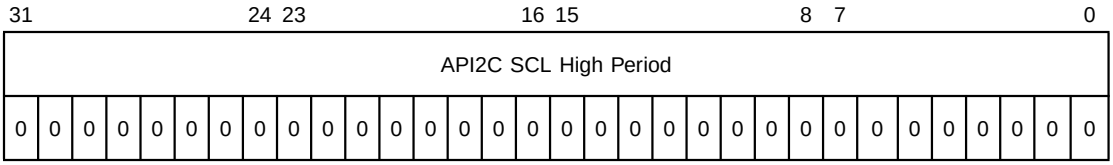


This register keeps track of the number of successfully received (acknowledged) data bytes.

R **Reserved** **[31:8]**
These bits are reserved.

NBR **Number of Bytes Received** **[7:0]**
This field counts the number of acknowledged data bytes that are transferred in the receive direction. The register clears each time a start or repeated start occurs, and counts the data bytes that are transferred after the address phase of an API2C transaction. If a data transfer stops due to an abnormal termination, the software can use the value in this register to determine how to complete the transfer.

Register: 0x4100
API2C SCL High Period
Read/Write



This register sets up the clock based on the 75 MHz PCLK frequency. It is configured automatically during system initialization. It is not necessary to program this register.

This register determines the high period for the SCL clock generated by API2C master state machine. The value specifies the number of PCLK cycles that the SCL clock is HIGH.

The programmed High Clock Count value loads into a 16-bit counter at the start of the master HIGH state. The counter decrements on each PCLK cycle. When the counter reaches 0, the SCL state machine transitions to the master LOW state.

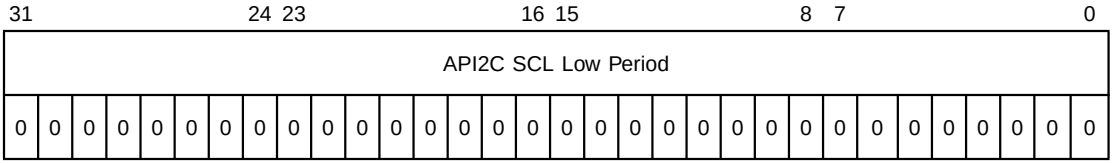
R **Reserved** **[31:16]**
These bits are reserved.

SHCC **SCL High Clock Count** **[15:0]**
This value loads in the 16-bit SCL timer.

Program an appropriate value for the desired speed mode. The available modes are the standard mode or the fast mode. Derive the appropriate value by using the following equation.

```
(Desired_SCL_High_Time/PCLK_period) – 1
```

Register: 0x4108
API2C SCL Low Period
Read/Write



This register sets up the clock based on the 75 MHz PCLK frequency. It is configured automatically during system initialization. It is not necessary to program this register.

This register determines the low period for the SCL clock that the API2C master state machine generates. The value specifies the number of PCLK cycles that the SCL clock is LOW.

The programmed Low Clock Count value loads into a 16-bit counter at the start of the master LOW state. The counter decrements on each PCLK cycle. When the counter reaches 0, the SCL state machine transitions to the master wait state and releases the SCL signal.

R **Reserved** **[31:16]**
These bits are reserved.

SLCC **SCL Low Clock Count** **[15:0]**

This is the value loads in the 16-bit SCL timer.

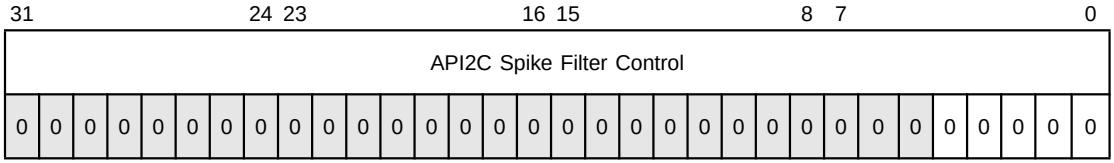
Program an appropriate value for the desired speed mode. The available modes are the standard mode or the fast mode. Derive the appropriate value by using the following equation.

```
(Desired_SCL_Low_Time/PCLK_period) – 1
```

Register: 0x4110

API2C Spike Filter Control

Read/Write



This register sets up the clock based on the 75 MHz PCLK frequency. It is configured automatically during system initialization. It is not necessary to program this register.

This register controls the spike filter for SCL and SDA. The filter stages determine the maximum size of the spike that the filter suppresses. The number of stages is expressed by the number of PCLK cycles.

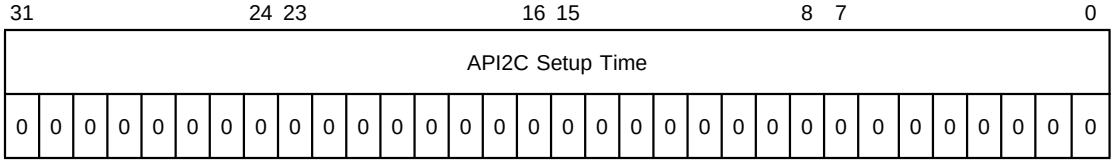
R **Reserved** **[31:5]**

These bits are reserved.

FS **Filter Stages** **[4:0]**

This five-bit value determines the number of spike filter stages. This is the size of the maximum spike suppressed. The value is in units of PCLK cycles. The maximum number of allowable stages is 16. Programming a value greater than 16 disables the filter, and no filtering occurs.

Register: 0x4118
API2C SDA Setup Time
Read/Write



This register sets up the clock based on the 75 MHz PCLK frequency. It is configured automatically during system initialization. It is not necessary to program this register.

This register controls the data setup time when the master is driving SCL and SDA. The programmed value specifies the length of time that SDA is stable prior to a rising edge on SCL. This value is in units of PCLK cycles.

Note: This register provides the $t_{SU:DAT}$ timing parameter for the API2C interface.

R **Reserved** **[31:16]**
These bits are reserved.

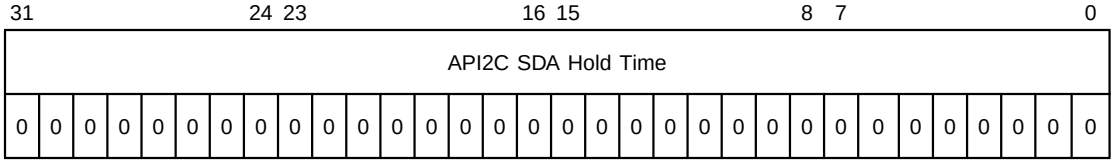
SDASC **SDA Setup Count** **[15:0]**
This 16-bit programmed value loads into the counter

- Following an extension of the SCL LOW period by the master state machine to hold the interface in a Wait state (when SDA might change with new data)
- When the master state machine is about to generate a Stop condition (when both SCL and SDA might change state)

The 16-bit counter begins counting down on each PCLK cycle. SCL is held LOW until the counter reaches 0.

Program an appropriate value for the desired speed mode. The available modes are the standard mode or the fast mode.

Register: 0x4120
API2C SDA Hold Time
Read/Write



This register sets up the clock based on the 75 MHz PCLK frequency. It is configured automatically during system initialization. It is not necessary to program this register.

This register controls the data hold time when the API2C core controls the SDA signal. The programmed value specifies the length of time that the SDA signal is stable following a falling edge on SCL. This value is in units of PCLK cycles.

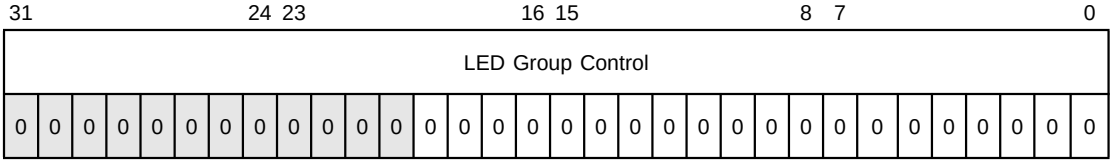
Note: This register provides the $t_{HD:DAT}$ timing parameter for the API2C interface.

R **Reserved** **[31:16]**
These bits are reserved.

SDAHC **SDA Hold Count** **[15:0]**
Whenever SDA transitions (from driven to not driven, or from not driven to driven), this value loads into a 16-bit counter that begins counting down on each PCLK cycle. The transition is delayed until the counter reaches 0.

Program an appropriate value for the desired speed mode. The available modes are the standard mode or the fast mode.

Register: 0x8000
LED Group Control
Read/Write



These configuration bits select the default mode of the pin group according to Table 4.3. Settings other than those defined in this table disable the pin group. The bit overrides in the Group Override registers supersede the settings in this register.

Table 4.3 LED Group Configuration Settings

Value	Setting	Description
0x0	GPIO	The associated Group GPIO Control registers and Group GPIO Value registers control the pin state.
0x1	Activity	The Activity setting turns an LED off if a SATA HOLD condition exists. Otherwise, the LED is on for the longer of either 32 ms or the duration from a SOF to the EOF, when a frame traverses through the link.
0x2	Fault	The Fault setting turns an LED on for any of the following conditions: • medium blink rate when waiting for permission to spin up • fast blink rate between OOB and first FIS • continuously on when the phy is not READY This output is off when the phy is connected and ready.
0x3	Combo	The Combo setting is the Exclusive-OR of the inverted Fault condition and the Activity condition.
0x4	Inverted Activity	The Inverted Activity setting asserts the LED in the same manner as the Activity setting case, except that the polarity is inverted.

Reserved **[31:20]**

Group 4 Configuration **[19:15]**
This field provides configuration for the LED_STATUS[11:0]/ signals.

Group 3 Configuration **[15:12]**
This field provides configuration for the LED_FAULT[11:0]/ signals.

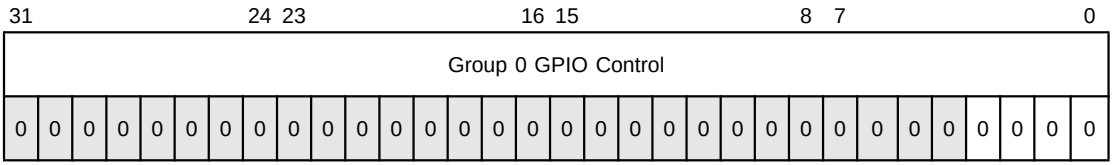
Group 2 Configuration **[11:8]**

This field provides configuration for the LED_ACTIVE[11:0]/ signals.

Group 1 Configuration **[7:4]**
This field provides configuration for the CBL_DET[11:0]/ signals.

Group 0 Configuration **[3:0]**
This field provides configuration for the GPIO[3:0] signals. The only value valid for this group is 0x0, which indicates a GPIO setting.

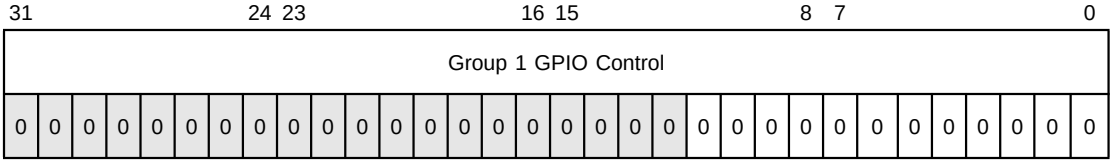
Register: 0x8004
Group 0 GPIO Control
Read/Write



Reserved **[31:4]**

GPIO Direction **[3:0]**
If the Group 0 configuration is set to GPIO, then this field controls the direction of the pin in the Group 0 pinset. Setting the associated bit configures the pin as an output. Clearing the associated bit configures the pin as an input. The Group 0 GPIO Value register controls the value of the output GPIO signal, and allows observation of the input GPIO signal.

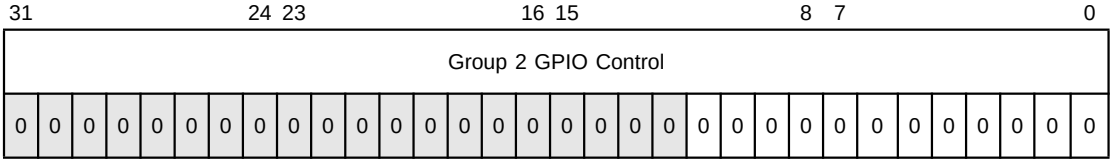
Register: 0x8008
Group 1 GPIO Control
Read/Write



Reserved **[31:12]**

GPIO Direction **[11:0]**
If the Group 1 configuration is set to GPIO, then this field controls the direction of the pin in the Group 1 pinset. Setting the associated bit configures the pin as an output. Clearing the associated bit configures the pin as an input. The Group 1 GPIO Value register controls the value of the output GPIO signal, and allows observation of the input GPIO signal.

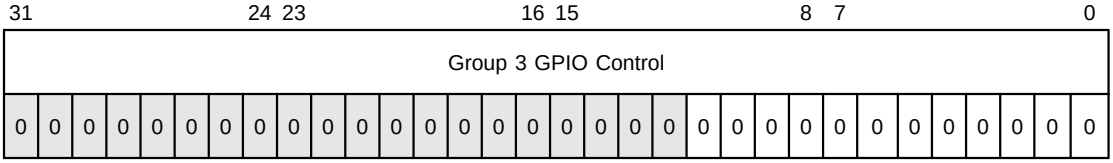
Register: 0x800C
Group 2 GPIO Control
Read/Write



Reserved **[31:12]**

GPIO Direction **[11:0]**
If the Group 2 configuration is set to GPIO, then this field controls the direction of the pin in the Group 2 pinset. Setting the associated bit configures the pin as an output. Clearing the associated bit configures the pin as an input. The Group 2 GPIO Value register controls the value of the output GPIO signal, and allows observation of the input GPIO signal.

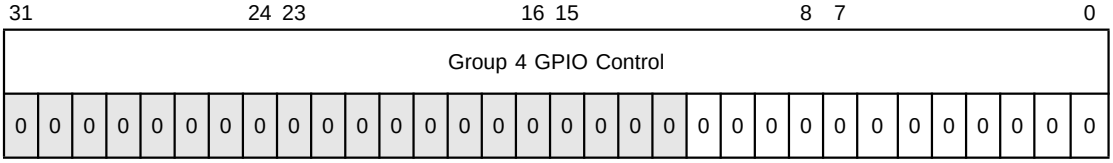
Register: 0x8010
Group 3 GPIO Control
Read/Write



Reserved **[31:12]**

GPIO Direction **[11:0]**
If the Group 3 configuration is set to GPIO, then this field controls the direction of the pin in the Group 3 pinset. Setting the associated bit configures the pin as an output. Clearing the associated bit configures the pin as an input. The Group 3 GPIO Value register controls the value of the output GPIO signal, and allows observation of the input GPIO signal.

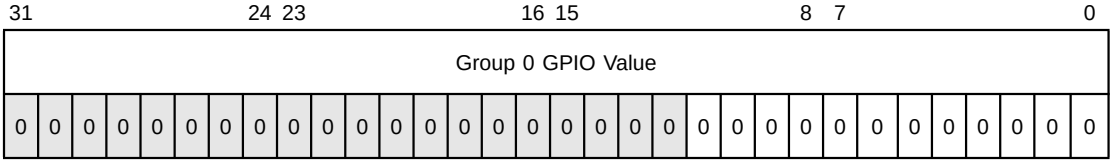
Register: 0x8014
Group 4 GPIO Control
Read/Write



Reserved **[31:12]**

GPIO Direction **[11:0]**
If the Group 4 configuration is set to GPIO, then this field controls the direction of the pin in the Group 4 pinset. Setting the associated bit configures the pin as an output. Clearing the associated bit configures the pin as an input. The Group 4 GPIO Value register controls the value of the output GPIO signal, and allows observation of the input GPIO signal.

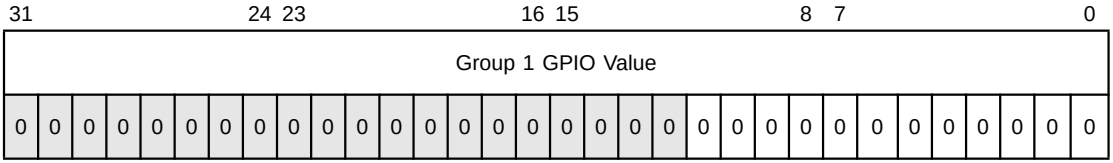
Register: 0x8018
Group 0 GPIO Value
Read/Write



Reserved **[31:12]**

GPIO Value **[11:0]**
If the Group 0 configuration is set to GPIO, then these bits control or observe the state of the associated pin. A write of this register sets the output value, and a read of this register returns the input value. If the pin is configured as an input then there is no method to determine to the most recent output value.

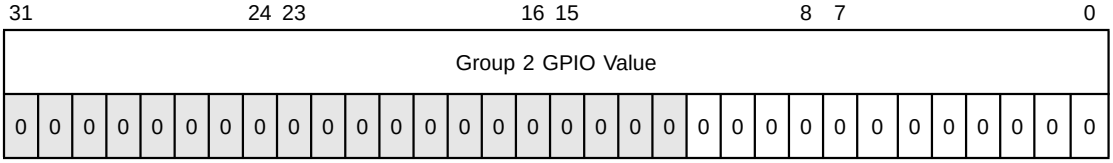
Register: 0x801C
Group 1 GPIO Value
Read/Write



Reserved **[31:12]**

GPIO Value **[11:0]**
If the Group 1 configuration is set to GPIO, then these bits control or observe the state of the associated pin. A write of this register sets the output value, and a read of this register returns the input value. If the pin is configured as an input then there is no method to determine to the most recent output value.

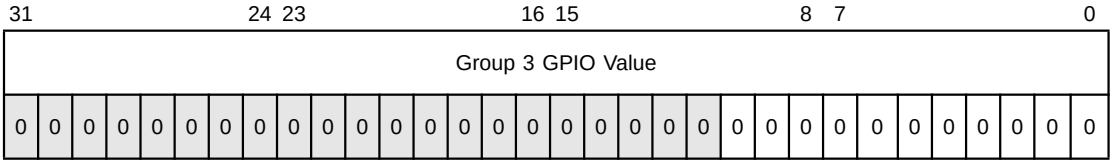
Register: 0x8020
Group 2 GPIO Value
Read/Write



Reserved **[31:12]**

GPIO Value **[11:0]**
If the Group 2 configuration is set to GPIO, then these bits control or observe the state of the associated pin. A write of this register sets the output value, and a read of this register returns the input value. If the pin is configured as an input then there is no method to determine to the most recent output value.

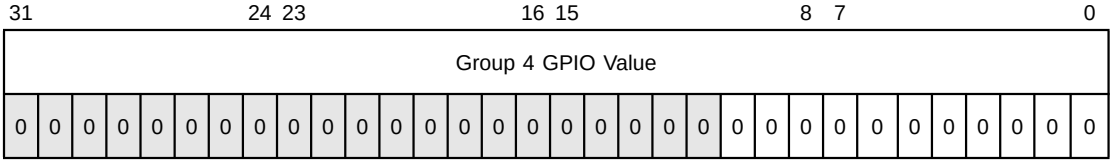
Register: 0x8024
Group 3 GPIO Value
Read/Write



Reserved **[31:12]**

GPIO Value **[11:0]**
If the Group 3 configuration is set to GPIO, then these bits control or observe the state of the associated pin. A write of this register sets the output value, and a read of this register returns the input value. If the pin is configured as an input then there is no method to determine to the most recent output value.

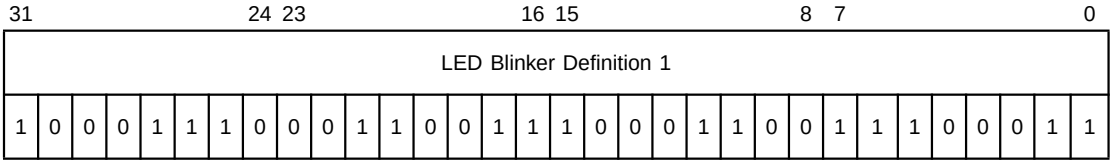
Register: 0x8028
Group 4 GPIO Value
Read/Write



Reserved **[31:12]**

GPIO Value **[11:0]**
If the Group 4 configuration is set to GPIO, then these bits control or observe the state of the associated pin. A write of this register sets the output value, and a read of this register returns the input value. If the pin is configured as an input then there is no method to determine to the most recent output value.

Register: 0x8030
LED Blinker Definition 1
Read/Write



This register defines the operation of the Blinker Control 1. This generic blinker definition can drive any pin in a group. The Group Override registers can associate this register with a specific pin when the Group Configuration is not set to GPIO. The default for this register is approximately a 1/3 s pulse.

Time Base **31**
This field selects the time base for the blink generator rotater. Setting this bit programs the time base to 64 ms, for a total period of 1.92 s. Clearing this bit programs the time base to 16 ms, for a total period of 0.48 s.

Use LEDSYNC_OUT Pin **30**
Setting this bit causes the blink generator to ignore the Time Base and Pattern fields, and to source its output from the LEDSYNC_OUT pin.

Blink Pattern **[29:0]**
This field rotates to the right at the rate selected by the Time Base bit. Bit 0 of the shifted pattern to the pin determines the LED state.

Register: 0x8034
LED Blinker Definition 2
Read/Write

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LED Blinker Definition 2																															
1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1

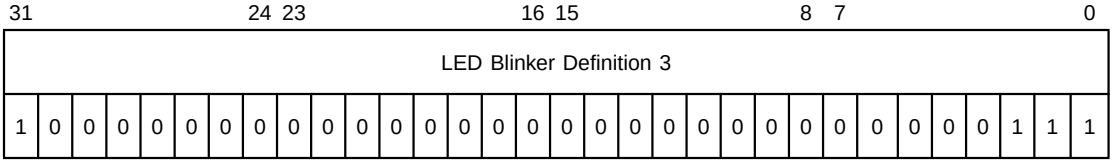
This register defines the operation of the Blinker Control 2. This generic blinker definition can drive any pin in a group. The Group Override registers can associate this register with a specific pin when the Group Configuration is not set to GPIO. The default for this register is approximately a 1 s pulse.

Time Base **31**
This field selects the time base for the blink generator rotater. Setting this bit programs the time base to 64 ms, for a total period of 1.92 s. Clearing this bit programs the time base to 16 ms, for a total period of 0.48 s.

Use LEDSYNC_OUT Pin **30**
Setting this bit causes the blink generator to ignore the Time Base and Pattern fields, and to source its output from the LEDSYNC_OUT pin.

Blink Pattern **[29:0]**
This field rotates to the right at the rate selected by the Time Base bit. Bit 0 of the shifted pattern to the pin determines the LED state.

Register: 0x8038
LED Blinker Definition 3
Read/Write



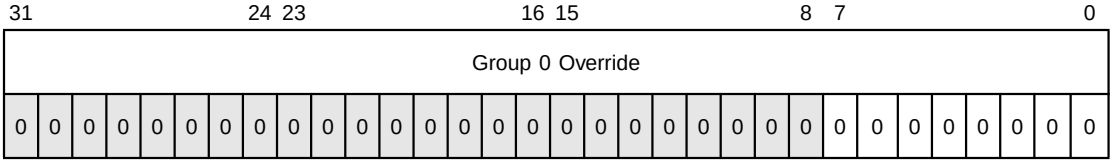
This register defines the operation of the blinker control 3. This generic blinker definition can drive any pin in a group. The Group Override registers can associate this register with a specific pin when the Group Configuration is not set to GPIO. The default for this register is intermittent blinking, approximately 1/5 s every 2 s.

Time Base **31**
This field selects the time base for the blink generator rotater. Setting this bit programs the time base to 64 ms, for a total period of 1.92 s. Clearing this bit programs the time base to 16 ms, for a total period of 0.48 s.

Use LEDSYNC_OUT Pin **30**
Setting this bit causes the blink generator to ignore the Time Base and Pattern fields, and to source its output from the LEDSYNC_OUT pin.

Blink Pattern **[29:0]**
This field rotates to the right at the rate selected by the Time Base bit. Bit 0 of the shifted pattern to the pin determines the LED state.

Register: 0x8040
Group 0 Override
Read/Write



If the Group Configuration field is not set to GPIO, then this field affects the pin operation according to Table 4.4. Note the transition to or from a custom blinker setting is immediate. The pattern is not completed when transitioning out. The pattern starts where it last was in its rotation when transitioning in.

Table 4.4 Group Override Pin Operation

Value	Function
0x00	No override. The pin functions according to their group configuration.
0x01	LED Blinker Definition 1 is used.
0x02	LED Blinker Definition 2 is used.
0x03	LED Blinker Definition 3 is used.

Reserved **[31:8]**

Bit 3 Override **[7:6]**

Bit 2 Override **[5:4]**

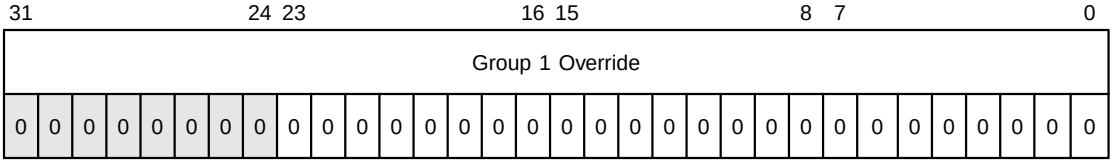
Bit 1 Override **[3:2]**

Bit 0 Override **[1:0]**

Register: 0x8048

Group 1 Override

Read/Write



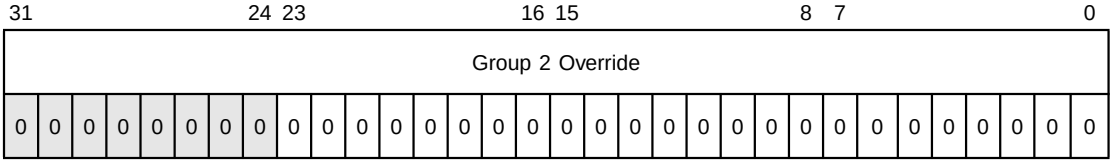
If the Group Configuration field is not set to GPIO, then this field affects the pin operation according to Table 4.4. Note the transition to or from a custom blinker setting is immediate. The pattern is not completed when transitioning out. The pattern starts where it last was in its rotation when transitioning in.

Reserved	**[31:24]**
Bit 11 Override	**[23:22]**
Bit 10 Override	**[21:20]**
Bit 9 Override	**[19:18]**
Bit 8 Override	**[17:16]**
Bit 7 Override	**[15:14]**
Bit 6 Override	**[13:12]**
Bit 5 Override	**[11:10]**
Bit 4 Override	**[9:8]**
Bit 3 Override	**[7:6]**
Bit 2 Override	**[5:4]**
Bit 1 Override	**[3:2]**
Bit 0 Override	**[1:0]**

Register: 0x8050

Group 2 Override

Read/Write



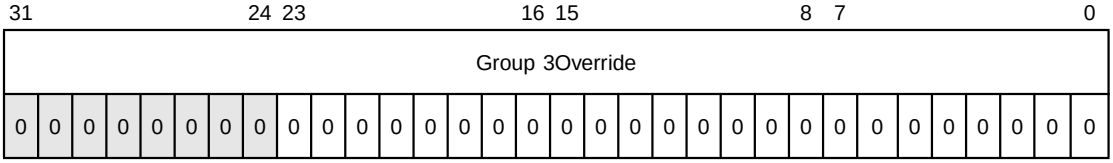
If the Group Configuration field is not set to GPIO, then this field affects the pin operation according to Table 4.4. Note the transition to or from a custom blinker setting is immediate. The pattern is not completed when transitioning out. The pattern starts where it last was in its rotation when transitioning in.

Reserved	**[31:24]**
Bit 11 Override	**[23:22]**
Bit 10 Override	**[21:20]**
Bit 9 Override	**[19:18]**
Bit 8 Override	**[17:16]**
Bit 7 Override	**[15:14]**
Bit 6 Override	**[13:12]**
Bit 5 Override	**[11:10]**
Bit 4 Override	**[9:8]**
Bit 3 Override	**[7:6]**
Bit 2 Override	**[5:4]**
Bit 1 Override	**[3:2]**
Bit 0 Override	**[1:0]**

Register: 0x8058

Group 3 Override

Read/Write



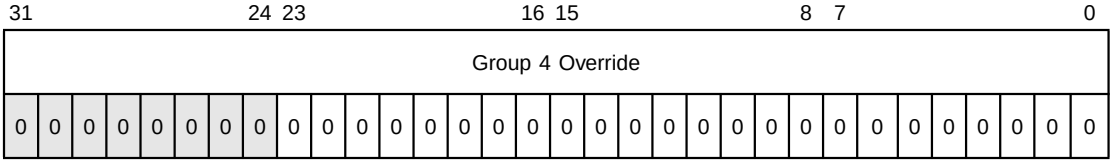
If the Group Configuration field is not set to GPIO, then this field affects the pin operation according to Table 4.4. Note the transition to or from a custom blinker setting is immediate. The pattern is not completed when transitioning out. The pattern starts where it last was in its rotation when transitioning in.

Reserved	**[31:24]**
Bit 11 Override	**[23:22]**
Bit 10 Override	**[21:20]**
Bit 9 Override	**[19:18]**
Bit 8 Override	**[17:16]**
Bit 7 Override	**[15:14]**
Bit 6 Override	**[13:12]**
Bit 5 Override	**[11:10]**
Bit 4 Override	**[9:8]**
Bit 3 Override	**[7:6]**
Bit 2 Override	**[5:4]**
Bit 1 Override	**[3:2]**
Bit 0 Override	**[1:0]**

Register: 0x8060

Group 4 Override

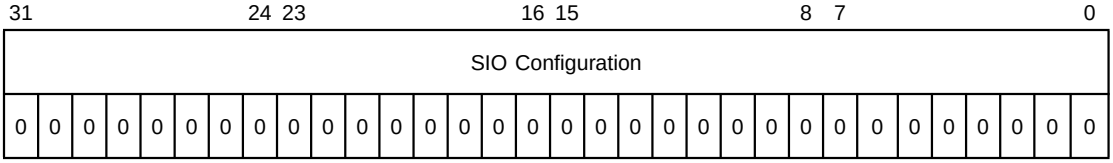
Read/Write



If the Group Configuration field is not set to GPIO, then this field affects the pin operation according to Table 4.4. Note the transition to or from a custom blinker setting is immediate. The pattern is not completed when transitioning out. The pattern starts where it last was in its rotation when transitioning in.

Reserved	**[31:24]**
Bit 11 Override	**[23:22]**
Bit 10 Override	**[21:20]**
Bit 9 Override	**[19:18]**
Bit 8 Override	**[17:16]**
Bit 7 Override	**[15:14]**
Bit 6 Override	**[13:12]**
Bit 5 Override	**[11:10]**
Bit 4 Override	**[9:8]**
Bit 3 Override	**[7:6]**
Bit 2 Override	**[5:4]**
Bit 1 Override	**[3:2]**
Bit 0 Override	**[1:0]**

Register: 0xC000
SIO Configuration
Read/Write



This SIO module participates in the generation of an SIO output data stream as an originator.

Mux Control Code **[31:28]**
When bit 0 of this register is set, the value in this field is the mux control code that is transmitted on the SioEnd line after setting the Transmit Mux Control bit. Encodings for the value in this register are receiver defined except for bit 28 (MX0), which determines whether the cycle transmits and receives device data or general purpose data. Bit 28 (MX0) is transmitted first.

Transmit Mux Control **27**
When bit 0 of this register is set, setting this bit causes the SIO to transmit the value in Mux Control Code in the next available transmit cycle. The SIO transmits the Mux Control Code one time only. When bit 0 of this is register cleared, the SIO ignores writes to this field.

Setting this bit may produce unexpected results when the SIO Mode bit is cleared and SIO Device Count is set to 0b0001.

SIO Device Count **[26:23]**
This field provides the number of devices that require LED output bits and input bits. The number of bits to be output on the SioDout line and the number of bits to be input on the SioDin line is 3 times the SIO Device Count ($3 \times$ (SIO Device Count)). Valid values are 1 to 12, which provide for 3 to 36 output bits and 3 to 36 input bits. Bit 0 is always transmitted first.

To send out the minimum number of bits and to eliminate the requirement that receivers skip over unused output bits, attach devices to consecutive phys, starting with Phy 0.

The first input bit of the receive cycle must be ignored.

Do not set this field to a value less than 2 when SIO Mode is set to a 1.

SIO Run Enable **22**
This bit enables transmit and receive operations on the SIO interface. Resets disable transmit and receive operations on the SIO interface until this bit is set. Once enabled, transmit and receive operations continue until this bit is cleared. Any transmit and receive operation in progress when this bit is cleared continue to completion.

Clock Divide **[21:7]**
This field sets the clock divide value for the SioClk clock generation logic. Equation 4.1 provides the frequency of SioClk.

Equation 4.1 Frequency = 75 MHz 2((Clock Divide × 8) + 1)

The lowest frequency available is approximately 143 Hz. The highest operational frequency is 200 KHz.

SIO Control ODEn **6**
Setting this bit drives SioClk and SioEnd LOW, but permits them to float HIGH. Clearing this bit drives SioClk and SioEnd to both the HIGH and the LOW states.

SIO Clock Filter Enable **5**
Setting this bit causes clock filtering of the selected SioClk.

Jog Activity **4**
Setting this bit causes bits representing the Activity signal in the serial bit stream to deassert for 256 ms after they assert for 4 seconds. Clearing this bit allows these bits to stay asserted for as long as the Active input remains in the asserted state.

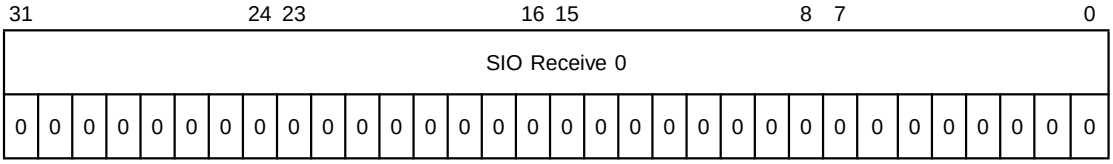
Clock Enable **3**
Setting this bit configures the SioClk and BlinkClk internal signals as inputs. Clearing this bit configures these signals as outputs.

SIO Enable **2**
Setting this bit selects SIO functionality for the SIO pins.

SIO Mode **1**
This bit indicates the SIO mode. Clearing this bit configures the SIO for a single participant. Setting this bit configures the SIO for multiple participants.

First SIO Device **0**
This bit is set to indicate that the device is the first SIO data transmitter (First Originator), and can append mux control codes to the SioEnd signal. When this bit is cleared, the device can not append mux control codes to the SioEnd signal, but must pass on any codes it receives.

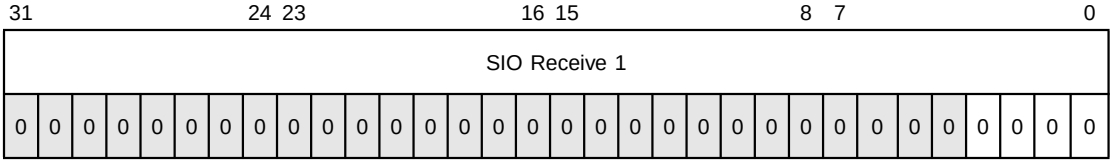
Register: 0xC004
SIO Receive 0
Read/Write



SIO Receive 0 **[31:0]**
The SIO receive registers (SIO_RCV0 and SIO_RCV1) receive data directly from the serial SIO at the end of each normal serial input/output cycle. The number of bits shifted in (and therefore the number of valid inputs bits) is 3 times the SIO Device Count. The SIO Device Count is located in the SIO Configuration register.

Bits [(((SioDeviceCnt) * 3) – 1):0] of this register contain the data shifted in during the most recent serial input/output cycle. When the SIO Device Count is set to 10 or 11, bits [31:0] of this register contain the data from bits [31:0] of the input shift register, and the rest of the bits are available in the SIO Receive 1 register.

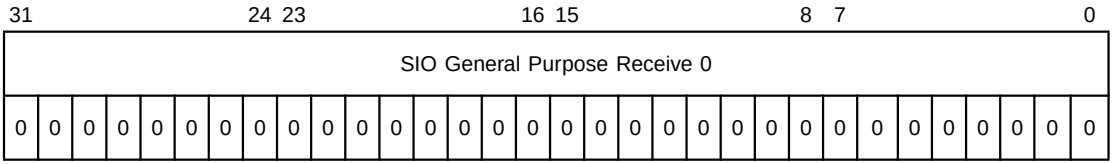
Register: 0xC008
SIO Receive 1
Read/Write



Reserved **[31:4]**

SIO Receive 1 **[3:0]**
When the SIO Device Count is set to 10 or 11, bit 0 of this register contains bit 32 from the SioDin line and bits [3:1] of this register are reserved. When SIO Device Count is set to 11, bits [3:0] of this register contain bits [35:32] from the SioDin line.

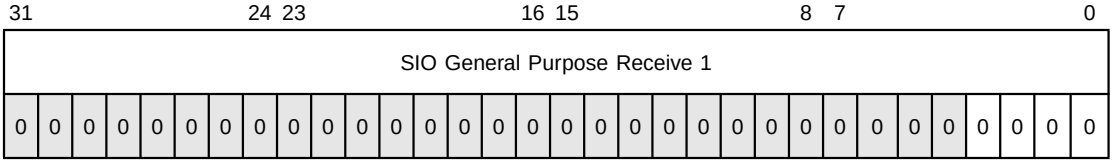
Register: 0xC00C
SIO General Purpose Receive 0
Read/Write



SIO General Purpose Receive 0 **[31:0]**
The SIO receive registers (SIO_RCV0 and SIO_RCV1) receive data directly from the serial SIO at the end of each normal serial input/output cycle. The number of bits shifted in (and therefore the number of valid inputs bits) is 3 times the SIO Device Count. The SIO Device Count is located in the SIO Configuration register.

Bits [(((SioDeviceCnt) * 3) – 1):0] of this register contain the data shifted in during the most recent serial input/output cycle. When the SIO Device Count is set to 10 or 11, bits [31:0] of this register contain the data from bits [31:0] of the input shift register, and the rest of the bits are available in the SIO General Purpose Receive 1 register.

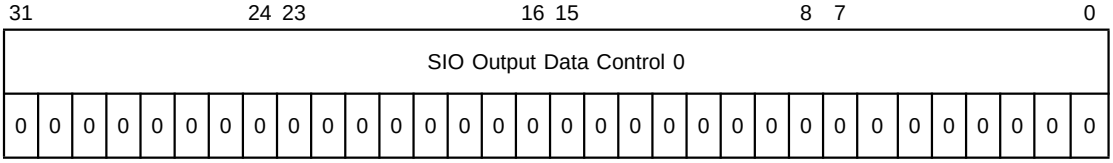
Register: 0xC010
SIO General Purpose Receive 1
Read/Write



Reserved **[31:4]**

SIO General Purpose Receive 0 **[3:0]**
When the SIO Device Count is set to 10 or 11, bit 0 of this register contains bit 32 from the SioDin line and bits [3:1] of this register are reserved. When SIO Device Count is set to 11, bits [3:0] of this register contain bits [35:32] from the SioDin line.

Register: 0xC014
SIO Output Data Control 0
Read/Write



The SIO Output Data Control registers provide data output control for PHY[11:0]. Each SIO Output Data register uses one of two possible bit field definitions. One bit field definition provides the bit definitions for the standard SGPIO mode; the other bit field definition provides the bit definitions for the enhanced SGPIO mode only available on the LSISASx12A.

Each byte of this register generates 3 serialized I/O bits on the SioDout line. This register controls SIO bits [11:0], which are for devices 3 through 0.

The following are the bit definitions for standard SIO mode:

SIO bit 11: Error Device 3 **[31:29]**

SIO bit 10: Locate Device 3	**[28:27]**
SIO bit 9: Active Device 3	**[26:24]**
SIO bit 8: Error Device 2	**[23:21]**
SIO bit 7: Locate Device 2	**[20:19]**
SIO bit 6: Active Device 2	**[18:16]**
SIO bit 5: Error Device 1	**[15:13]**
SIO bit 4: Locate Device 1	**[12:11]**
SIO bit 3: Active Device 1	**[10:8]**
SIO bit 2: Error Device 0	**[7:5]**
SIO bit 1: Locate Device 0	**[4:3]**
SIO bit 0: Active Device 0	**[2:0]**

The following are the bit definitions for enhanced SIO mode:

SIO bit 11: Error Device 3	**[31:29]**
SIO bit 10: Locate Device 3	**[28:26]**
SIO bit 9: Active Device 3	**[25:24]**
SIO bit 8: Error Device 2	**[23:21]**
SIO bit 7: Locate Device 2	**[20:18]**
SIO bit 6: Active Device 2	**[17:16]**
SIO bit 5: Error Device 1	**[15:13]**
SIO bit 4: Locate Device 1	**[12:10]**
SIO bit 3: Active Device 1	**[9:8]**
SIO bit 2: Error Device 0	**[7:5]**
SIO bit 1: Locate Device 0	**[4:2]**
SIO bit 0: Active Device 0	**[1:0]**

When the SIO Control Select bit in the SIO Adapter Control register 0xC054 is cleared, the standard SGPIO mode is enabled and the bit encodings are.

Active Field Encoding	Definition
0b000	OFF
ob001	Activity
0b010	Blink at 1 Hz (500 ms ON, 500 ms OFF)
0b011	Blink at Hz (250 ms ON, 250 ms OFF)
0b100	ON
0b101	Inverted Activity
0b110	Inverted Blink at 1 Hz (500 ms OFF, 500 ms ON)
0b111	Inverted Blink at Hz (250 ms OFF, 250 ms ON)

Locate Field Encoding	Definition
0b00	OFF
ob01	ON
0b10	Blink at 1 Hz (500 ms ON, 500 ms OFF)
0b11	Blink at Hz (250 ms ON, 250 ms OFF)

Error Field Encoding	Definition
0b000	OFF
0b001	ON
0b010	Blink at 1 Hz (500 ms ON, 500 ms OFF)
0b011	Blink at Hz (250 ms ON, 250 ms OFF)
0b100	ON
0b101	OFF
0b110	Inverted Blink at 1 Hz (500 ms OFF, 500 ms ON)
0b111	Inverted Blink at Hz (250 ms OFF, 250 ms ON)

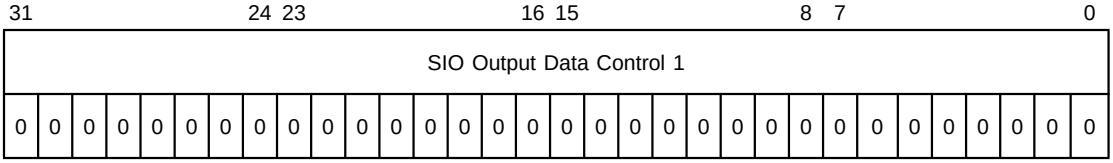
When the SIO Control Select bit in the SIO Adapter Control register 0xC054 is set, the enhanced SGPIO mode is enabled and the bit encodings are:

Active Field Encoding	Definition
0b00	OFF
ob01	ON
0b10	Activity
0b11	Inverted Activity

Locate Field Encoding	Definition
0b000	OFF
0b001	ON
0b010	Pattern 0 (see SIO Pattern Definition 0 register)
0b011	Pattern 1 (see SIO Pattern Definition 1 register)
0b100	Pattern 2 (see SIO Pattern Definition 2 register)
0b101	Pattern 3 (see SIO Pattern Definition 3 register)
0b110	Pattern 3 with a 90 phase difference (see SIO Pattern Definition 3 register)
0b111	Pattern 3 with a 180 phase difference (see SIO Pattern Definition 3 register)

Error Field Encoding	Definition
0b000	OFF
0b001	ON
0b010	Pattern 0 (see SIO Pattern Definition 0 register)
0b011	Pattern 1 (see SIO Pattern Definition 1 register)
0b100	Pattern 2 (see SIO Pattern Definition 2 register)
0b101	Pattern 3 (see SIO Pattern Definition 3 register)
0b110	Pattern 3 with a 90 phase difference (see SIO Pattern Definition 3 register)
0b111	Pattern 3 with a 180 phase difference (see SIO Pattern Definition 3 register)

Register: 0xC018
SIO Output Data Control 1
Read/Write



The SIO Output Data Control registers provide data output control for PHY[11:0]. Each SIO Output Data register uses one of two possible bit field definitions. One bit field definition provides the bit definitions for the standard SGPIO mode; the other bit field definition provides the bit definitions for the enhanced SGPIO mode only available on the LSISASx12A.

Each byte of this register generates 3 serialized I/O bits on the SioDout line. This register controls SIO bits [23:12], which are for devices 7 through 4.

See the encoding for Error, Active, and Locate bits above as defined in register 0xC014.

The following are the bit definitions for standard SIO mode:

SIO bit 23: Error Device 7 **[31:29]**

SIO bit 22: Locate Device 7 **[28:27]**

SIO bit 21: Active Device 7 **[26:24]**

SIO bit 20: Error Device 6 **[23:21]**

SIO bit 19: Locate Device 6 **[20:19]**

SIO bit 18: Active Device 6 **[18:16]**

SIO bit 17: Error Device 5 **[15:13]**

SIO bit 16: Locate Device 5 **[12:11]**

SIO bit 15: Active Device 5 **[10:8]**

SIO bit 14: Error Device 4 **[7:5]**

SIO bit 13: Locate Device 4 **[4:3]**

SIO bit 12: Active Device 4 **[2:0]**

The following are the bit definitions for enhanced SIO mode:

SIO bit 11: Error Device 7 **[31:29]**

SIO bit 10: Locate Device 7 **[28:26]**

SIO bit 9: Active Device 7 **[25:24]**

SIO bit 8: Error Device 6 **[23:21]**

SIO bit 7: Locate Device 6 **[20:18]**

SIO bit 6: Active Device 6 **[17:16]**

SIO bit 5: Error Device 5 **[15:13]**

SIO bit 4: Locate Device 5 **[12:10]**

SIO bit 3: Active Device 5 **[9:8]**

SIO bit 2: Error Device 4 **[7:5]**

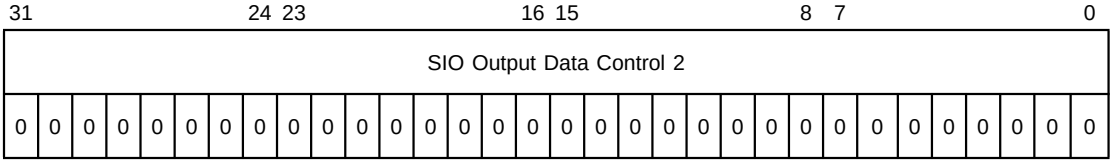
SIO bit 1: Locate Device 4 **[4:2]**

SIO bit 0: Active Device 4 **[1:0]**

Register: 0xC01C

SIO Output Data Control 2

Read/Write



The SIO Output Data Control registers provide data output control for PHY[11:0]. Each SIO Output Data register uses one of two possible bit field definitions. One bit field definition provides the bit definitions for the standard SGPIO mode; the other bit field definition provides the bit definitions for the enhanced SGPIO mode only available on the LSISASx12A.

Each byte of this register generates 3 serialized I/O bits on the SioDout line. This register controls SIO bits [35:24], which are for devices 11 through 8.

See the encoding for Error, Active, and Locate bits above as defined in register 0xC014.

The following are the bit definitions for standard SIO mode:

SIO bit 35: Error Device 11	**[31:29]**
SIO bit 34: Locate Device 11	**[28:27]**
SIO bit 33: Active Device 11	**[26:24]**
SIO bit 32: Error Device 10	**[23:21]**
SIO bit 31: Locate Device 10	**[20:19]**
SIO bit 30: Active Device 10	**[18:16]**
SIO bit 29: Error Device 9	**[15:13]**
SIO bit 28: Locate Device 9	**[12:11]**
SIO bit 27: Active Device 9	**[10:8]**
SIO bit 26: Error Device 8	**[7:5]**
SIO bit 25: Locate Device 8	**[4:3]**
SIO bit 24: Active Device 8	**[2:0]**

The following are the bit definitions for enhanced SIO mode:

SIO bit 11: Error Device 11	**[31:29]**
SIO bit 10: Locate Device 11	**[28:26]**
SIO bit 9: Active Device 11	**[25:24]**
SIO bit 8: Error Device 10	**[23:21]**
SIO bit 7: Locate Device 10	**[20:18]**
SIO bit 6: Active Device 10	**[17:16]**
SIO bit 5: Error Device 9	**[15:13]**

SIO bit 4: Locate Device 9 **[12:10]**

SIO bit 3: Active Device 9 **[9:8]**

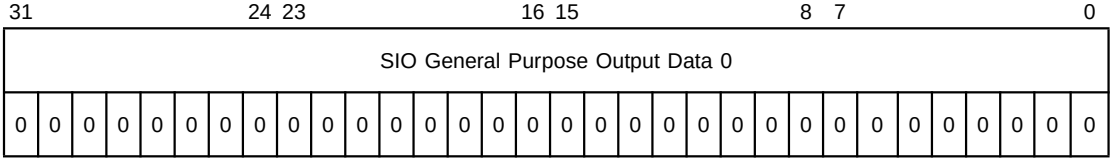
SIO bit 2: Error Device 8 **[7:5]**

SIO bit 1: Locate Device 8 **[4:2]**

SIO bit 0: Active Device 8 **[1:0]**

Register: 0xC020

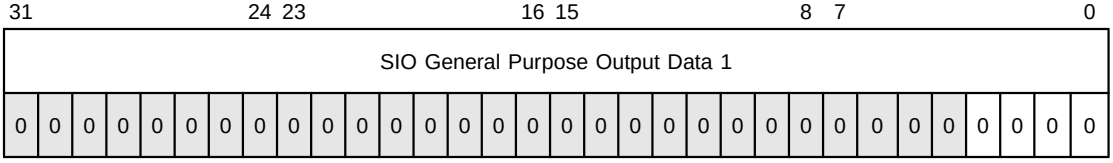
SIO General Purpose Output Data 0
Read/Write



General Purpose Output Data [31:0] **[31:0]**
These bits are transmitted bit for bit.

Register: 0xC024

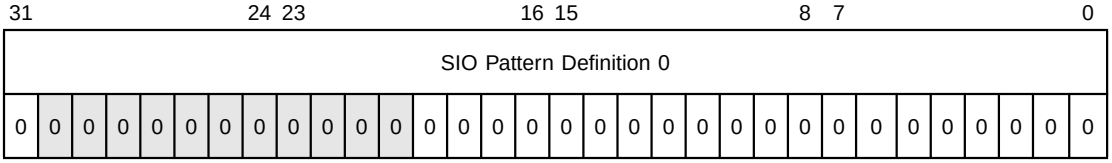
SIO General Purpose Output Data 1
Read/Write



Reserved **[31:4]**

General Purpose Output Data [35:32] **[3:0]**
These bits are transmitted bit for bit.

Register: 0xC040
SIO Pattern Definition 0
Read/Write



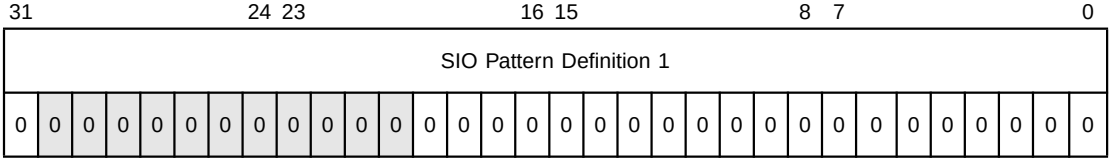
This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

R/W **Timebase 0** **31**
Clearing this bit causes the pattern to rotate at 40 Hz, which provides an interval of 25 ms between each bit shift. Setting this bit causes the pattern to rotate at 10 Hz, which provides an interval of 100 ms between each bit shift.

Reserved **[30:20]**

R/W **Pattern 0** **[19:0]**
This field contains pattern 0. This pattern is sampled at bit 0.

Register: 0xC044
SIO Pattern Definition 1
Read/Write



This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

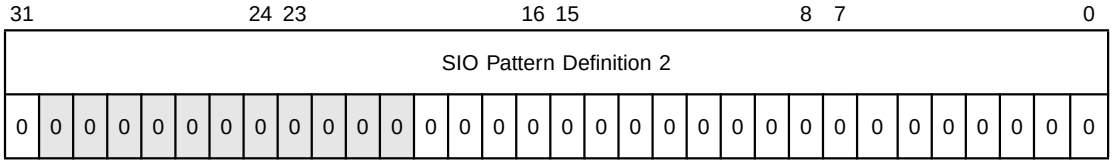
Timebase 1 **31**
Clearing this bit causes the pattern to rotate at 40 Hz, which provides an interval of 25 ms between each bit

shift. Setting this bit causes the pattern to rotate at 10 Hz, which provides an interval of 100 ms between each bit shift.

Reserved **[30:20]**

Pattern 1 **[19:0]**
This field contains pattern 1. This pattern is sampled at bit 0.

Register: 0xC048
SIO Pattern Definition 2
Read/Write



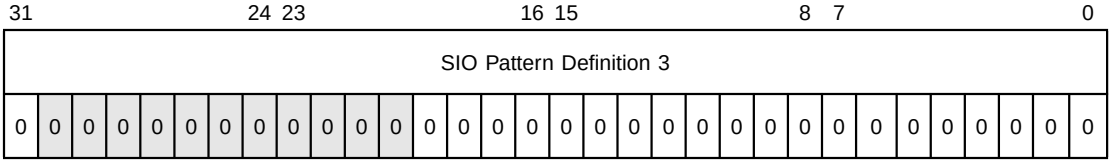
This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

Timebase 2 **31**
Clearing this bit causes the pattern to rotate at 40 Hz, which provides an interval of 25 ms between each bit shift. Setting this bit causes the pattern to rotate at 10 Hz, which provides an interval of 100 ms between each bit shift.

Reserved **[30:20]**

Pattern 2 **[19:0]**
This field contains pattern 2. This pattern is sampled at bit 0.

Register: 0xC04C
SIO Pattern Definition 3
Read/Write



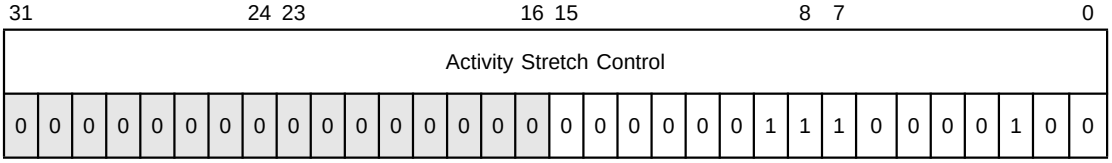
This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

Timebase 3 **31**
Clearing this bit causes the pattern to rotate at 40 Hz, which provides an interval of 25 ms between each bit shift. Setting this bit causes the pattern to rotate at 10 Hz, which provides an interval of 100 ms between each bit shift.

Reserved **[30:20]**

Pattern 3 **[19:0]**
This field contains pattern 3. This pattern is sampled at bits 0, 5, and 10 to allow for 0/90/180 phase shifts in the output pattern.

Register: 0xC050
Activity Stretch Control
Read/Write



This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

Minimum Deassert Time **[15:12]**

This field provides the minimum amount of time to deassert a Link Active signal. The default setting of this field is 0b0000, indicating 1/64 seconds. The encoding for this field follows.

Setting	Definition
0b0000	1/64 s
0b0001	2/64 s
0b0010	3/64 s
...	...
0b1110	15/64 s
0b1111	16/64 s

Minimum Assert **[11:8]**

This field provides the minimum amount of time to assert a Link Active signal. The default value of this field is 0b0011, indicating 4/64 seconds.The encoding for this field follows.

Setting	Definition
0b0000	1/64 s
0b0001	2/64 s
0b0010	3/64 s
0b0011	4/64 s
...	...
0b1110	15/64 s
0b1111	16/64 s

Maximum Assert **[7:4]**

Maximum amount of time to assert a Link Active signal in units of 1/4 seconds. The default value of this field is 0b1000, indicating 8/4 seconds (2 seconds). The encoding for this field follows.

Setting	Definition
0b0000	No Maximum
0b0001	1/4 s
0b0010	2/4 s
0b0011	3/4 s
...	...
0b1110	14/4 s
0b1111	15/4 s

Force Off **[3:0]**

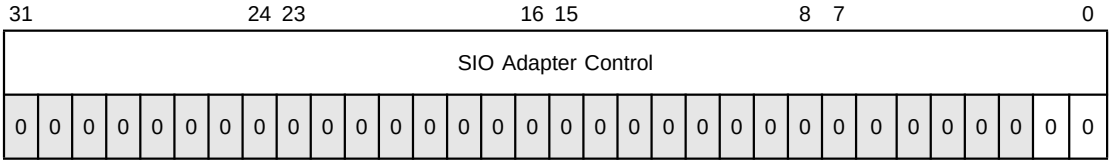
Amount of time to deassert a Link Active signal in units of 1/8 seconds following a maximum assertion time. This field is ignored if Maximum Assert field is set to 0b0000. The default value of this field is 0b0100, indicating 4/8 seconds (1/2 seconds). The encoding for this field follows.

Setting	Definition
0b0000	–
0b0001	1/8 s
0b0010	2/8 s
0b0011	3/8 s
...	...
0b1110	14/8 s
0b1111	15/8 s

Register: 0xC054

SIO Adapter Control

Read/Write



This register controls the enhanced SGPIO pattern definition on the LSISASx12A expander. This register is not present on the LSISASx12 expander.

Reserved **[31:2]**

Select C **1**

When clear, Link Activity data is provided unconditioned to SGPIO. When set, Link Activity data is adjusted according to controls in Activity Stretch Control register.

SIO Control Select **0**

This bit determines if the LSISASx12A uses the standard SGPIO mode that the LSISASx12 uses, or if the LSISASx12A uses the enhanced SPGIO mode that is described in this document.

Clearing this bit enables the standard SGPIO mode. Setting this bit enables the enhanced SGPIO mode.

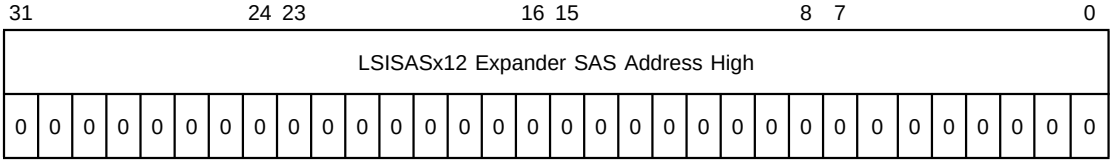
4.3 SMP Target Registers

Table 4.5 provides the SMP target register map. Detailed register descriptions follow the register map.

Table 4.5 SMP Target Register Map

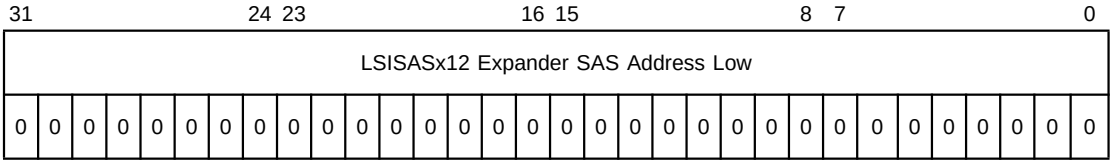
Offset	Register Name
0x00000	LSISASx12 Expander SAS Address High
0x00004	LSISASx12 Expander SAS Address Low
0x00008–0x0001C	Reserved
0x00020	Remote Routing Table Configuration 00
0x00024	Remote Routing Table Configuration 01
0x00028	Remote Routing Table Configuration 02
0x0002C	Remote Routing Table Configuration 03
0x00030	Remote Routing Table Configuration 04
0x00034	Remote Routing Table Configuration 05
0x00038	Remote Routing Table Configuration 06
0x0003C	Remote Routing Table Configuration 07
0x00040	Remote Routing Table Configuration 08
0x00044	Remote Routing Table Configuration 09
0x00048	Remote Routing Table Configuration 10
0x0004C	Remote Routing Table Configuration 11
0x00050–0x1FFFC	Reserved

Register: 0x00000
LSISASx12 Expander SAS Address High
Read/Write



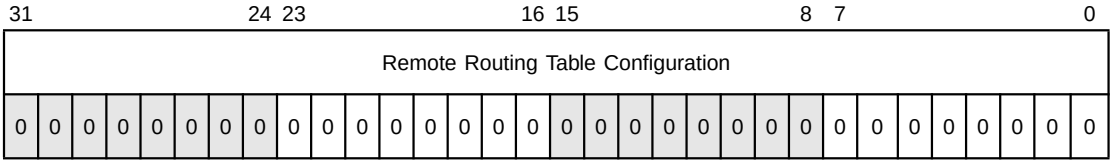
Expander SAS Address High **[31:0]**
This register provides the upper 32 bits of the 64-bit value that is used in the LoadID request from the SMP Target to the Expander Connection Manager.

Register: 0x00004
LSISASx12 Expander SAS Address Low
Read/Write



Expander SAS Address Low **[31:0]**
This register provides the lower 32 bits of the 64-bit value that is used in the LoadID request from the SMP Target to the Expander Connection Manager. Write the lower nibble of this register to 0xF.

Register: 0x00020–0x0004C
Remote Routing Table Configuration
Read/Write



These registers establish the number of routing entries for each phy. There are 12 Remote Routing Table Configuration registers; one for each phy. Table 4.5 provides the address offset for each of these registers.

The SMP Target uses these registers to convert a phy identifier and route index from the SMP REPORT ROUTE INFORMATION and CONFIGURE ROUTE INFORMATION commands into a local LSISASx12 address corresponding to a specific location within the remote route table.

Each of these registers generates an output that informs each of the 12 Sphynx modules if the routing attribute of their associated phy is set to Table Route. If any of the Route Table Size[7:0] bits are set for a phy, then the TableRoute[PhyID] is also set.

Reserved **[31:24]**

Route Table Offset NN **[23:16]**
This field specifies the route table offset for Phy[NN].

Reserved **[15:8]**

Route Table Size NN **[7:0]**
This field specifies the route table size for Phy[NN].

4.4 SPhynx Registers

This section describes the registers for each Sphynx module. Each SPhynx module consists of Link Registers and Phy Registers.

4.4.1 Link Registers

Table 4.6 provides the register map for the SPhynx Link registers.

Table 4.6 Sphynx Link Register Map

Offset	Register Name
0x00	Reserved
0x04	Identify Address Information
0x08–0x0C	Reserved
0x10	Identify SAS Address High
0x14	Identify SAS Address Low
0x18–0x40	Reserved
0x44	Discover Information 1
0x48	Discover Information 2 – SAS Address High
0x4C	Discover Information 3 – SAS Address Low
0x50	Discover Information 4 – Attached SAS Address High
0x54	Discover Information 5 – Attached SAS Address Low
0x58	Discover Information 6 – Attached Phy Identifier
0x5C	Discover Information 7 – Reserved
0x60	Discover Information 8 – Link Rates
0x64	Discover Information 9 – Routing Attributes
0x68	Discover Information 10 – Vendor Specific
0x6C	Reserved
0x70	Report SATA Phy ID
0x74	Reserved
0x78	SATA Target Address High
0x7C	SATA Target Address Low
0x80	SATA Dev2Host FIS0
0x84	SATA Dev2Host FIS1
0x88	SATA Dev2Host FIS2

Table 4.6 Sphynx Link Register Map (Cont.)

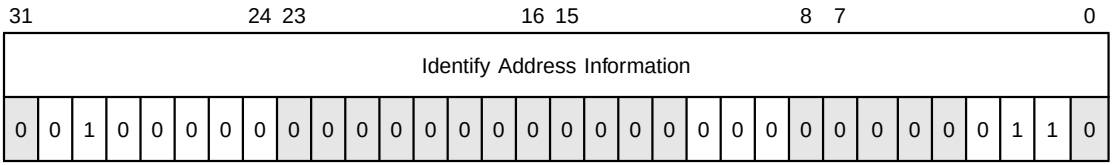
Offset	Register Name
0x8C	SATA Dev2Host FIS3
0x90–0x94	Reserved
0x98	Affiliated STP Initiator Address High
0x9C	Affiliated STP Initiator Address Low
0xA0	Phy Control Link Rates
0xA4	Phy Control PPTOV
0xA8	Reserved
0xAC	Connection Information
0xB0–0xBC	Reserved
0xC0	Broadcast Change High
0xC4	Broadcast Change Low
0xC8	Broadcast SES High
0xCC	Broadcast SES Low
0xD0–0x104	Reserved
0x108	SATA Nexus Loss Timeout
0x10C–0x11C	Reserved
0x120	SATA Connection Mode
0x124–0x140	Reserved

The following provides detailed bit definitions for the Link registers.

Register: 0x0004

Identify Address Information

Read/Write



Reserved **31**

Device Type **[30:28]**
This field must be set to 0b010 to indicate an edge expander device.

Address Frame Type **[27:24]**
This field is set to 0b000 to indicate the Identify Frame Type.

Reserved **[23:12]**

SSP Initiator **11**
Setting this bit configures the device as an SSP initiator. For normal operation, leave this bit at its default setting.

STP Initiator **10**
Setting this bit configures the device as an STP initiator. For normal operation, leave this bit at its default setting.

SMP Initiator **9**
Setting this bit configures the device as an SMP initiator. For normal operation, leave this bit at its default setting.

Reserved **[8:4]**

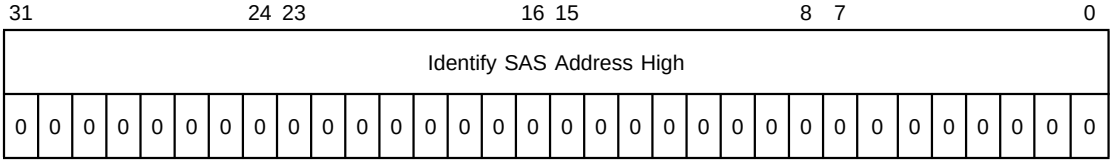
SSP Target **3**
Setting this bit configures the device as an SSP Target. For normal operation, leave this bit at its default setting.

STP Target **2**
Setting this bit configures the device as an STP Target. For normal operation, leave this bit at its default setting.

SMP Target **1**
Setting this bit configures the device as an SMP Target. For normal operation, leave this bit at its default setting.

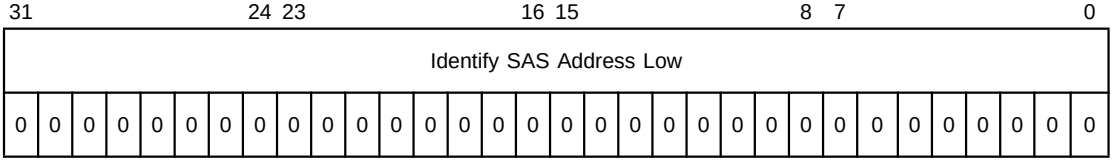
Reserved **0**

Register: 0x0010
Identify SAS Address High
Read/Write



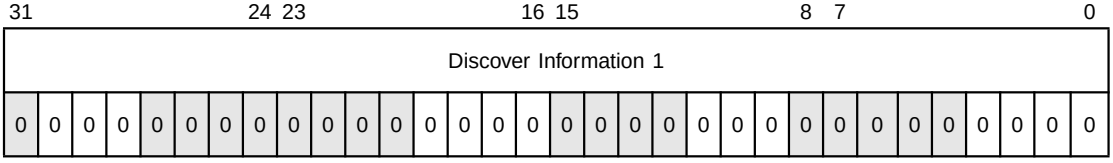
SAS Address High **[31:0]**
This field provides bits [63:32] of the LSISASx12 64-bit SAS address.

Register: 0x0014
Identify SAS Address Low
Read/Write



SAS Address Low **[31:0]**
This field provides bits [31:0] of the LSISASx12 64-bit SAS address. Set the lower nibble of this register to 0xF.

Register: 0x0044
Discover Information 1
Read/Write



Reserved **31**

Attached Device Type **[31:28]**
This read-only field provides the attached device type. The encoding of this field is

Encoding	Definition
0b000	No Device Attached
0b001	End Device Only
0b010	Edge Expander Device
0b011	Fanout Expander Device

All other encodings are reserved.

Reserved **[27:20]**

Negotiated Link Rate **[19:16]**
This read-only field provides the negotiated link rate. The encoding of this field is

Encoding	Definition
0b0000	Phy enabled with an unknown link rate.
0b0001	Phy disabled.
0b0010	Phy enabled. The speed negotiation failed.
0b0011	Phy enabled. Entered SATA spin-up hold state.
0b0100	Phy enabled. Detected SATA port selector.
0b1000	Phy enabled. 1.5 Gbits/s negotiated.
0b1001	Phy enabled. 3.0 Gbits/s negotiated.

All other encodings are reserved.

Reserved **[15:12]**

Attached SSP Initiator **11**
When set, this read-only bit indicates that the attached device is an SSP initiator.

Attached STP Initiator **10**
When set, this read-only bit indicates that the attached device is an STP initiator.

Attached SMP Initiator **9**
When set, this read-only bit indicates that the attached device is an SMP initiator.

Reserved **[8:4]**

Attached SSP Target **3**
When set, this read-only bit indicates that the attached device is an SSP target.

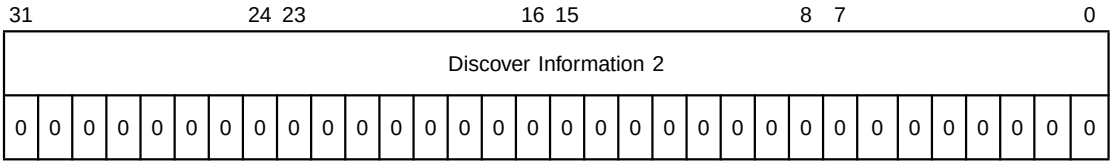
Attached STP Target **2**
When set, this read-only bit indicates that the attached device is an STP target.

Attached SMP Target **1**
When set, this read-only bit indicates that the attached device is an SMP target.

Attached SATA Target **0**
When set, this read-only bit indicates that the attached device is a SATA target.

Register: 0x0048

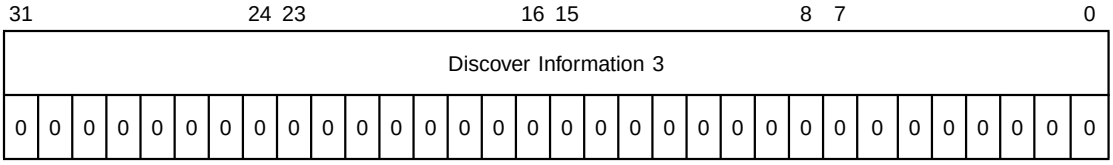
Discover Information 2 – SAS Address High
Read/Write



SAS Address High **[31:0]**
This read-only field provides bits [63:32] of the LSISASx12 SAS address.

Register: 0x004C

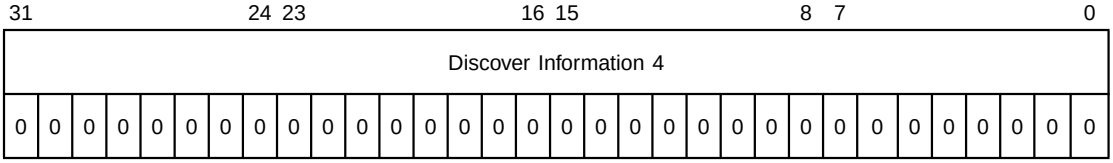
Discover Information 3 – SAS Address Low
Read/Write



SAS Address Low **[31:0]**
This read-only field provides bits [31:0] of the LSISASx12 SAS address.

Register: 0x0050

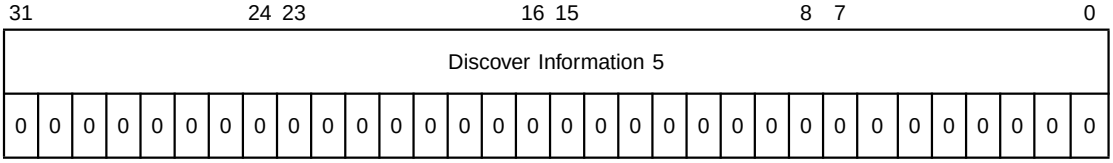
Discover Information 4 – Attached SAS Address High
Read/Write



Attached SAS Address High **[31:0]**
This read-only field provides bits [63:32] of the attached SAS address.

Register: 0x0054

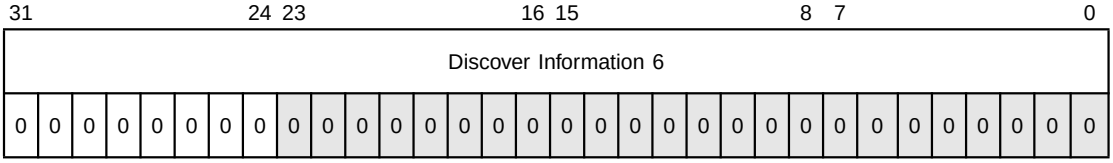
Discover Information 5 – Attached SAS Address Low
Read/Write



Attached SAS Address Low **[31:0]**
This read-only field provides bits [31:0] of the attached SAS address.

Register: 0x0058

Discover Information 6 – Attached Phy Identifier
Read/Write

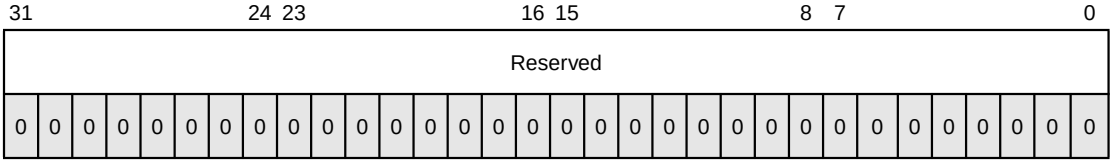


Attached Phy Identifier **[31:24]**
This read-only field provides the attached phy identifier from the identify frame.

Reserved **[23:0]**

Register: 0x005C

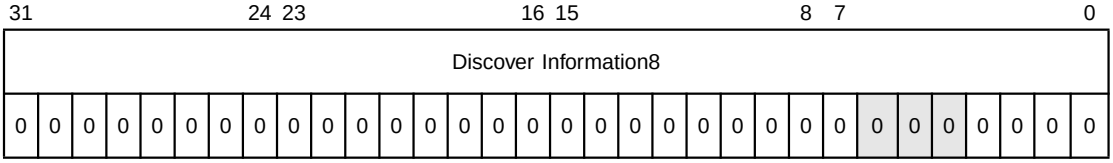
Discover Information 7 – Reserved
Read/Write



Reserved **[31:0]**

Register: 0x0060

Discover Information 8 – Link Rates
Read/Write



Programmed Minimum Physical Link Rate **[31:28]**
This field is read-only.

Hardware Minimum Physical Link Rate **[27:24]**
This field is read-only.

Programmed Maximum Physical Link Rate **[23:20]**
This field is read-only.

Hardware Maximum Physical Link Rate **[19:16]**
This field is read-only.

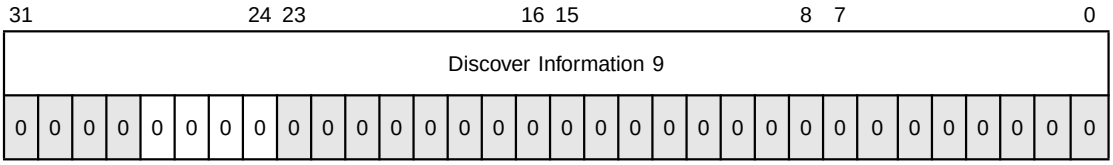
Phy Change Count **[15:8]**
This read-only field indicates the number of times that the SPhynx module has requested a BROADCAST(CHANGE) to the broadcast processor. This counter wraps to 0x00 when it reaches the maximum value of 0xFF.

Internal Phy Bit **7**
When set, this read-only bit indicates that the phy is an internal port.

Reserved **[6:4]**

PPTOV **[3:0]**
This read-only field provides the partial pathway timeout value (PPTOV).

Register: 0x0064
Discover Information 9 – Routing Attributes
Read/Write

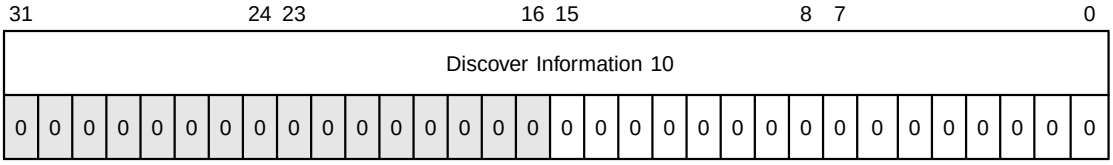


Reserved **[31:28]**

Routing Attribute **[27:24]**
This field provides the routing attributes, as defined in the SAS standard.

Reserved **[23:0]**

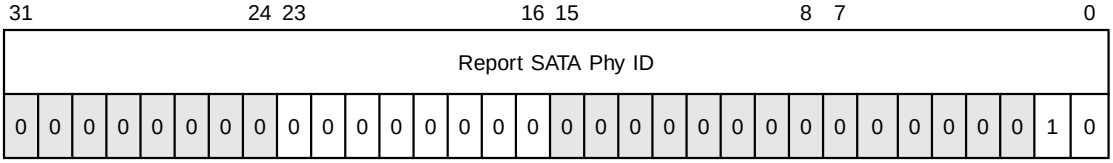
Register: 0x0068
Discover Information 10 – Vendor Specific
Read/Write



Reserved **[31:16]**

Vendor Specific **[15:0]**

Register: 0x0070
Report SATA Phy ID
Read Only



Reserved **[31:24]**

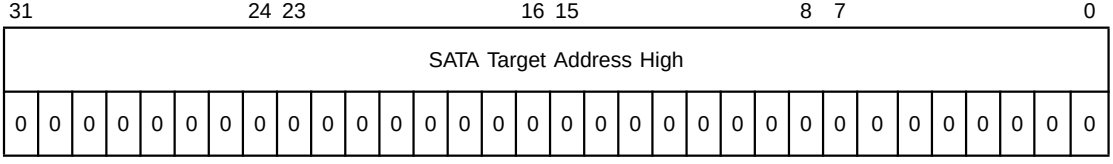
Phy Identifier **[23:16]**
This read-only field provides the Phy ID.

Reserved **[15:2]**

Affiliations Supported **1**
This read-only bit indicates if affiliations are supported.

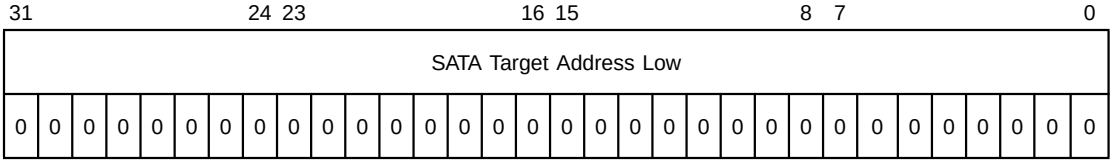
Affiliation Valid **0**
This read-only bit indicates if the affiliations are valid.

Register: 0x0078
SATA Target Address High
Read Only



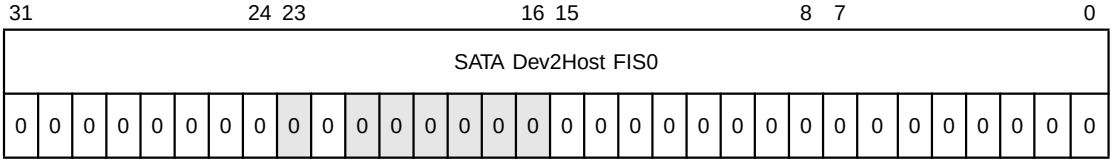
SATA Target Address High **[31:0]**
This read-only field provides bits [63:32] of the SATA target address.

Register: 0x007C
SATA Target Address Low
Read Only



SATA Target Address Low **[31:0]**
This read-only field provides bits [31:0] of the SATA target address.

Register: 0x0080
SATA Dev2Host FIS0
Read Only



FIS Type **[31:24]**
This field provides the FIS type. If the FIS type is 0x34, all Device-to-Host FISes are valid. If the FIS type is 0x00, all Dev-to-Host FISes are invalid.

Reserved **23**

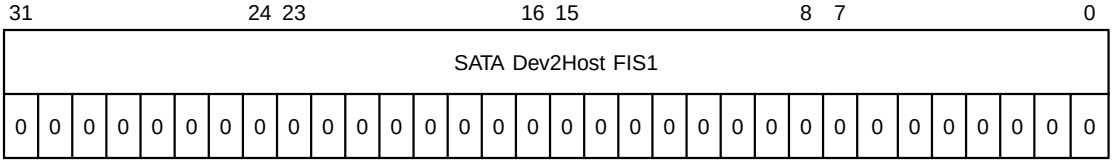
Interrupt **22**
This read-only bit reflects the status of the interrupt line of the device.

Reserved **[21:16]**

Status **[15:8]**
This read-only field contains the value of the Status register in the shadow register block.

Error **[7:0]**
This read-only field contains the value of the Error register in the shadow register block.

Register: 0x0084
SATA Dev2Host FIS1
Read Only



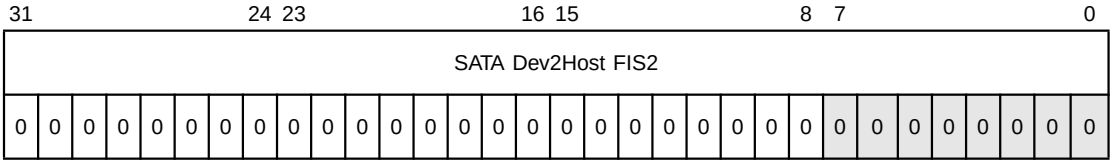
Sector Number **[31:24]**
This read-only field provides the value of the sector number register in the shadow register block.

Cyl Low **[23:16]**
This read-only field provides the value of the cylinder low register in the shadow register block.

Cyl High **[15:8]**
This read-only field provides the value of the cylinder high register in the shadow register block.

Dev/Head **[7:0]**
This read-only field provides the values of the device register and head register in the shadow register block.

Register: 0x0088
SATA Dev2Host FIS2
Read Only



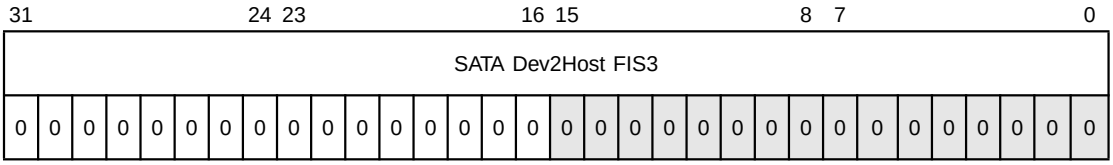
Sector Number (exp) **[31:24]**
This read-only field provides the value of the expanded sector number register in the shadow register block.

Cyl Low (exp) **[23:16]**
This read-only field provides the value of the expanded cylinder low register in the shadow register block.

Cyl High (exp) **[15:8]**
This read-only field provides the value of the expanded cylinder high register in the shadow register block.

Reserved **[7:0]**

Register: 0x008C
SATA Dev2Host FIS3
Read Only

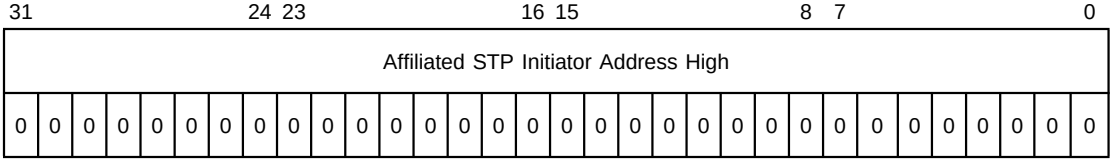


Sector Count **[31:24]**
This read-only field provides the value of the sector count register of the shadow register block.

Sector Count (exp) **[23:16]**
This read-only field provides the value of the expanded sector count register in the shadow register block.

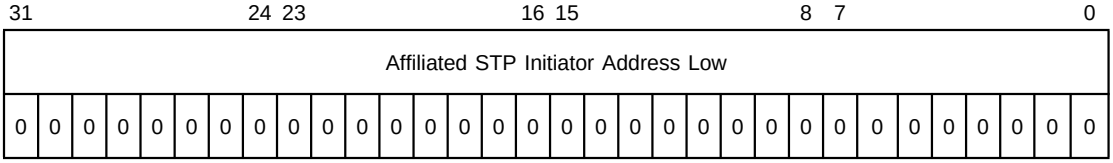
Reserved **[15:0]**

Register: 0x0098
Affiliated STP Initiator Address High
Read/Write



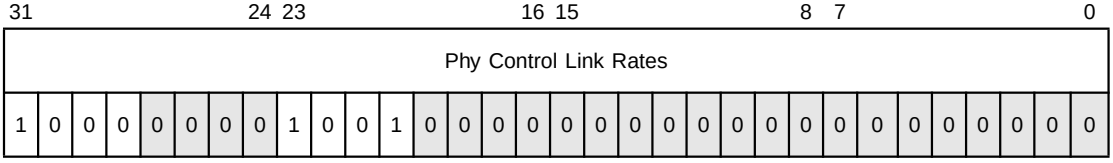
Affiliated STP Initiator Address High **[31:0]**
This read-only field provides bits [63:32] of the affiliated STP initiator address.

Register: 0x009C
Affiliated STP Initiator Address Low
Read/Write



Affiliated STP Initiator Address Low [31:0]
This read-only field provides bits [31:0] of the affiliated STP initiator address.

Register: 0x00A0
Phy Control Link Rates
Read/Write



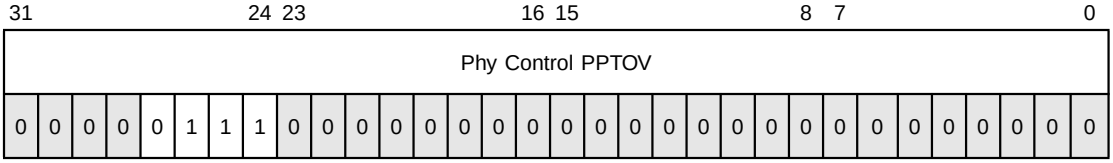
Programmed Minimum Physical Link Rate [31:28]
This field programs the minimum physical link rate. The default value for this field is 0b1000, which indicates a minimum physical link rate of 1.5 Gbits/s.

Reserved [28:24]

Programmed Maximum Physical Link Rate [23:20]
This field programs the maximum physical link rate. The default value for this field is 0b1001, which indicates a minimum physical link rate of 3.0 Gbits/s.

Reserved [19:0]

Register: 0x00A4
Phy Control PPTOV
Read/Write

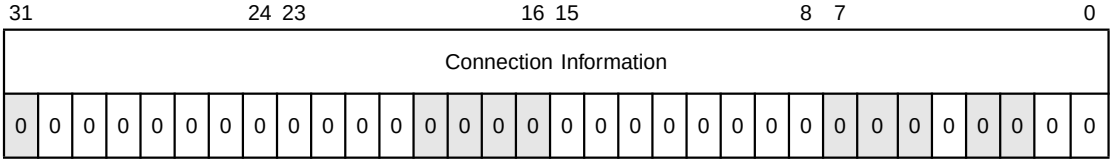


Reserved **[31:28]**

Partial Pathway Timeout Value **[27:24]**
This field programs the partial pathway timeout value (PPTOV).

Reserved **[23:0]**

Register: 0x00AC
Connection Information
Read/Write



Reserved **31**

Align Count **[30:20]**
This read-only field provides the count of dwords since last ALIGN insertion.

Reserved **[19:16]**

Function Result **[15:8]**
This read-only field indicates the phy currently supports SATA. The phy currently supports SATA if a SATA device is attached. The encoding of this field is

Encoding	Definition
0x00	Phy currently supports SATA.
0x12	Phy does not currently support SATA.

All other encodings of this field are reserved.

Reserved **[7:5]**

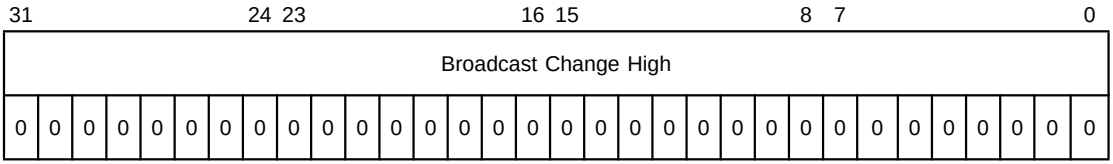
Connection Open **4**
This read-only bit indicates is set if a connection is open.

Reserved **[3:2]**

Transmitter Scramble Override **1**
Setting this bit causes the transmitter to not scramble dwords. Clearing this bit causes the transmitter to scramble dwords according the SAS/SATA rules.

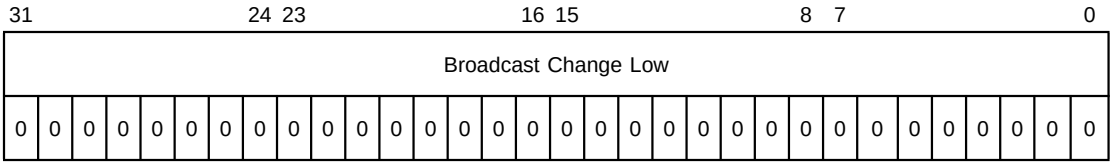
Receiver Descramble Control **0**
Setting this bit causes the receiver to not descramble dwords. Clearing this bit causes the receiver to descramble dwords according the SAS/SATA rules.

Register: 0x00C0
Broadcast Change High
Read/Write



Broadcast Change High Address **[31:0]**
This field provides upper 32 bits of the 64-bit SAS address of the most recent Broadcast Change register written.

Register: 0x00C4
Broadcast Change Low
Read/Write

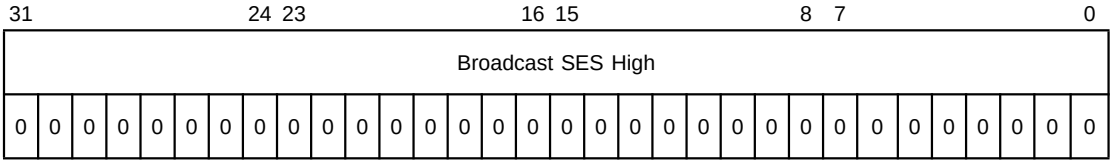


Writes to the Broadcast Change Low register cause the associated link to transmit the broadcast type. To generate a broadcast primitive, write the attached address of the phy that originates a broadcast request to

the Broadcast Change registers. If the attached address of a link matches the address written the Broadcast Change registers, no broadcast primitive is transmitted on that link.

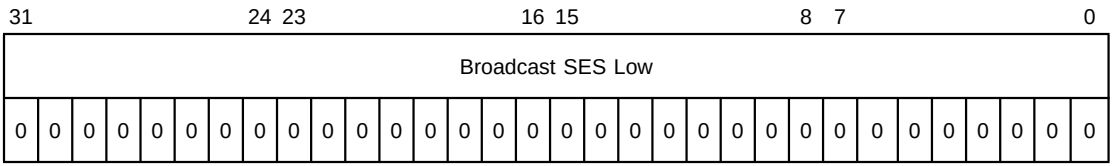
Broadcast Change Low Address **[31:0]**
This field provides lower 32 bits of the 64-bit SAS address of the most recent Broadcast Change register written.

Register: 0x00C8
Broadcast SES High
Read/Write



Broadcast SES High Address **[31:0]**
This field provides upper 32 bits of the 64-bit SAS address of the most recent Broadcast SES register written.

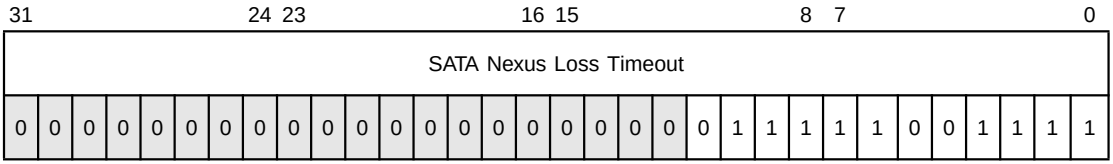
Register: 0x00CC
Broadcast SES Low
Read/Write



Writes to the Broadcast SES Low register cause the associated link to transmit the broadcast type. To generate a broadcast primitive, write the attached address of the phy that originates a broadcast request to the Broadcast SES registers. If the attached address of a link matches the address written the Broadcast SES registers, no broadcast primitive is transmitted on that link.

Broadcast SES Low **[31:0]**
This field provides lower 32 bits of the 64-bit SAS address of the most recent Broadcast SES register written.

Register: 0x0108
SATA Nexus Loss Timeout
Read/Write



Reserved **[31:12]**

Nexus Loss Timeout Value (ms) **[11:0]**
This field sets the nexus loss timeout value in milliseconds. When the nexus loss timer reaches the value in this field, and when the next reject or OpenTimeout occurs, the STP/SATA bridge requests a new reset sequence. The default value for this field is 0x7CF, which provides a 2-second Nexus Loss Timeout value. Possible encodings of this field are:

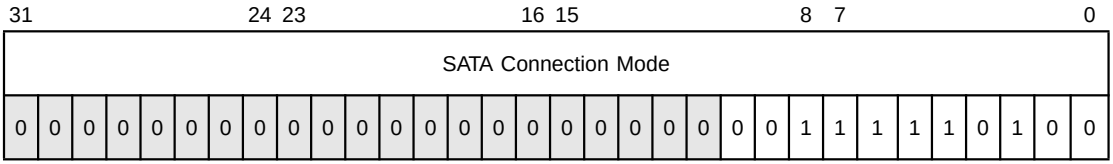
Encoding	Definition
0x000	No Retries Permitted
0x001–0xFFE	Permits retries until the Nexus Loss Timer reaches this value.
0xFFF	Disables Nexus Loss Timeouts

The Nexus Loss Timer starts counting when the STP/SATA bridge receives one of the following OPEN_REJECTS:

- Pathway Blocked
- Stop_0
- Stop_1
- No Destination

It also starts counting in the event of an OpenTimeout. Receipt of an OPEN_REJECT(RETRY/CONTINUE_/CONTINUE_1) frame resets and disables the nexus timer.

Register: 0x0120
SATA Connection Mode
Read/Write



Reserved **[31:11]**

FIS Close **10**
Setting this bit causes SATA connections to close after every FIS. Clearing bit causes SATA connections to not close after every instruction.

SATA Close Timeout **[9:0]**
This register provides the timeout value in milliseconds for closing a dormant SATA connection. The default SATA close timeout value is 500 ms.

4.4.2 Phy Registers

Table 4.7 provides the SAS phy register map. Detailed register descriptions follow the register map.

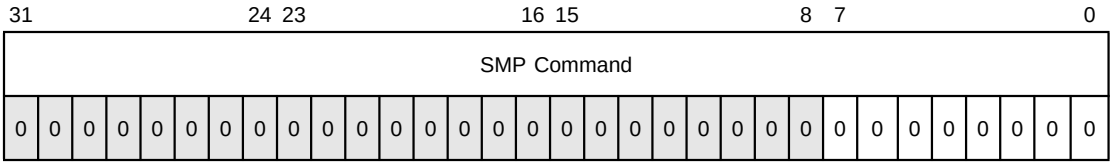
Table 4.7 SAS Phy Register Map

Offset	Register Name
0x00–0x0C	Reserved
0x10	SMP Command
0x14	Reserved
0x18	Error Log Invalid Word
0x1C	Error Log Display Error
0x20	Error Log Loss of Sync
0x24	Error Log Reset Sequence Fail

Table 4.7 SAS Phy Register Map (Cont.)

Offset	Register Name
0x28	Test Register
0x2C–0x30	Reserved
0x34	Custom Jitter
0x38–0xFF	Reserved

Register: 0x0010
SMP Command
Read/Write



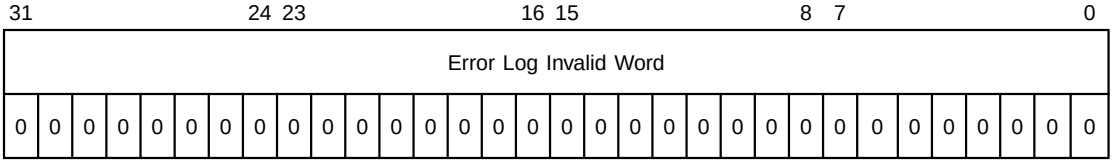
Reserved [31:8]

PhySMPOp [7:0]
Writing to this field initiates commands as outlined in Table 4.8. The LSISASx12 clears this bit field when the respective command completes. If the current command does not complete before writing a new command to this field, then the old command aborts and the new command initiates.

Table 4.8 Phy Operations Encodings

Phy Op	Operation
0x00	No Operation.
0x01	Perform a phy reset sequence.
0x02	Perform a phy reset sequence and send a HARD_RESET primitive.
0x03	Power down the GigaBlaze transceivers and reset the phy.
0x04	Reserved.
0x05	Clear all error log registers.
0x06	Clear active affiliation.
0x07	Send port selector sequence
0x08–0xFF	Reserved.

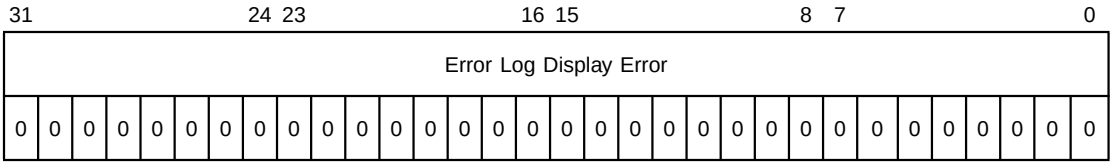
Register: 0x0018
Error Log Invalid Word
Read/Write



Error Log Invalid Word **[31:0]**
This register indicates the number of invalid words received. This register increments when an invalid word is received while the PhyReady signal is asserted.

This register no longer increments after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the PhySMPOp field of the SMP Command register clears this register.

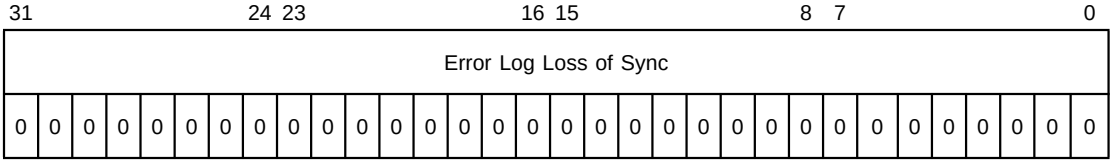
Register: 0x001C
Error Log Display Error
Read/Write



Error Log Disparity Error **[31:0]**
This register indicates the number of disparity error occurrences. This register increments when a word tagged with a disparity error is received while the PhyReady signal is asserted.

This register no longer increments after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the PhySMPOp field of the SMP Command register clears this register.

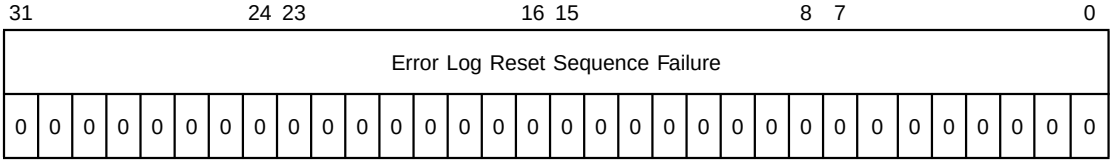
Register: 0x0020
Error Log Loss of Sync
Read/Write



Error Log Loss of Synchronization **[31:0]**
This register indicates the number of loss-of-synchronization occurrences. This register increments when a loss of synchronization condition is reported outside of a reset sequence and following word synchronization.

This register no longer increments after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the PhySMPOp field of the SMP Command register clears this register.

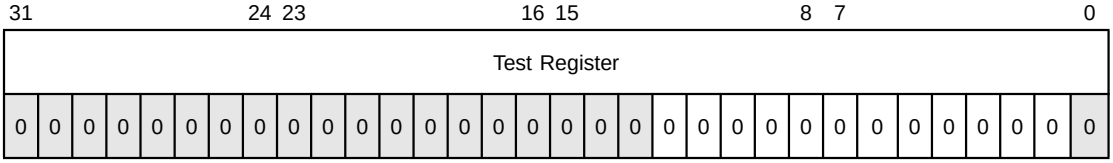
Register: 0x0024
Error Log Reset Sequence Fail
Read/Write



Error Log Reset Sequence Failure **[31:0]**
This register indicates the number of reset sequences failures. This register increments every time a reset sequence does not complete successfully. A reset sequence fails when speed negotiation fails.

This register no longer increments after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the PhySMPOp field of the SMP Command register clears this register.

Register: 0x0028
Test Register
Read/Write



This register provides test control signals. These controls are not used during normal operation.

Reserved **[31:13]**

RX Jitter Enable **12**
Setting this bit enables receive jitter pattern checking.

TX Jitter Enable **11**
Setting this bit enables transmit jitter pattern generation.

JPAT Select **[10:8]**
This field specifies the jitter pattern for transmit generation and receive checking. Table 4.9 provides the encoding for this field.

Table 4.9 Jitter Pattern Selection

Select	Pattern	Description
0,7	ALIGN	Transmit 6 ALIGN0 frames for synchronization before the jitter test
1	CJTPAT	Jitter patter from SAS Annex A
2	LTDP	SATA Low Transaction Density Pattern
3	HRQR	SATA Half-rate/Quarter-rate High Transaction Density Pattern
4	LFSC	SATA Low Frequency Spectral Content Pattern
5	SSOP	SATA Simultaneous Switching Outputs Pattern
6	COMP	SATA Composite of patterns 2–5

Force Data Rate **[7:4]**
When set, this field overrides the minimum and maximum data rate settings for the phy layer. The data rate selected in this field applies to both the receivers and transmitters. The encodings of this field are

Encoding	Definition
0b1000	1.5 Gbits/s
0b1001	3.0 Gbits/s

All others encodings of this field are reserved.

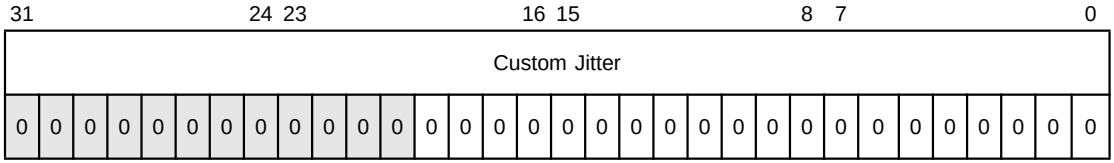
Force SATA **3**
Setting this bit causes the LSISASx12 to report as a SATA device. Use this bit in conjunction with the Bypass Reset Sequence bit.

Select Starting Disparity **2**
This bit selects the starting disparity to be transmitted in the encoder when the LSISASx12 transitions from IDLE. Clearing this bit indicates negative disparity. Setting this bit indicates positive disparity.

Bypass Reset Sequence **1**
Setting this bit causes the LSISASx12 to bypass the normal OOB reset sequence, and to use the rate set in the Force Data Rate field as the negotiated rate.

Reserved **0**

Register: 0x0034
Custom Jitter
Read/Write



This register provides status signals to observe during testing. It is not for use during normal operation.

Reserved **[31:20]**

JitterReg **[19:0]**
When set to pattern 7, the jitter engine uses this value. Use this field to generate a “clock” pattern of various frequencies to assist in jitter measurements. The upper and lower halves of this register are flipped prior to serialization, i.e. TxDataOut = {JitterReg[9:0],JitterReg[19:10]}.

4.5 STP Target Registers

Table 4.10 provides the register map for the STP target portion of the EMB. Detailed bit descriptions follow the register map.

Table 4.10 STP Target Register Map

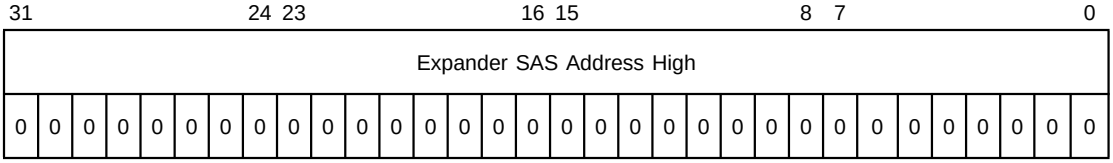
Offset	Register Name
0x000–0x00C	Reserved
0x010	Expander SAS Address High
0x014	Expander SAS Address Low
0x018–0x40	Reserved
0x044	Discover Information
0x048	Discover Information 2 – SAS Address High
0x04C	Discover Information 3 – SAS Address Low
0x050	Discover Information 4 – Attached SAS Address High
0x054	Discover Information 5 – Attached SAS Address Low
0x058	Discover Information 6 – Attached Phy Identifier
0x05C	Discover Information 7 – Reserved
0x060	Discover Information 8 – Link Rates
0x064	Discover Information 9 – Routing Attributes
0x068	Discover Information 10 – Vendor Specific
0x06C	Reserved
0x070	Phy SATA Phy ID
0x074	Reserved
0x078	STP Target SAS Address High
0x07C	STP Target SAS Address Low
0x080	SATA Dev2Host Reset FIS0
0x084	SATA Dev2Host Reset FIS1
0x088	SATA Dev2Host Reset FIS2

Table 4.10 STP Target Register Map (Cont.)

Offset	Register Name
0x08C	SATA Dev2Host Reset FIS3
0x090–0x94	Reserved
0x098	Affiliated STP Initiator Address High
0x09C	Affiliated STP Initiator Address Low
0x0A0	Phy Control Link Rates
0x0A4	Phy Control PPTOV
0x0A8	Reserved
0x0AC	Connection Information
0x0B0–0x107	Reserved
0x108	SATA Nexus Loss Timeout
0x10C–0x1FC	Reserved
0x200	Last H2D Register FIS Received Dword 0
0x204	Last H2D Register FIS Received Dword 1
0x208	Last H2D Register FIS Received Dword 2
0x20C	Last H2D Register FIS Received Dword 3
0x210–0x21C	Reserved
0x220	STP Connection Control
0x224–0x30C	Reserved
0x310	SMP Phy Operation
0x314	Reserved
0x318	Error Log – Invalid DWord
0x31C	Error Log – I^2C CRC Error Count
0x320	Error Log – Unexpected SYNC Count
0x324	Error Log – R_ERR Transmitted/Received Count
0x328	Reserved
0x32C	STP Error Status
0x330–0x3FF	Reserved

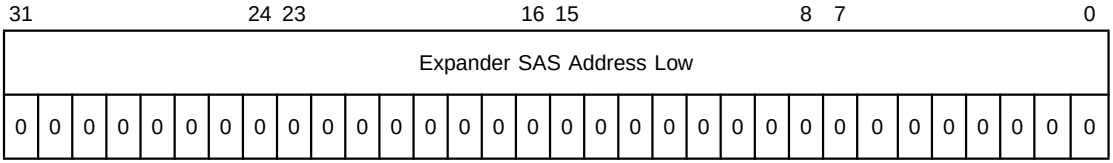
The STP Target registers base address is 0x43000. The STP Target registers can also be written via a broadcast write, using the Broadcast Base Address of 0x18000. However, only use this mechanism to write to the SATA Nexus Loss Timeout registers.

Register: 0x0010
Expander SAS Address High
Read/Write



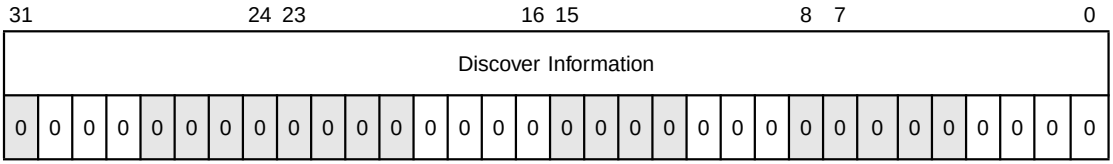
Expander SAS Address High **[31:0]**
This field is written with the same value as Configuration Manager LSISASx12 Expander SAS Address High register.

Register: 0x0014
Expander SAS Address Low
Read/Write



Expander SAS Address Low **[31:0]**
This field is written with the same value as Configuration Manager LSISASx12 Expander SAS Address High register.

Register: 0x0044
Discover Information
Read/Write



Reserved **31**

Attached Device Type **[31:28]**
This read-only field provides the attached device type. The encoding of this field is

Encoding	Definition
0b000	No Device Attached
0b001	End Device Only
0b010	Edge Expander Device
0b011	Fanout Expander Device

All other encodings are reserved.

Reserved **[27:20]**

Negotiated Link Rate **[19:16]**
This read-only field provides the negotiated link rate. The encoding of this field is

Encoding	Definition
0b0001	Phy disabled, if the STP Target is disabled through the SMP.
0b1001	Phy enabled at 3.0 Gbits/s, if the STP Target is not disabled through SMP.

All other encodings of this field are reserved.

Reserved **[15:12]**

Attached SSP Initiator **11**
When set, this read-only bit indicates that the attached device is an SSP initiator.

Attached STP Initiator **10**
When set, this read-only bit indicates that the attached device is an STP initiator.

Attached SMP Initiator **9**
When set, this read-only bit indicates that the attached device is an SMP initiator.

Reserved **[8:5]**

Attached SATA Port Selector **4**
When set, this read-only bit indicates that the attached device is a SATA port selector.

Attached SSP Target **3**
When set, this read-only bit indicates that the attached device is an SSP target.

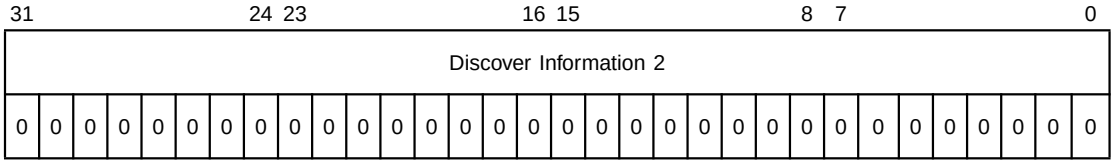
Attached STP Target **2**
When set, this read-only bit indicates that the attached device is an STP target.

Attached SMP Target **1**
When set, this read-only bit indicates that the attached device is an SMP target.

Attached SATA Target **0**
When set, this read-only bit indicates that the attached device is a SATA target.

Register: 0x0048

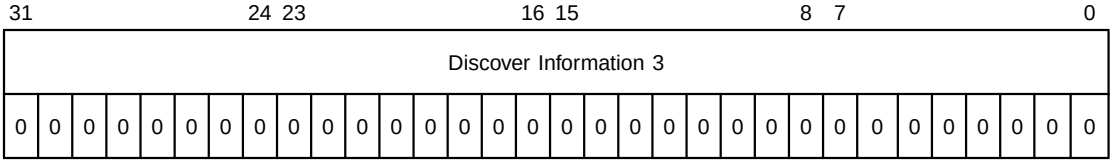
Discover Information 2 – SAS Address High
Read/Write



SAS Address High **[31:0]**
This read-only field provides bits [63:32] of the SAS address.

Register: 0x004C

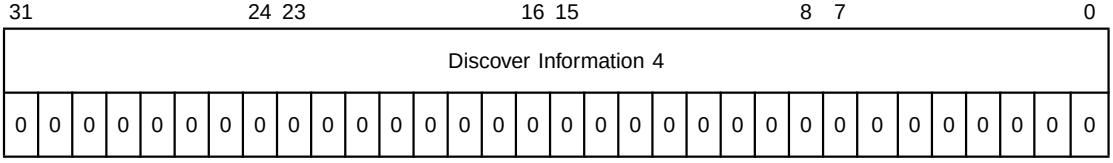
Discover Information 3 – SAS Address Low
Read/Write



SAS Address Low **[31:0]**
This read-only field provides bits [31:0] of the SAS address.

Register: 0x0050

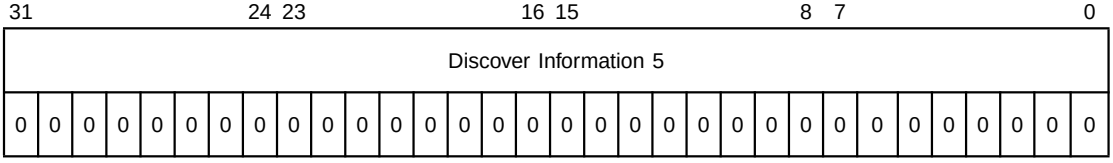
Discover Information 4 – Attached SAS Address High
Read/Write



Attached SAS Address High **[31:0]**
This read-only field provides bits [63:32] of the attached SAS address.

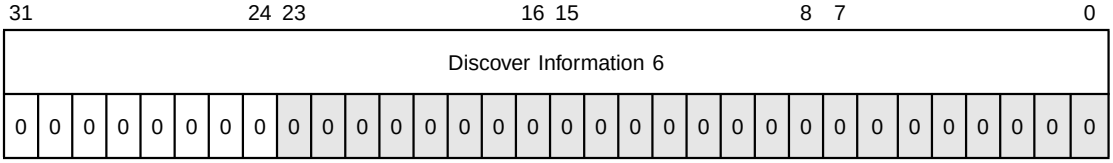
Register: 0x0054

Discover Information 5 – Attached SAS Address Low
Read/Write



Attached SAS Address Low **[31:0]**
This read-only field provides bits [31:0] of the attached SAS address.

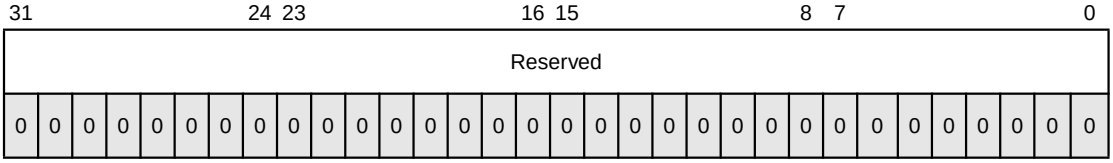
Register: 0x0058
Discover Information 6 – Attached Phy Identifier
Read/Write



Attached Phy Identifier **[31:24]**
This read-only field provides the attached phy identifier.

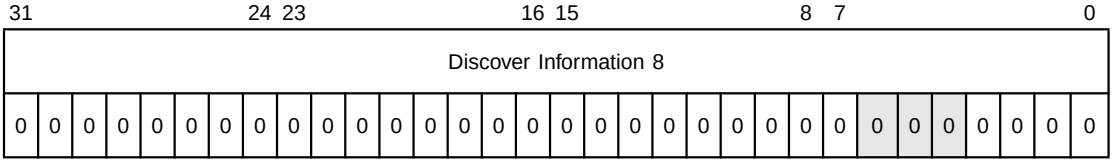
Reserved **[23:0]**

Register: 0x005C
Discover Information 7 – Reserved
Read/Write



Reserved **[31:0]**

Register: 0x0060
Discover Information 8 – Link Rates
Read/Write



Programmed Minimum Physical Link Rate **[31:28]**
This field is read-only.

Hardware Minimum Physical Link Rate **[27:24]**
This field is read-only.

Programmed Maximum Physical Link Rate **[23:20]**
This field is read-only.

Hardware Maximum Physical Link Rate **[19:16]**
This field is read-only.

Phy Change Count **[15:8]**
This read-only field indicates the number of times that the STP Target has requested a BROADCAST(CHANGE) to the broadcast processor. This counter wraps to 0x00 when it reaches the maximum value of 0xFF.

Virtual Phy Bit **7**
When set, this read-only bit indicates that the phy is an internal port.

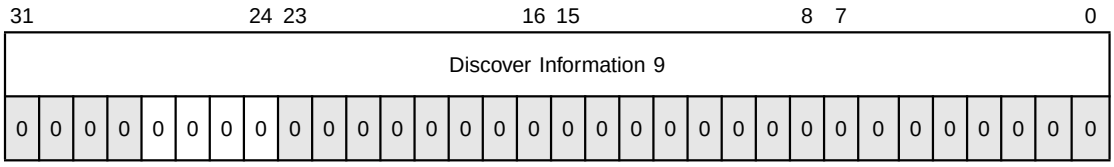
Reserved **[6:4]**

PPTOV **[3:0]**
This field is read-only.

Register: 0x0064

Discover Information 9 – Routing Attributes

Read/Write



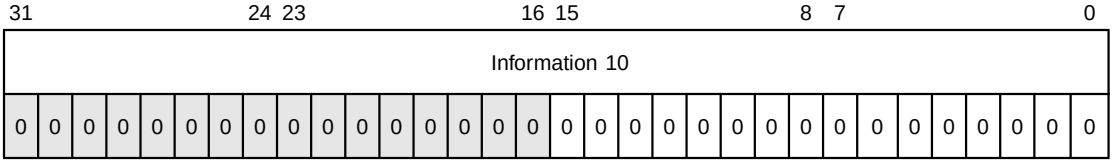
Reserved **[31:28]**

Routing Attribute **[27:24]**
Clearing this field to 0b0000 indicates direct routing.

Reserved **[23:0]**

Register: 0x0068

Discover Information 10 – Vendor Specific
Read/Write

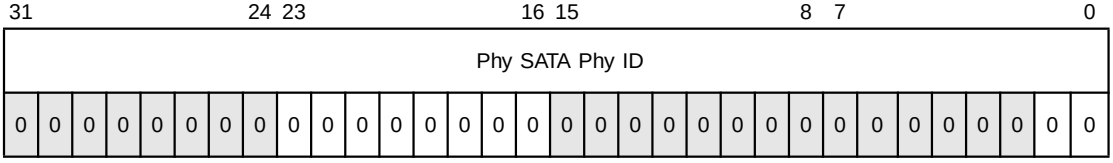


Reserved **[31:16]**

Vendor Specific **[15:0]**

Register: 0x0070

Phy SATA Phy ID
Read/Write



Reserved **[31:24]**

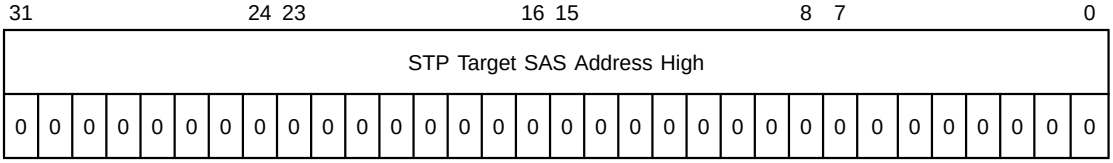
Phy Identifier **[23:16]**
This read-only field provides the phy identifier.

Reserved **[15:2]**

Affiliations Supported **1**
This bit is read-only.

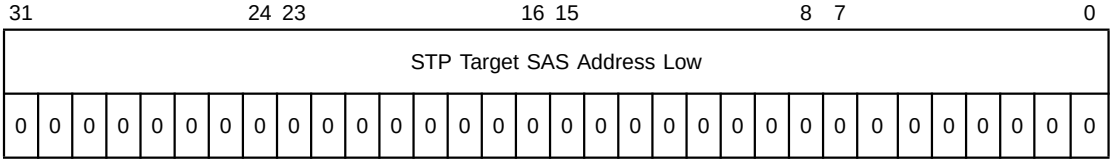
Affiliation Valid **0**
This bit is read-only.

Register: 0x0078
STP Target SAS Address High
Read/Write



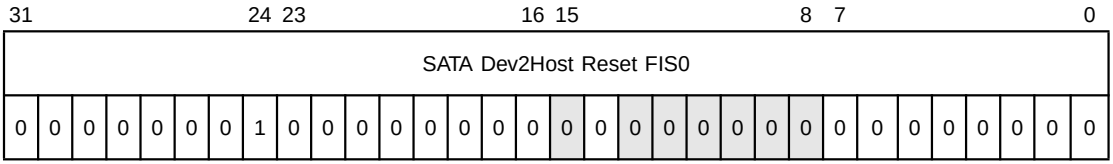
STP Target SAS Address High **[31:0]**
This read-only field provides bits [63:32] of the STP target SAS address.

Register: 0x007C
STP Target SAS Address Low
Read/Write



STP Target SAS Address Low **[31:0]**
This read-only field provides bits [31:0] of the STP target SAS address.

Register: 0x0080
SATA Dev2Host Reset FIS0
Read/Write



During reset, SATA devices transmit an unsolicited D2H Register FIS to an attached SATA host device to provide a signature value contained within the Sector Count, Sector Number, Cylinder Low, and Cylinder High registers. If the signature value in these registers is set to 0x010114EB, the host recognizes that the device supports the ATAPI PACKET Feature

Set. Otherwise, the host assumes the device does not support the ATAPI PACKET feature set.

This register contains the reset signature for an STP host to read in order for it to determine that the STP Target is a PACKET device. STP hosts have read access to the contents of these registers through the SMP REPORT PHY SATA function.

Error **[31:24]**
This field is set to 0x01 if the device passes boot diagnostics.

Status **[23:16]**
This field is set to 0x00 if there is not an error.

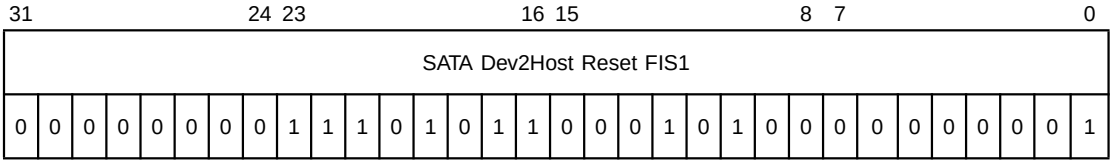
Reserved **15**

Interrupt **14**
This bit reflects the status of the interrupt bit line.

Reserved **[13:8]**

FIS Type **[7:0]**
This field provides the FIS type.

Register: 0x0084
SATA Dev2Host Reset FIS1
Read/Write



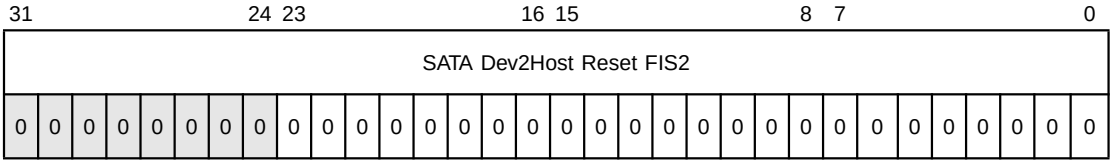
Dev/Head **[31:24]**
This read-only field provides the values of the device register and head register.

Cyl High **[23:16]**
This read-only field provides signature byte 3. This field is set to 0xEB, indicating that this is a PACKET device.

Cyl Low **[15:8]**
This read-only field provides signature byte 2. This field is set to 0x14, indicating that this is a PACKET device.

Sector Number **[7:0]**
This read-only field provides signature byte 1. This field is set to 0x01, indicating that this is a PACKET device.

Register: 0x0088
SATA Dev2Host Reset FIS2
Read/Write



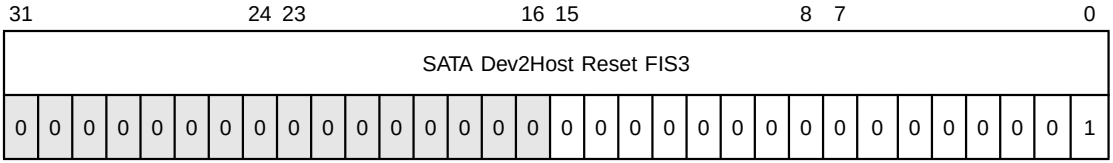
Reserved **[31:24]**

Cyl High (exp) **[23:0]**

Cyl Low (exp) **[15:8]**

Sector Number (exp) **[7:0]**

Register: 0x008C
SATA Dev2Host Reset FIS3
Read/Write

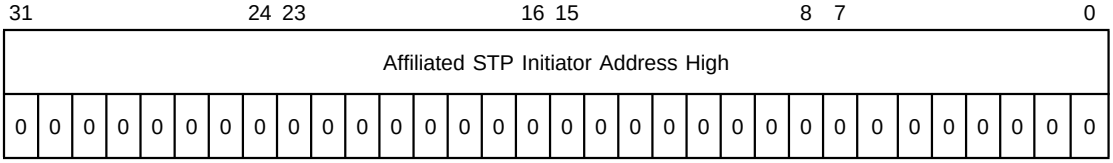


Reserved **[31:16]**

Sector Count (exp) **[15:8]**

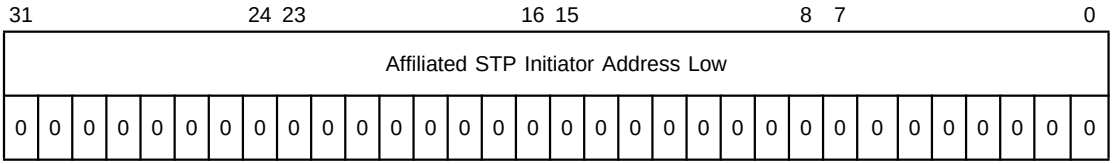
Sector Count **[7:0]**
This read-only field provides signature byte 0. It is set to 0x01, indicating that this is a PACKET device.

Register: 0x0098
Affiliated STP Initiator Address High
Read/Write



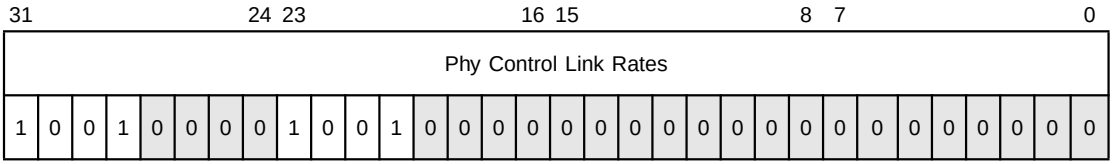
Affiliated STP Initiator Address High [31:0]
This read-only field provides bits [63:32] of the affiliated STP initiator address.

Register: 0x009C
Affiliated STP Initiator Address Low
Read/Write



Affiliated STP Initiator Address Low [31:0]
This read-only field provides bits [31:0] of the affiliated STP initiator address.

Register: 0x00A0
Phy Control Link Rates
Read/Write



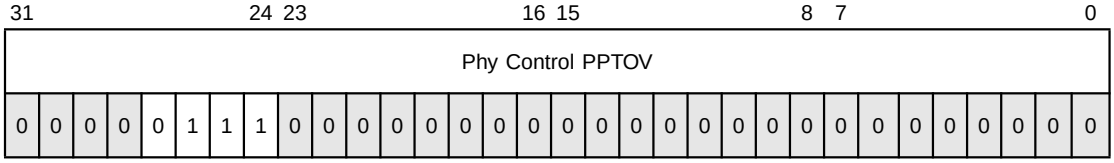
Programmed Minimum Physical Link Rate [31:28]
This read-only field programs the minimum physical link rate. The default value for this field is 0b1001, which indicates a minimum physical link rate of 3.0 Gbits/s.

Reserved **[28:24]**

Programmed Maximum Physical Link Rate **[23:20]**
This read-only field programs the maximum physical link rate. The default value for this field is 0b1001, which indicates a maximum physical link rate of 3.0 Gbits/s.

Reserved **[19:0]**

Register: 0x00A4
Phy Control PPTOV
Read/Write

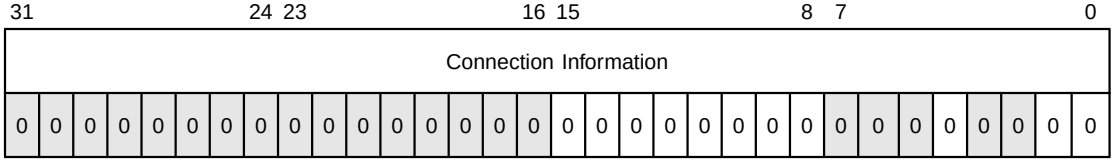


Reserved **[31:28]**

Partial Pathway Timeout Value **[27:24]**
This field programs the partial pathway timeout value (PPTOV).

Reserved **[23:0]**

Register: 0x00AC
Connection Information
Read/Write



Reserved **[31:16]**

Function Result **[15:8]**
This read-only field indicates that the phy supports SATA. The encoding of this field is

Encoding	Definition
0x00	Phy supports SATA.

All other encoding of this field are reserved.

Reserved **[7:5]**

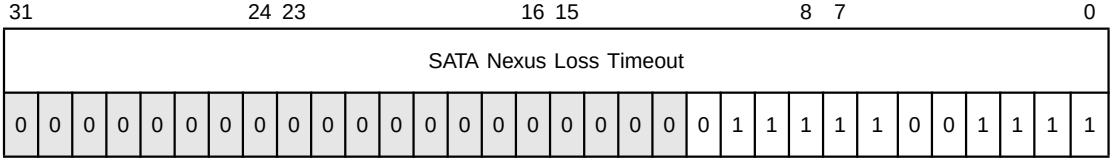
Connection Open **4**
This read-only bit is set if a connection is open.

Reserved **[3:2]**

Transmit Scramble Override **1**
Setting this bit causes the transmitter to not scramble dwords. Clearing this bit causes the transmitter to scramble dwords according the SAS/SATA rules.

Receive Descramble Control **0**
Setting this bit causes the receiver to not descramble dwords. Clearing this bit causes the receiver to descramble dwords according the SAS/SATA rules.

Register: 0x0108
SATA Nexus Loss Timeout
Read/Write



Reserved **[31:12]**

Nexus Loss Timeout Value (ms) [11:0]
This field sets the nexus loss timeout value in milliseconds. When the nexus loss timer reaches the value in this field and when the next reject or OpenTimeout occurs, the STP/SATA bridge requests a new reset sequence. The default value for this field is 0x7CF, which provides a 2-second Nexus Loss Timeout. The encoding of this field is:

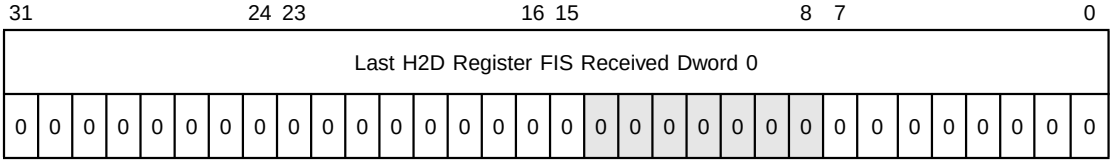
Encoding	Definition
0x000	No Retries Permitted
0x001–0xFFE	Permits retries until the Nexus Loss Timer reaches this value.
0xFFF	Disables Nexus Loss Timeouts

The Nexus Loss Timer starts counting when the STP/SATA bridge receives one of the following OPEN_REJECTS:

- Pathway Blocked
- Stop_0
- Stop_1
- No Destination
- Initialize_0
- Initialize_1

It also starts counting in the event of an OpenTimeout. Receipt of an OPEN_REJECT(RETRY/CONTINUE_/CONTINUE_1) frame resets and disables the nexus timer.

Register: 0x0200
Last H2D Register FIS Received Dword 0
Read/Write



This register contains the first dword of a non-Data FIS that is received from an STP initiator. The fields displayed here are for FIS Type field 0x27.

Features **[31:24]**
This read-only field displays the ATAPI Features register.

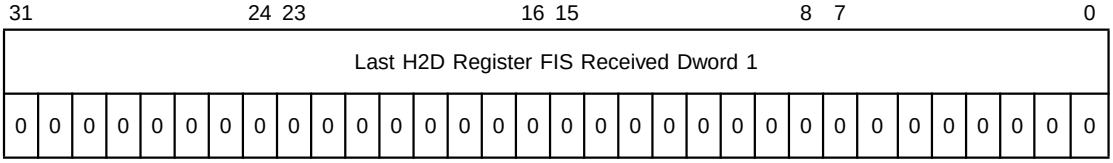
Command **[23:16]**
This read-only field displays the ATAPI command.

C **15**
If this read-only bit is cleared, an update to the ATAPI command register has been done.

Reserved **[14:8]**

FIS Type **[7:0]**
In this example, this field is set to 0x27, indicating a Host-to-Device Register FIS.

Register: 0x0204
Last H2D Register FIS Received Dword 1
Read/Write

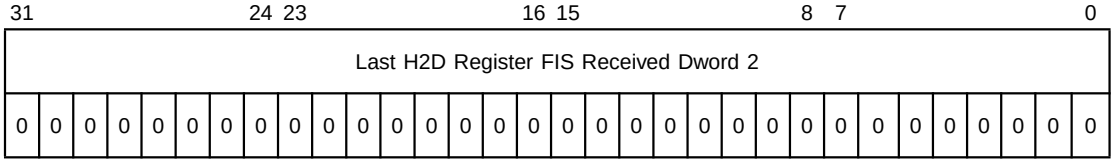


This register contains the contents of the second dword of a non-Data FIS that was received from an STP Initiator. The fields displayed here are for FIS Type field 0x27.

Dev/Head	**[31:24]**
Cyl High	**[23:16]**
Cyl Low	**[15:8]**
Sector Number	**[7:0]**

Register: 0x0208

Last H2D Register FIS Received Dword 2
Read/Write

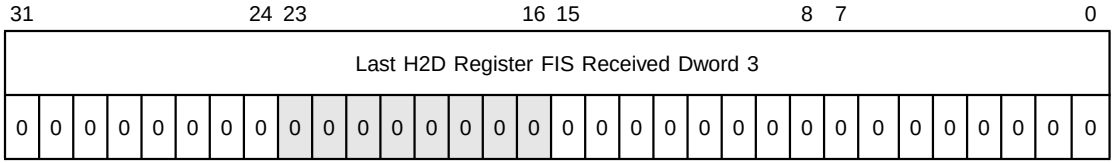


This register contains the contents of the third dword of a non-Data FIS that was received from an STP Initiator. The fields displayed here are for FIS Type field 0x27.

Features (exp)	**[31:24]**
Cyl High (exp)	**[23:16]**
Cyl Low (exp)	**[15:8]**
Sector Number (exp)	**[7:0]**

Register: 0x020C

Last H2D Register FIS Received Dword 3
Read/Write



This register contains the contents of the fourth dword of a non-Data FIS that was received from an STP Initiator. The fields displayed here are for FIS Type field 0x27.

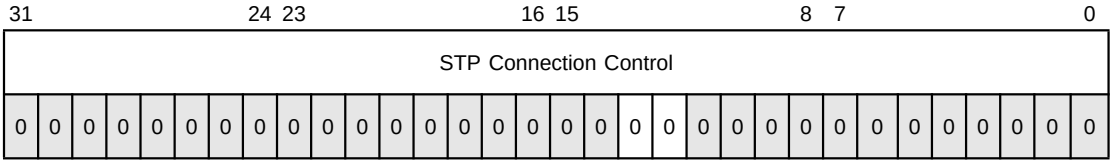
Control (SRST = bit 26)	**[31:24]**

Reserved **[23:16]**

Sector Count (exp) **[15:8]**

Sector Count **[7:0]**

Register: 0x0220
STP Connection Control
Read/Write



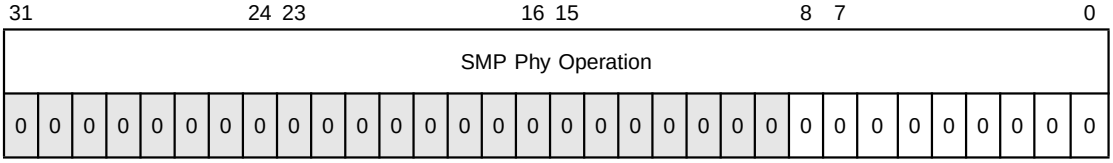
Reserved **[31:14]**

Force Bad CRC **[13:12]**
This field is for verifying the reaction of the STP Target to receiving R_ERR. The encoding of this field is

Encoding	Definition
0b00	Disable bad CRC generation.
0b01	Transmit one bad CRC for all FIS types.
0b10	Transmit bad CRC for all D2H FIS types, all the time.
0b11	Transmit bad CRC for PIO setup FIS types, all the time.

Reserved **[11:0]**

Register: 0x0310
SMP Phy Operation
Read/Write



Reserved **[31:9]**

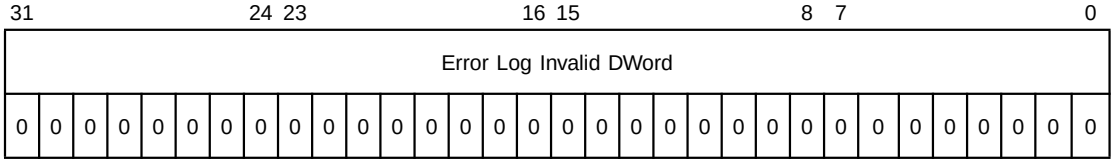
Disable STP **8**
When set, this read-only bit indicates that a SMP PHY CONTROL(DISABLE) command has disabled the STP Target. When cleared, this bit indicates the STP Target is not disabled.

SmpPhyOp **[7:0]**
This bit field initiates the commands in Table 4.11. The LSISASx12 clears this field one clock after the associated control signal asserts.

Table 4.11 SmpPhyOp Commands

Phy Op	Operation
0x00	No Operation.
0x01	This command clears the Disable STP bit and resets the LSISASx12 STP target.
0x02	This command clears the Affiliation Valid and Disable STP bits, and resets the LSISASx12 STP target.
0x03	This command causes any Inbound Open Address Frames targeting the STP Target to receive OPEN_REJECT (NO DESTINATION). It also prevents transmission of any Outbound Open Address Frames.
0x04	Reserved.
0x05	This command clears all the STP Error Status register and all of the error count registers.
0x06	This command clears the Affiliation Valid bit.
0x07	Activate Port Selector. No operation is performed.
0x08–0xFF	Reserved.

Register: 0x0318
Error Log – Invalid DWord
Read Only

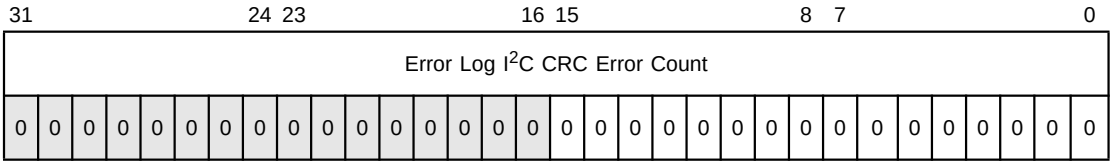


Error Log Invalid Dword Count **[31:0]**
This read-only field indicates the number of invalid dwords received by the STP Target. The value increments every time the STP Target receives an invalid

dword between a pair of SATA_SOF and SATA_EOF primitives.

This count does not increment when an ERROR primitive is detected because a different phy in the expander chain increments its Invalid Dword count. This register does not increment after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the SmpPhyOp field in the SMP Phy Operation register clears this register. Bytes 12–15 of the SMP REPORT PHY ERROR LOG Response Frame report the contents of this register.

Register: 0x031C
Error Log – I²C CRC Error Count
Read Only



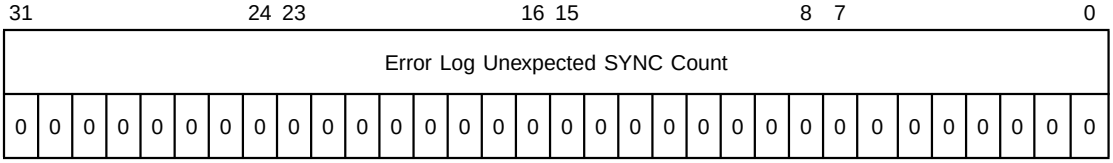
Reserved **[31:16]**

Error Log I²C Error Count **[15:0]**
This read-only field indicates the number of times the SEMB detects a CRC error in write data that is transmitted by the SEP I²C master, or when the SEMB determines that the SEP has detected a CRC error during a read operation.

This register does not increment after reaching the maximum value of 0xFFFF. Writing 0x05 to the SmpPhyOp field in the SMP Phy Operation register clears this register. Bytes 16–19 of the SMP REPORT PHY ERROR LOG Response Frame report the contents of this register.

Register: 0x0320

Error Log – Unexpected SYNC Count
Read Only

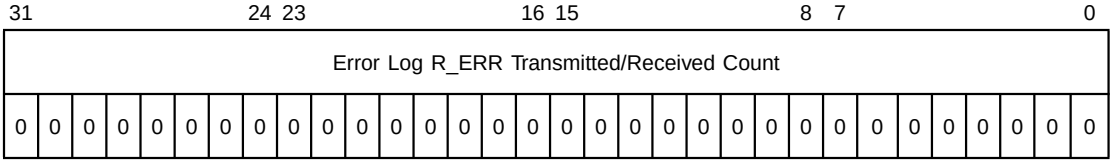


Error Log Unexpected SATA Sync Count [31:0]
This read-only field indicates the number of disparity error occurrences. This field increments when a SATA_SYNC primitive is detected while the STP Target is expecting a different primitive.

This register does not increment after reaching the maximum value of 0xFFFFFFFF. Writing 0x05 to the SmpPhyOp field in the SMP Phy Operation register clears this register. Bytes 20–23 of the SMP REPORT PHY ERROR LOG Response Frame report the contents of this register.

Register: 0x0324

Error Log – R_ERR Transmitted/Received Count
Read Only



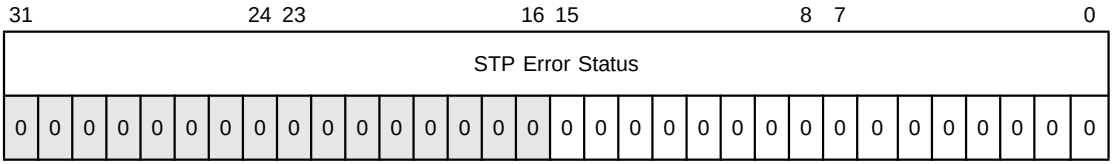
R_ERRs Transmitted Count 31:16
This read-only field indicates the number of times the STP Target has asserted R_ERR. The value increments when the STP Target transmits an R_ERR primitive.

This register does not increment after reaching the maximum value of 0xFFFF. Writing 0x05 to the SmpPhyOp field in the SMP Phy Operation register clears this register. Bytes 24–25 of the SMP REPORT PHY ERROR LOG Response Frame report the contents of this register.

R_ERRs Received Count **15:0**
This read-only field indicates the number of times the STP Target has received an R_ERR in response to a transmitted FIS. The value increments when the STP Target detects an R_ERR primitive.

This register does not increment after reaching the maximum value of 0xFFFF. Writing 0x05 to the SmpPhyOp field in the SMP Phy Operation register clears this register. Bytes 26–27 of the SMP REPORT PHY ERROR LOG Response Frame report the contents of this register.

Register: 0x032C
STP Error Status
Read/Write



All bits in this register are persistent, and remain set until an SMP PHY CONTROL (CLEAR ERROR LOG), (HARD RESET), (LINK RESET) function is issued to Virtual Phy 12.

Reserved **[31:16]**

Unrecovered I^2C CRC Error **15**
The LSISASx12 sets this read-only bit when an I^2C CRC error occurs, the SEP I^2C master has reached its retry limit, and the I^2C master has aborted the command.

I^2C CRC Error **14**
The LSISASx12 sets this read-only bit when an I^2C CRC error occurs. The SEP I^2C master may retry the transfer.

Unrecovered Data FIS **13**
The LSISASx12 sets this read-only bit when the STP Target receives an R_ERR primitive in response to a Data FIS transmission attempt. If this occurs, the STP Target transmits a CLOSE sequence to close the connection if the command was not a PACKET command. If the failed Data FIS is part of a PACKET command sequence, the STP Target transmits a D2H

Register FIS with Status[BSY]=0 and Error[ABRT] = 1 prior to transmitting a CLOSE sequence.

Unrecovered D2H Reg FIS 12
The LSISASx12 sets this read-only bit when the STP Target receives an R_ERR in response to a second attempt to transmit a D2H Register FIS. If this occurs, the STP Target transmits a CLOSE sequence to close the connection.

Unrecovered PIO Setup FIS 11
The LSISASx12 sets this read-only bit when the STP Target receives an R_ERR in response to a second attempt to transmit a PIO Setup FIS. If this occurs, the STP Target transmits a CLOSE sequence to close the connection.

Unsupported FIS Received 10
The LSISASx12 sets this read-only bit when the STP Target receives one of the following FIS Types:

- D2H Register FIS
- DMA Activate FIS
- DMA Setup FIS
- PIO Setup FIS
- Set Device Bits FIS
- BIST Activate FIS

If the STP Target receives one of these FIS types, the STP Target transmits an R_ERR, followed by a CLOSE sequence to close the connection.

Payload/Transfer-Count Mismatch 9
The LSISASx12 sets this read-only bit when the number of dwords received in a Host-to-Device DATA FIS does not match the Transfer Count that was placed in the previously sent PIO Setup FIS. If this occurs, the Application Layer notifies the Link Layer to send an R_ERR, and a D2H Register FIS is transmitted to the host with a SCSI Sense Key of ILLEGAL REQUEST (0x5) placed in the Error register.

CRC Error Detected **8**
The LSISASx12 sets this read-only bit when it detects a bad CRC dword in a received FIS. The STP Target transmits an R_ERR and leaves the connection open. The Initiator decides if to retransmit the failed FIS.

PMREQ Received **7**
The LSISASx12 sets this read-only bit when it receives either a PMREQ_P or PMREQ_S primitive. If this occurs, the STP Target transmits PMNAK, and then transmits a CLOSE sequence to close the connection.

R_RDY Timeout **6**
The LSISASx12 sets this read-only bit if the STP Target does not receive an R_RDY primitive within 1 ms after transmitting X_RDY primitives. If this occurs, the STP Target transmits a CLOSE sequence to close the connection.

R_OK/R_ERR Timeout **5**
The LSISASx12 sets this read-only bit if the STP Target does not receive either an R_OK or R_ERR primitive within 1 ms after transmitting an EOF/WTRM sequence. If this occurs, the STP Target transmits a CLOSE sequence to close the connection.

Unexpected SYNC **4**
The LSISASx12 sets this read-only bit when the Link Layer receives an unexpected SATA_SYNC primitive. The Error Log – Unexpected SYNC Count provides a cumulative count of received unexpected SYNC primitives.

R_ERR Received **3**
The LSISASx12 sets this read-only bit when it receives an R_ERR in response to a DATA FIS, PIO Setup FIS, or D2H Register FIS. The Error Log – R_ERR Transmitted/Received Count provides a cumulative count of R_ERR primitives that have been sent or received.

R_ERR Transmitted **2**
The LSISASx12 sets this read-only bit when either the Link Layer or Transport Layer detect a problem with a received FIS that caused the Link Layer to assert R_ERR. Bits [8:10] of this register provide the cause of the error. The Error Log – R_ERR Transmitted/Received

Count register provides a cumulative count of R_ERR primitives that have been sent or received.

ERROR Primitive Received **1**
The LSISASx12 sets this read-only bit when it detects an ERROR primitive between a SATA_SOF and SATA_EOF pair.

Bad Dword Received **0**
The LSISASx12 sets this read-only bit when it detects an invalid dword between a SATA_SOF and SATA_EOF pair. The invalid dword is caused by either an 8b/10b encoding violation or a running disparity error. The Error Log – Invalid DWord register provides a cumulative count of invalid dword errors.

4.6 Expander Connection Manager Registers

The connection manager register space consists of phy configuration registers, a global configuration register, and remote bank configuration registers. Section 4.6.1, "Phy Configuration Registers," describes the phy configuration registers and global configuration register. Section 4.6.1, "Phy Configuration Registers," describes the remote bank configuration registers.

4.6.1 Phy Configuration Registers

Table 4.12 provides the offsets and register locations for each phy in the expander connection manager register map.

Table 4.12 Connection Manager Phy Offsets

Offset	Register Name
0x00000	Phy 0 ECM Config Register
0x00004	Phy 0 Remote Bank Enable
0x00008–0x0000C	Reserved
0x00010	Phy 1 ECM Config Register
0x00014	Phy 1 Remote Bank Enable
0x00018–0x0001C	Reserved
0x00020	Phy 2 ECM Config Register

Table 4.12 Connection Manager Phy Offsets (Cont.)

Offset	Register Name
0x00024	Phy 2 Remote Bank Enable
0x00028–0x0002C	Reserved
0x00030	Phy 3 ECM Config Register
0x00034	Phy 3 Remote Bank Enable
0x00038–0x0003C	Reserved
0x00040	Phy 4 ECM Config Register
0x00044	Phy 4 Remote Bank Enable
0x00048–0x0004C	Reserved
0x00050	Phy 5 ECM Config Register
0x00054	Phy 5 Remote Bank Enable
0x00058–0x0005C	Reserved
0x00060	Phy 6 ECM Config Register
0x00064	Phy 6 Remote Bank Enable
0x00068–0x0006C	Reserved
0x00070	Phy 7 ECM Config Register
0x00074	Phy 7 Remote Bank Enable
0x00078–0x0007C	Reserved
0x00080	Phy 8 ECM Config Register
0x00084	Phy 8 Remote Bank Enable
0x00088–0x0008C	Reserved
0x00090	Phy 9 ECM Config Register
0x00094	Phy 9 Remote Bank Enable
0x00098–0x0009C	Reserved
0x000A0	Phy 10 ECM Config Register
0x000A4	Phy 10 Remote Bank Enable
0x000A8–0x000AC	Reserved
0x000B0	Phy 11 ECM Config Register
0x000B4	Phy 11 Remote Bank Enable
0x000B8–0x000BC	Reserved
0x000C0	STP Target ECM Config Register
0x000C4	STP Target Remote Bank Enable
0x000C8–0x000CC	Reserved

Table 4.12 Connection Manager Phy Offsets (Cont.)

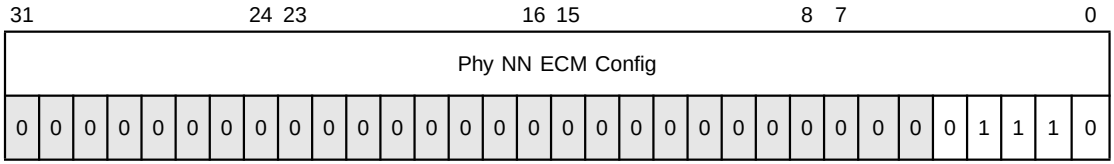
Offset	Register Name
0x00024	Phy 2 Remote Bank Enable
0x00028–0x0002C	Reserved
0x00030	Phy 3 ECM Config Register
0x00034	Phy 3 Remote Bank Enable
0x00038–0x0003C	Reserved
0x00040	Phy 4 ECM Config Register
0x00044	Phy 4 Remote Bank Enable
0x00048–0x0004C	Reserved
0x00050	Phy 5 ECM Config Register
0x00054	Phy 5 Remote Bank Enable
0x00058–0x0005C	Reserved
0x00060	Phy 6 ECM Config Register
0x00064	Phy 6 Remote Bank Enable
0x00068–0x0006C	Reserved
0x00070	Phy 7 ECM Config Register
0x00074	Phy 7 Remote Bank Enable
0x00078–0x0007C	Reserved
0x00080	Phy 8 ECM Config Register
0x00084	Phy 8 Remote Bank Enable
0x00088–0x0008C	Reserved
0x00090	Phy 9 ECM Config Register
0x00094	Phy 9 Remote Bank Enable
0x00098–0x0009C	Reserved
0x000A0	Phy 10 ECM Config Register
0x000A4	Phy 10 Remote Bank Enable
0x000A8–0x000AC	Reserved
0x000B0	Phy 11 ECM Config Register
0x000B4	Phy 11 Remote Bank Enable
0x000B8–0x000BC	Reserved
0x000C0	STP Target ECM Config Register
0x000C4	STP Target Remote Bank Enable
0x000C8–0x000CC	Reserved

Table 4.12 Connection Manager Phy Offsets (Cont.)

Offset	Register Name
0x000D0	SMP Target ECM Config Register
0x000D4	SMP Target Remote Bank Enable
0x000D8–0x01FFF	Reserved
0x02000	Global Config

The following provides descriptions of the phy configuration registers, and the Global Configuration register.

Register: 0xn0
ECM Config
Read/Write



This register provides per-phy configuration of arbitration and routing attributes.

Reserved **[31:5]**

Reserved **4**
This bit is reserved for LSI Logic use. Clear this bit to 0b0 for normal operation.

Reserved **3**
This bit is reserved for LSI Logic use. Clear this bit to 0b0 for normal operation.

Connect Wait **2**
If the Partial Wait bit (bit 1 of this register) is set, then this bit determines the action taken by the Connection Manager when

- at least one destination matches the requested destination
- no destinations that match are available, and
- at least one of the destinations is connected.

Clearing this bit causes the Connection Manager to stop arbitration and to report OPEN REJECT(PathwayBlocked).

Setting this bit causes the Connection Manager to continue arbitration to complete the requested connection, and to report Wait On Connect to the requester.

Partial Wait **1**

This bit determines the operation of the Connection Manager when

- at least one destination matches the requested destination,
- no destinations that match are available, and
- all of the matching links have a link status of either Blocked or Partial.

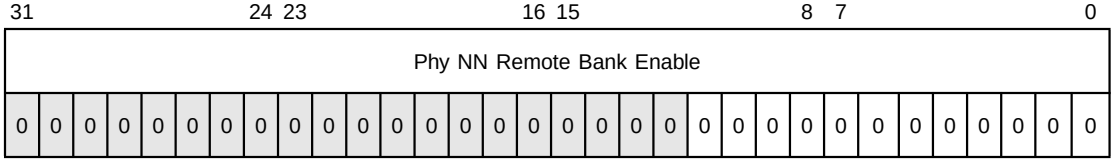
Clearing this bit causes the Connection Manager stop arbitration and report OPEN REJECT(PathwayBlocked). Setting this bit causes the Connection Manager to continue arbitration to complete the requested connection, and to report WaitOnPartial or BlockedPartial to the requester.

Subtractive Decode Enable **0**

Setting this bit causes this phy to utilize subtractive routing.

Register: 0xn4

Remote Bank Enable
Read/Write

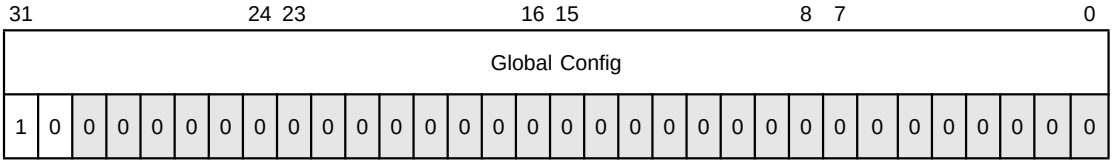


This register specifies the remote routing banks that Phy [NN] uses.

Reserved **[31:12]**

Phy NN Remote Bank Enables **[11:0]**
This field contains a bit vector specifying which remote routing banks Phy NN uses. A set bit indicates that the LSISASx12 utilizes remote route entries within the corresponding bank during routing comparisons for this phy.

Register: 0x2000
Global Config
Read/Write



Remote Disable Under Control of Link Ready **31**
Setting this bit sets all of the Remote Entry Disable bits in a remote entry bank to the disabled state when Link Ready goes to inactive. This setting affects all links that are mapped to the bank. Clearing this bit allows individual control of the Remote Entry Disable bits.

Round Robin Disable **30**
Setting this bit causes the Request state machine to use a priority encoder to determine which grant to accept when it receives multiple grants, as is the case with wide ports. As a result, if the same grants are always received, then the same port is always accepted.

Clearing this bit causes the Request state machine to use a round-robin arbitration scheme to accept a different port each time.

Reserved **[29:0]**

4.6.2 Remote Bank Configuration Registers

Table 4.13 provides the offsets of each remote bank in the expander connection manager register map. The register space for each Remote Bank [NN] Config is a 96 byte space with the format defined in Table 4.14. The format and register definitions for each remote bank are identical with the exception of a differing offset.

Table 4.13 Connection Manager Remote Bank Offsets

Offset	Register Name
0x10000	Remote Bank 00 Config
0x100C0	Remote Bank 01 Config
0x10180	Remote Bank 02 Config
0x10240	Remote Bank 03 Config
0x10300	Remote Bank 04 Config
0x103C0	Remote Bank 05 Config
0x10480	Remote Bank 06 Config
0x10540	Remote Bank 07 Config
0x10600	Remote Bank 08 Config
0x106C0	Remote Bank 09 Config
0x10780	Remote Bank 10 Config
0x10840	Remote Bank 11 Config
0x10900–0x1FFFF	Reserved

Table 4.14 Connection Manager Remote Bank Register Map

Offset from Remote Bank	Register Name
0x00	Remote Bank NN SAS Address High 0
0x04	Remote Bank NN SAS Address Low 0
0x08	Remote Bank NN Config 0
0x0C	Reserved
0x10	Remote Bank NN SAS Address High 1
0x14	Remote Bank NN SAS Address Low 1
0x18	Remote Bank NN Config 1
0x1C	Reserved
0x20	Remote Bank NN SAS Address High 2
0x24	Remote Bank NN SAS Address Low 2
0x28	Remote Bank NN Config 2
0x2C	Reserved
0x030	Remote Bank NN SAS Address High 3
0x34	Remote Bank NN SAS Address Low 3

Table 4.14 Connection Manager Remote Bank Register Map (Cont.)

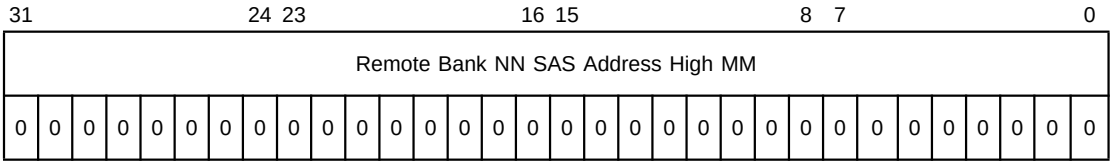
Offset from Remote Bank	Register Name
0x38	Remote Bank NN Config 3
0x3C	Reserved
0x40	Remote Bank NN SAS Address High 4
0x44	Remote Bank NN SAS Address Low 4
0x048	Remote Bank NN Config 4
0x4C	Reserved
0x50	Remote Bank NN SAS Address High 5
0x54	Remote Bank NN SAS Address Low 5
0x58	Remote Bank NN Config 5
0x5C	Reserved
0x60	Remote Bank NN SAS Address High 6
0x64	Remote Bank NN SAS Address Low 6
0x68	Remote Bank NN Config 6
0x6C	Reserved
0x70	Remote Bank NN SAS Address High 7
0x74	Remote Bank NN SAS Address Low 7
0x78	Remote Bank NN Config 7
0x7C	Reserved
0x80	Remote Bank NN SAS Address High 8
0x84	Remote Bank NN SAS Address Low 8
0x88	Remote Bank NN Config 8
0x8C	Reserved
0x90	Remote Bank NN SAS Address High 9
0x94	Remote Bank NN SAS Address Low 9
0x98	Remote Bank NN Config 9
0x9C	Reserved
0xA0	Remote Bank NN SAS Address High 10
0xA4	Remote Bank NN SAS Address Low 10
0xA8	Remote Bank NN Config 10
0xAC	Reserved
0xB0	Remote Bank NN SAS Address High 11

Table 4.14 Connection Manager Remote Bank Register Map (Cont.)

Offset from Remote Bank	Register Name
0xB4	Remote Bank NN SAS Address Low 11
0xB8	Remote Bank NN Config 11
0xBC	Reserved

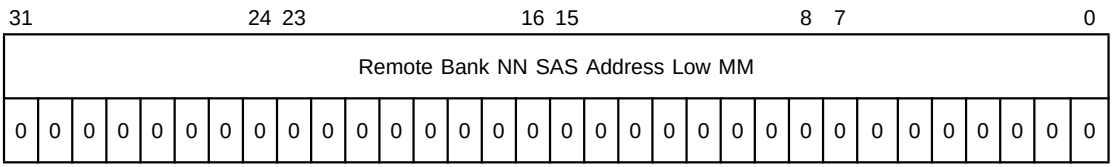
The following provides descriptions of the Remote Bank registers.

Register: 0x00
Remote Bank NN SAS Address High
Read/Write



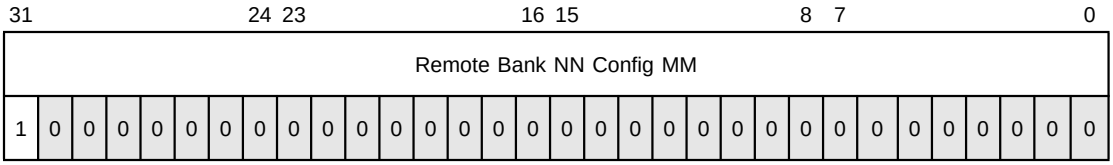
Remote Bank NN SAS Address High MM [31:0]
This register specifies the Remote SAS Address [63:32] present in entry MM of bank NN of the remote routing table.

Register: 0x04
Remote Bank NN SAS Address Low
Read/Write



Remote Bank NN SAS Address Low MM [31:0]
This register specifies the Remote SAS Address [31:0] present in entry MM of bank NN of the remote routing table.

Register: 0x08
Remote Bank NN Config
Read/Write



Remote Entry Disable **31**
Setting this bit disables the routing entry MM of bank NN of the remote routing table.

Reserved **[30:0]**

4.7 EMB Slave Registers

The EMB slave register space is separate from the primary internal register space, and can only be accessed through the EMB. Table 4.15 provides the EMB slave register map. Detailed register descriptions follow the register map.

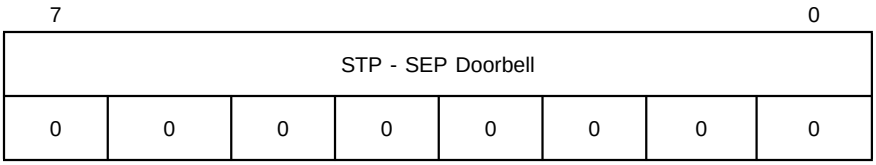
Table 4.15 I^2C Slave Register Map

Address	Register
0x000-0x003	Reserved
0x004–0x41F	Data FIS RAM
0x420–0x4FF	Reserved
0x500	STP - SEP Doorbell
0x501	SEP - STP Doorbell
0x502	STP - SEP Transfer Length
0x503	SEP - STP Transfer Length
0x504	Transfer Attributes
0x505	Command Status
0x506	Error Detection Control
0x507	Error Status
0x508–0x5FF	Reserved
0x600–0x603	RR_Command0 - RR_Command3

Table 4.15 I^2C Slave Register Map (Cont.)

Address	Register
0x604–0x60B	RR_Data0 - RR_Data7
0x60C	RR_Control
0x60D–0xFFF	Reserved

Register: 0x500
STP - SEP Doorbell
Read/Write



Execute Device Diagnostics Received **7**
This read-only bit indicates that the LSISASx12 received an Execute Device Diagnostics command. The SEP can perform diagnostic tests and write the Command Status register with 0x01 if the tests pass, or 0x00 if the tests fail. Write a 1 to this bit to clear it.

Abort Command **6**
Write this bit once to clear it. This bit causes the SEP to abort the current command for one of the following reasons:

- The STP Target was DISABLED by an SMP PHY CONTROL Request
- An R_ERR primitive was received in response to a transmitted DataFIS
- An R_ERR primitive was transmitted in response to a received DataFIS
- A BREAK sequence was detected
- An Unexpected SYNC primitive was detected during the transmission or reception of a Data FIS
- The STP Target detects that the host reconnected to the target after the connection was previously closed. This indicates that the host is probably aborting the

command and reconnected to send a Device Reset command.

- The STP Target tried to re-establish a connection by sending an OAF, and either received an OPEN Timeout or a fatal OPEN_REJECT primitive, such as NO DESTINATION and WRONG DESTINATION.

Link/Device Reset Received **5**
This bit indicates that the LSISASx12 detected either a link reset from an SMP PHY CONTROL (Link/Hard Reset) Request Frame or received a DEVICE RESET command. The SEP aborts the current operation and sets a flag to indicate a UNIT ATTENTION condition. Write this bit once to clear it. The SEP also sets the SEP Aborted Command bit in the SEP - STP Doorbell register.

Read Data Transmitted **4**
This bit is an acknowledgement to the Read Data Ready bit of the SEP-STP Doorbell register. It indicates that the STP Target has transferred the data from the Data FIS RAM to the host. Write this bit once to clear it.

Write Data Ready **3**
This bit indicates that there is valid write data in the Data FIS RAM to transfer to the SEP. Write this bit once to clear it.

Soft Reset Sequence Received **2**
This bit indicates that the LSISASx12 detected a soft reset sequence. The SEP must abort any command under execution and runs diagnostic tests. Upon completion of the diagnostic tests, the SEP must write the results to the Command Status register. A value of 0x00 indicates that the diagnostics tests failed. A value of 0x01 indicates that the diagnostics tests passed. The SEP also must set a UNIT ATTENTION flag when this bit is set.

Identify Packet Device Command Received **1**
This bit indicates that an Identify Packet Device Command was received. The SEP transfers the 512 bytes of response data to the Data FIS RAM and sets the Read Data Ready bit of the SEP-STP Doorbell register. Write this bit once to clear it.

Packet Command Received **0**

This bit indicates that an ATA Packet command was received. The Data FIS RAM contains the CDB for the Packet command.

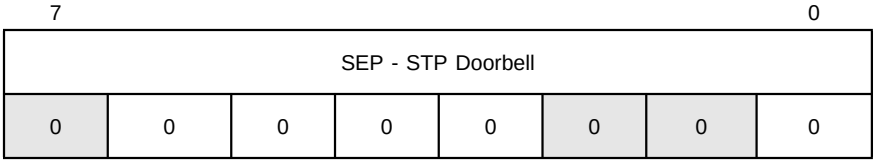
The STP-SEP Transfer Length register contains the Byte Count Limit parameter of the Packet command. A non-zero value indicates the number of dwords that can be transferred in each Data FIS required to complete the Packet command. A value of 0x00 indicates that up to 256 dwords (1 Kbyte) can be transferred at a time.

If the CDB indicates a data-out operation the SEP sets the SEP-STP Transfer Length Register with the value of the Byte Count Limit. The SEP-STP Transfer Attributes register indicates the data direction and whether or not the next data transfer is the last transfer for the Packet command. The SEP sets the Packet Command Acknowledge bit of the SEP-STP Doorbell register.

If the CDB indicates a data-in operation the SEP sets the SEP-STP Transfer Attributes register to indicate the data direction and whether or not the next data transfer is the last transfer for the Packet command. After transferring the appropriate data to the Data FIS RAM, the SEP sets the Read Data Ready bit of the SEP-STP Doorbell Register. The Packet Command Acknowledge bit is not set for write operations.

If the CDB has an unsupported command or invalid parameters, the SEP writes the Command Status register with a Sense Key of “ILLEGAL REQUEST” (0x05) and sets the Command Status Ready bit of the SEP-STP Doorbell Register.

Register: 0x501
SEP - STP Doorbell
Read/Write



Reserved **7**

SEP Aborted Command **6**
This bit is set when an I^2C CRC error occurs while the SEP is transferring data to or from the I^2C Slave.

Command Status Ready **5**
The SEP sets this bit to indicate that the Command Status register has valid status information.

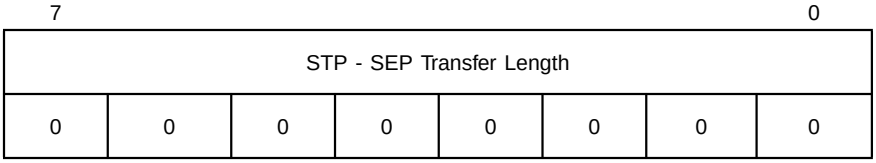
Read Data Ready **4**
The SEP sets this bit to indicate that the Data FIS RAM contains valid data for the STP target to transfer to the host.

Write Data Acknowledge **3**
The SEP sets this bit to indicate that the SEP has transferred the write data from the Data FIS RAM.

Reserved **[2:1]**

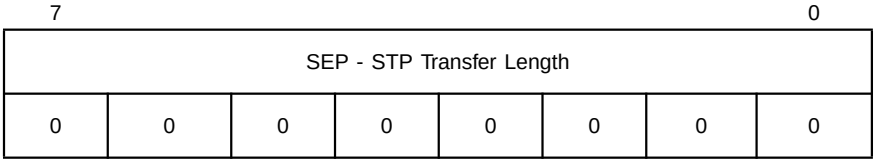
PACKET Command ACK **0**
This bit is set after a Packet data-in command is parsed successfully. The Read Data Ready bit is used when a Packet data-out command is received.

Register: 0x502
STP - SEP Transfer Length
Read/Write



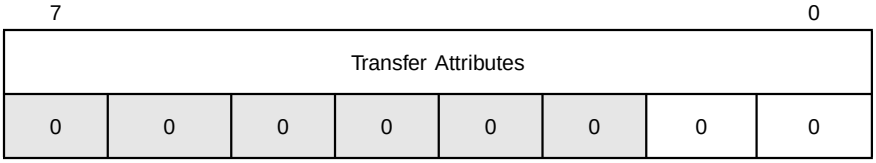
Transfer Length **[7:0]**
This register contains the dword value indicating the maximum length of each Data FIS to be transferred from the host to the SEP. A value of 0x00 indicates a transfer length of 256 dwords.

Register: 0x503
SEP - STP Transfer Length
Read/Write



Transfer Length **[7:0]**
This register contains the dword value indicating the maximum length of each Data FIS to be transferred from the SEP to the host.

Register: 0x504
Transfer Attributes
Read/Write

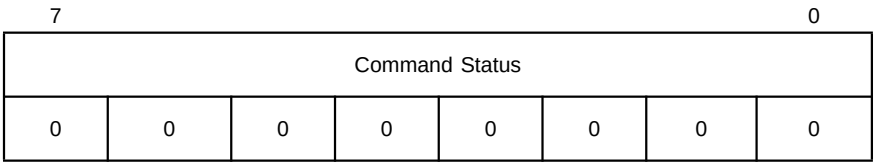


Reserved **[7:2]**

Last Transfer **1**
For data-in packet commands such as Receive Diagnostics, a value of 1 indicates that the current data in the Data FIS buffer that is to be transmitted to the host is the last data transfer required to complete the Packet command. For data-out packet commands such as Send Diagnostics, a value of 1 indicates that the last block of data read from the Data FIS buffer by the SEP is the last data transfer required to complete the packet command.

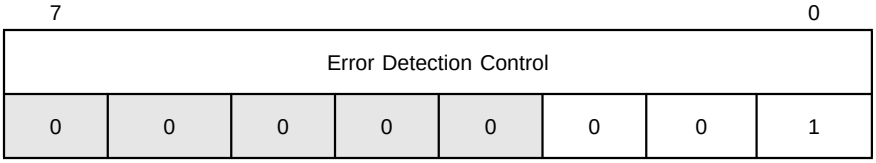
Data Direction **0**
A value of 0b0 indicates a data-out packet command (host to device). A value of 0b1 indicates a data-in packet command (device to host).

Register: 0x505
Command Status
Read/Write



Command Status **[7:0]**
This register contains the status of the current packet command.

Register: 0x506
Error Detection Control
Read/Write



Reserved **[7:3]**

Payload CRC Error **2**
When CRC checking is enabled, the LSISASx12 sets this bit when it detects a CRC error in the payload bytes of a write transfer. The LSISASx12 asserts an interrupt to the SEP to alert the SEP of the CRC error condition. Write a 1 to this bit to clear it.

The Transfer Type bits of the Error Status register indicate the type of I^2C write transaction (byte or block) that contain the CRC error. The Actual CRC and Expected CRC registers indicate the actual CRC that was received and the CRC value that was expected.

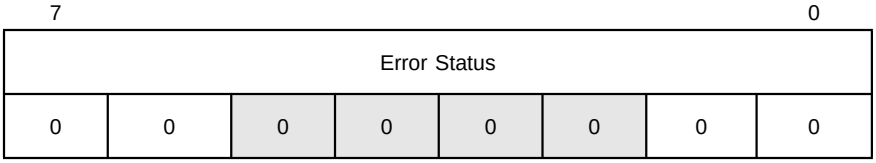
Header CRC Error **1**
When CRC checking is enabled, the LSISASx12 sets this bit when it detects a CRC error in the header bytes of a write or read transfer. The LSISASx12 asserts an interrupt to the SEP to alert the SEP of the CRC error condition. Write a 1 to this bit to clear it.

The Transfer Type bits of the Error Status register indicate the type of I^2C write transaction (byte or block) that contain the CRC error. The Actual CRC and Expected CRC registers indicate the actual CRC that was received and the CRC value that was expected.

Enable CRC **0**
Setting this bit enables CRC error detection for I^2C transfers. Clearing this bit disables error detection. Error detection is enabled after a reset.

Register: 0x507

Error Status
Read/Write



Transaction Type **7**

This bit indicates the type of transaction in progress when a CRC error or transfer length error occurred. The value of this register bit remains latched until another error is detected.

If a transfer length error occurred:

- A value of 1 indicates a READ transaction had a transfer length error.
- A value of 0 indicates a WRITE transaction had a transfer length error.

If a payload CRC error occurred, a value of 0 indicates a WRITE transaction had a payload CRC error. (A payload CRC error cannot be detected by the SEMB for a READ transaction because the SEMB transmits the payload CRC for READ transactions).

If a header CRC error occurred, this bit indicates the value of the least-significant bit of the Control Byte for the transfer that had the header CRC error.

Transfer Type **6**

This bit indicates the type of transfer in progress when a CRC error or Transfer Length error occurred. The value remains latched until another error is detected.

If a payload CRC error or transfer length error occurred:

- A value of 1 indicates a BLOCK transaction had the error.
- A value of 0 indicates a BYTE transaction had the error.

If a header CRC error occurred, this bit indicates the value of the most-significant bit (BLK) of the Address

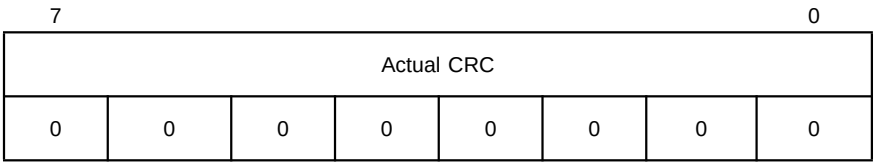
Pointer Hi byte for the transfer that had the header CRC error.

Reserved **[5:2]**

Boot Error Doorbell Interrupt **1**
This LSISASx12 sets this bit when the boot sequencer indicates that a boot up did not occur properly. Setting this bit generates an interrupt to the SEP. Write a 1 to this bit to clear it.

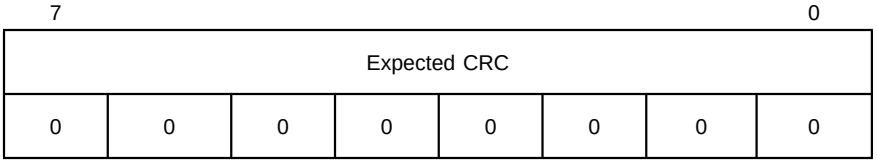
Transfer Length Error Doorbell Interrupt **0**
The LSISASx12 sets this bit when the SEP transfers an incorrect number of bytes during an I^2C write or read transaction. Setting this bit generates an interrupt to the SEP. Write a 1 to this bit to clear it.

Register: 0x508
Actual CRC
Read/Write



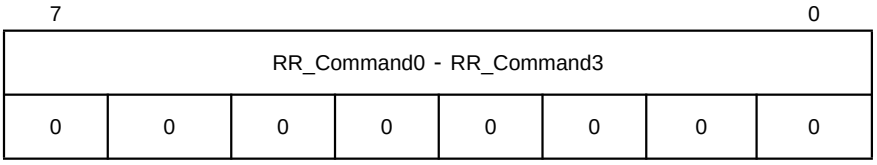
Actual CRC **[7:0]**
When CRC checking is enabled and the LSISASx12 detects a CRC error, this register contains the value of the header or payload CRC that contains the error. The Expected CRC register contains the expected CRC value. This register value remains latched until the CRC error is cleared in the Error Detection Control register.

Register: 0x509
Expected CRC
Read/Write



Expected CRC **[7:0]**
When CRC checking is enabled and the LSISASx12 detects a CRC error, this register contains the value of the header or payload CRC was expected. The Actual CRC register contains the CRC value that was received. This register value remains latched until the CRC error is cleared in the Error Detection Control register.

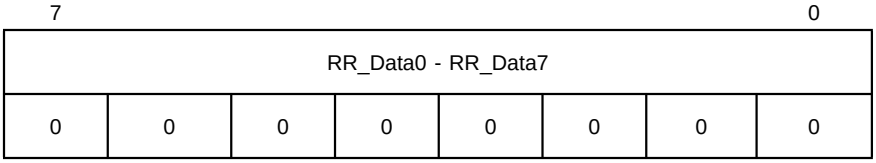
Register: 0x600 - 0x603
RR_Command0 - RR_Command3
Read/Write



RR_Command **[7:0]**
These four registers contain the value to be presented during the Command/Address phase on the register bus interface. RR_Command0 contains the least-significant byte of the command. RR_Command3 contains the most-significant byte of the command.

Register: 0x604 - 0x60B

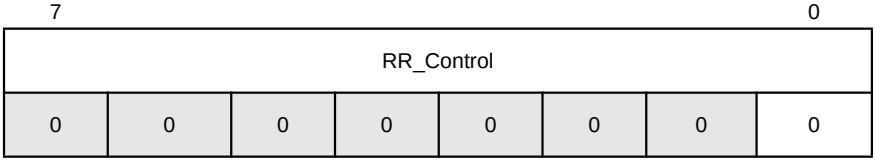
RR_Data0 - RR_Data7
Read/Write



RR_Data **[7:0]**
These eight registers contain the value presented during the data phases of a write operation on the register bus interface. On read operations these registers contain the data returned from the register bus. RR_Data0 contains the least-significant byte of the data. RR_Data7 contains the most-significant byte of the data.

Register: 0x60C

RR_Control
Read/Write



Reserved **[7:1]**

RR_Request **0**
Setting this bit initiates the register bus operation described in the RR_Command–RR_Command3 registers. This bit must be cleared following the completion of each operation on the register bus interface.

Chapter 5
Specifications

This chapter specifies the LSISASx12 electrical and mechanical characteristics. It is divided into the following sections:

- Section 5.1, “DC Characteristics”
- Section 5.2, “AC Characteristics”
- Section 5.3, “Pinout”
- Section 5.4, “Package Diagram”

Refer the *Serial Attached SCSI Standard* for timing information. The LSISASx12 timings conform to the information that these specifications provide.

5.1 DC Characteristics

This section of the manual describes the LSISASx12 DC characteristics. Tables 5.1 through 5.18 give environmental, current, voltage, and capacitance specifications.

Table 5.1 Absolute Maximum Stress Ratings[1]

Symbol	Parameter	Min	Max	Unit	Test Conditions
T_{STG}	Storage Temperature	–65	150	°C	–
$V_{DD\text{-}Core}$	Supply Voltage	-0.3	2.0	V	–
$V_{DD\text{-}IO}$	I/O Supply Voltage	-0.3	3.96	V	–
I_{LP}	Latch-up Current	150	–	mA	EIA/JESD78
ESD_{HBM}	Electrostatic Discharge - Human Body Model (HBM)	–	2000	V	JESD-A114-B

1. Stresses beyond those listed above can damage the device. These are stress ratings only; functional operation of the device at or beyond these values is not implied.

Table 5.2 Operating Conditions[1]

Symbol	Parameter	Min	Nominal	Max	Unit	Test Conditions
$V_{DD\text{-}Core}$	Core and Analog Supply Voltage	1.14	1.2	1.26	V	–
$V_{DD\text{-}IO}$	I/O Supply Voltage	2.97	3.3	3.63	V	–
$I_{DD\text{-}Core}$	Core and Analog Supply Current (dynamic)[2]	–	–	2	A	–
$I_{DD\text{-}I/O}$	I/O Supply Current (dynamic)[3]	–	–	90	mA	FAULT_LED[11:0]/ signals asserted
T_j	Junction Temperature	–	–	115	°C	–
T_A	Operating Free Air	–	–	70	°C	–
θ_{JA}	Thermal Resistance (junction to ambient air)	–	17.8	–	°C/W	0 Linear Feet/Minute

1. Conditions that exceed the operating limits can cause the device to function incorrectly.
2. Core and analog supply only.
3. These numbers are specified for the design of the I/O power network. Not all of the $I_{DD\text{-}I/O}$ supplied to the LSISASx12 dissipates on-chip.

There are no power sequencing requirements for the LSISASx12 expander.

Table 5.3 GigaBlaze Transmitter Voltage Characteristics — TX[11:0]

Speed and Technology	Parameter	Min V_{p-p} Inside EYE	Max V_{p-p} Outside EYE	Unit
SAS - 1.5 Gbit/s	Peak-to-Peak Voltage (V_{p-p})	1050	1180	mV
SAS - 3.0 Gbit/s	V_{p-p}	1658	1780	mV
SATA - 1.5 Gbit/s	V_{p-p}	476	620	mV
SATA - 3.0 Gbit/s	V_{p-p}	505	694	mV

Table 5.4 GigaBlaze Receiver Voltage Characteristics — RX[11:0]

Parameter	Min	Max	Unit	Condition
$V_{p-p–OOB}$	150	–	mV	inside the EYE
$V_{p-p–normal\ operation}$	275	–	mV	inside the EYE

Table 5.5 GigaBlaze Transceiver Rise/Fall Characteristics — TX[11:0] ,RX[11:0]

Speed and Technology	Nominal Rise Time	Nominal Fall Time	Specified Range	Unit
SAS - 1.5 Gbit/s	141	153	67 - 273	psec
SAS - 3.0 Gbit/s	129	125	67 - 137	psec
SATA - 1.5 Gbit/s	141	141	100 - 273	psec
SATA - 3.0 Gbit/s	112	112	66.6 - 136.6	psec

Table 5.6 5 mA Bi-directional Signals — BSL_SCL, BSL_SD, EMB_SCL, EMB_SD, GPIO[3:0], LED_ACTIVE[11:0]/, LED_FAULT[11:0]/, LED_STATUS[11:0]/

Symbol	Parameters	Min	Max	Unit
V_{IL}	Input low voltage	VSS - 0.5	0.8	V
V_{IH}	Input high voltage	2	VDD + 0.3	V
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA
$I_{PULL-UP}$	Pull current	70	200	µA

Table 5.7 5 mA Bi-directional Signals — CABLE_DET[11:0]/

Symbol	Parameters	Min	Max	Unit
V_{IL}	Input low voltage	VSS - 0.5	0.8	V
V_{IH}	Input high voltage	2	VDD + 0.3	V
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA
$I_{PULL-DOWN}$	Pull current	70	350	µA

Table 5.8 10 mA, 3-state Outputs Signals — EMB_CRC_INT, EMB_DB_INT

Symbol	Parameters	Min	Max	Unit
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA

Table 5.9 5 mA, 3-state Outputs — TDO

Symbol	Parameters	Min	Max	Unit
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA

Table 5.10 5 mA Outputs — UART_TX, LEDSYNCOUT

Symbol	Parameters	Min	Max	Unit
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA

Table 5.11 4 mA Outputs — PROCMON

Symbol	Parameters	Min	Max	Unit
V_{OL}	Output low voltage	–	0.4	V
V_{OH}	Output high voltage	2.4	–	V
I_{OZ}	3-state leakage	-10	10	µA

Table 5.12 Inputs — ISTWI_ADDR[1:0], IDDT

Symbol	Parameters	Min	Max	Unit
V_{IL}	Input low voltage	VSS - 0.5	0.8	V
V_{IH}	Input high voltage	2	VDD + 0.3	V
I_{OZ}	3-state leakage	-10	10	µA

Table 5.13 Inputs — UART_RX, LEDSYNCIN, TN

Symbol	Parameters	Min	Max	Unit
V_{IL}	Input low voltage	VSS - 0.5	0.8	V
V_{IH}	Input high voltage	2	VDD + 0.3	V
I_{OZ}	3-state leakage	-10	10	µA
$I_{PULL-UP}$	Pull current	70	200	µA

Table 5.14 Inputs — MODE[3:0], SCAN_ENABLE, SCAN_MODE

Symbol	Parameters	Min	Max	Unit
V_{IL}	Input low voltage	VSS - 0.5	0.8	V
V_{IH}	Input high voltage	2	VDD + 0.3	V
I_{OZ}	3-state leakage	-10	10	µA
$I_{PULL-DOWN}$	Pull current	70	350	µA

Table 5.15 Schmitt Trigger Inputs — CLK

Symbol	Min	Nominal	Max	Unit
VT+	–	1.6	2	V
VT-	1	1.2	–	V
Hysteresis	0.3	0.4	–	V
I_{IN}	-10	–	10	µA

Table 5.16 Schmitt Trigger Inputs — RESET/, SCANCLK1, SCANCLK2, SCANCLK3, TCK, TDI, TMS, TRST/

Symbol	Min	Nominal	Max	Unit
VT+	–	1.6	2	V
VT-	1	1.2	–	V
Hysteresis	0.3	0.4	–	V
I_{IN}	-10	–	10	µA
$I_{PULL-UP}$	70	105	200	µA

Table 5.17 PECL Buffer — REFCLK+, REFCLK-

Symbol	Min	Nominal	Max	Unit
V_{IN_CM}	1.6	2	2.4	V
$VT_{IN\_DIFF\_PP}$	600	–	2000	mV
V_{IL}	0.6	–	2.1	V
V_{IH}	1.9	–	3.4	V
I_{IN}	-10	–	10	µA

Table 5.18 Capacitance[1]

Symbol	Parameters	Nominal	Unit
C_{IN}	Input Capacitance	3.5	pF
C_{OUT}	Output Capacitance	3.5	pF
$C_{IN-PECL}$	PECL Input Capacitance	1.0	pF

1. Capacitance values do not include package capacitance.

5.2 AC Characteristics

The AC characteristics described in this section apply over the entire range of operating conditions (refer to Section 5.1, “DC Characteristics.”) Chip timing is based on simulation at worst case voltage, temperature, and processing. Timing was developed with a load capacitance of 50 pF.

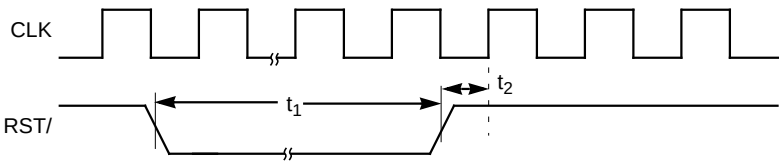
The LSISASx12 requires a 75MHz input clock having an accuracy of at least 50ppm on both the REFCLK and CLK pins. The duty cycle required is 40-60% worst case. Refer to SEN S11054: LSISASx12 Design Considerations (Document Number: DB05-000116-xx) for information concerning using a single oscillator to drive both input clocks.

Table 5.19 and Figure 5.1 provide reset input timing data.

Table 5.19 Reset Input

Symbol	Parameter	Min	Max	Units
t_1	Reset pulse width	10	–	ns
t_2	Reset deasserted setup to CLK HIGH	0	–	ns

Figure 5.1 Reset Input



5.3 Pinout

Table 5.20 provides the signal listing by signal name. Table 5.21 provides the signal listing by BGA position. Figure 5.2 provides a BGA diagram.

Table 5.20 Listing by Signal Name[1, 2]

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
BSL_SCL	AE3	LED_STATUS[11]/	AF9	NC	E13	NC	AC14
BSL_SD	AD4	LEDSYNCIN	AF15	NC	E14	NC	AC15
CBL_DET[0]/	E16	LEDSYNCOUT	AE17	NC	E15	NC	AC18
CBL_DET[1]/	D15	MODE[0]	B4	NC	E22	NC	AC19
CBL_DET[2]/	C16	MODE[1]	A4	NC	E26	NC	AC20
CBL_DET[3]/	B16	MODE[2]	C5	NC	F1	NC	AC21
CBL_DET[4]/	D14	MODE[3]	B5	NC	F7	NC	AC26
CBL_DET[5]/	A16	NC	A3	NC	F8	NC	AD3
CBL_DET[6]/	B8	NC	A9	NC	F9	NC	AD11
CBL_DET[7]/	C8	NC	A10	NC	F20	NC	AD17
CBL_DET[8]/	A8	NC	A11	NC	F21	NC	AD19
CBL_DET[9]/	A7	NC	A12	NC	F22	NC	AD25
CBL_DET[10]/	A6	NC	A13	NC	F24	NC	AE1
CBL_DET[11]/	A5	NC	A14	NC	G3	NC	AE10
CLK	AE11	NC	A15	NC	G19	NC	AE14
EMB_CRC_INT	AD5	NC	A17	NC	G20	NC	AF8
EMB_DB_INT	AF4	NC	A18	NC	G24	NC	AF10
EMB_SCL	AE4	NC	A19	NC	H3	NC	AF11
EMB_SD	AF3	NC	A20	NC	H24	NC	AF12
GPIO[0]	AD24	NC	A21	NC	J26	NC	AF13
GPIO[1]	AA21	NC	A22	NC	K5	NC	AF14
GPIO[2]	AE24	NC	A23	NC	L1	PLLVDD	B10
GPIO[3]	AC23	NC	A24	NC	L4	PLLVSS	C10
IDDT	A2	NC	B3	NC	L5	PROCMON	D20
ISTWI_ADDR[0]	AE5	NC	B9	NC	M3	REFCLK+	AB13
ISTWI_ADDR[1]	AA8	NC	B13	NC	M4	REFCLK–	AC13
LED_ACT[0]/	AF23	NC	B17	NC	M5	RESET/	AD14
LED_ACT[1]/	AD22	NC	B18	NC	N3	RTRIM	AA6
LED_ACT[2]/	AE22	NC	B21	NC	N4	RX[0]+	AB24
LED_ACT[3]/	AF22	NC	B22	NC	N26	RX[0]–	AB25
LED_ACT[4]/	AA18	NC	B23	NC	P23	RX[1]+	V24
LED_ACT[5]/	AF21	NC	B24	NC	P24	RX[1]–	V25
LED_ACT[6]/	AF7	NC	B25	NC	R22	RX[2]+	N25
LED_ACT[7]/	AF6	NC	C4	NC	R24	RX[2]–	N24
LED_ACT[8]/	AF5	NC	C9	NC	T1	RX[3]+	J25
LED_ACT[9]/	AE6	NC	C13	NC	T22	RX[3]–	J24
LED_ACT[10]/	AC7	NC	C14	NC	T23	RX[4]+	E25
LED_ACT[11]/	AD6	NC	C15	NC	U22	RX[4]–	E24
LED_FAULT[0]/	AD23	NC	C17	NC	V3	RX[5]+	C25
LED_FAULT[1]/	AB21	NC	C18	NC	W24	RX[5]–	C24
LED_FAULT[2]/	AF25	NC	C21	NC	W26	RX[6]+	C2
LED_FAULT[3]/	AA20	NC	C22	NC	Y1	RX[6]–	B2
LED_FAULT[4]/	AF24	NC	C23	NC	Y8	RX[7]+	F2
LED_FAULT[5]/	AE23	NC	D2	NC	Y24	RX[7]–	F3
LED_FAULT[6]/	AC12	NC	D6	NC	AA7	RX[8]+	L2
LED_FAULT[7]/	AD10	NC	D7	NC	AA9	RX[8]–	L3
LED_FAULT[8]/	AB11	NC	D8	NC	AA19	RX[9]+	P1
LED_FAULT[9]/	AE9	NC	D9	NC	AA25	RX[9]–	N1
LED_FAULT[10]/	AD9	NC	D12	NC	AB3	RX[10]+	W3
LED_FAULT[11]/	AC9	NC	D13	NC	AB7	RX[10]–	W2
LED_STATUS[0]/	AB18	NC	D16	NC	AB8	RX[11]+	AC2
LED_STATUS[1]/	AF20	NC	D23	NC	AB9	RX[11]–	AC1
LED_STATUS[2]/	AF19	NC	D25	NC	AB10	RXBVDD[0]	AA23
LED_STATUS[3]/	AB17	NC	E2	NC	AB12	RXBVDD[1]	W23
LED_STATUS[4]/	AE19	NC	E4	NC	AB14	RXBVDD[2]	T26
LED_STATUS[5]/	AD18	NC	E7	NC	AB16	RXBVDD[3]	K25
LED_STATUS[6]/	AE18	NC	E8	NC	AB19	RXBVDD[4]	J22
LED_STATUS[7]/	AB15	NC	E9	NC	AB20	RXBVDD[5]	G22
LED_STATUS[8]/	AF18	NC	E10	NC	AB23	RXBVDD[6]	H7
LED_STATUS[9]/	AF17	NC	E11	NC	AC6	RXBVDD[7]	H5
LED_STATUS[10]/	AF16	NC	E12	NC	AC8	RXBVDD[8]	K3

1. *NC* pins are not connected inside of the package.
2. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy through the Phy Transmit Polarity and Phy Receiver Polarity registers.

Table 5.20 Listing by Signal Name[1, 2] (Cont.)

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
RXBVDD[9]	R4	TX[2]–	T24	TXVSS[2]	U24	VSS	B15
RXBVDD[10]	V4	TX[3]+	L26	TXVSS[3]	N22	VSS	B19
RXBVDD[11]	Y4	TX[3]–	M26	TXVSS[4]	L22	VSS	B20
RXBVSS[0]	Y23	TX[4]+	G26	TXVSS[5]	H22	VSS	D5
RXBVSS[1]	V23	TX[4]–	H26	TXVSS[6]	G4	VSS	D11
RXBVSS[2]	R26	TX[5]+	B26	TXVSS[7]	J4	VSS	D17
RXBVSS[3]	K24	TX[5]–	C26	TXVSS[8]	M1	VSS	E21
RXBVSS[4]	J21	TX[6]+	C1	TXVSS[9]	T5	VSS	E23
RXBVSS[5]	G21	TX[6]–	D1	TXVSS[10]	W5	VSS	F5
RXBVSS[6]	G7	TX[7]+	G1	TXVSS[11]	AA5	VSS	F25
RXBVSS[7]	G5	TX[7]–	H1	UART_RX	E18	VSS	G2
RXBVSS[8]	J2	TX[8]+	P2	UART_TX	F18	VSS	G8
RXBVSS[9]	P4	TX[8]–	P3	VDD2	K23	VSS	G25
RXBVSS[10]	U5	TX[9]+	U1	VDD2	L24	VSS	H2
RXBVSS[11]	W6	TX[9]–	V1	VDD2	M13	VSS	H20
RXVDD[0]	AD26	TX[10]+	AA1	VDD2	M15	VSS	K4
RXVDD[1]	Y26	TX[10]–	AB1	VDD2	M24	VSS	L23
RXVDD[2]	P22	TX[11]+	AC3	VDD2	N12	VSS	L25
RXVDD[3]	K26	TX[11]–	AC4	VDD2	N14	VSS	M2
RXVDD[4]	F26	TXBVDD[0]	Y21	VDD2	P13	VSS	M12
RXVDD[5]	D24	TXBVDD[1]	V21	VDD2	P15	VSS	M14
RXVDD[6]	D3	TXBVDD[2]	V26	VDD2	R3	VSS	M25
RXVDD[7]	E1	TXBVDD[3]	M23	VDD2	R12	VSS	N2
RXVDD[8]	N5	TXBVDD[4]	J23	VDD2	R14	VSS	N13
RXVDD[9]	R1	TXBVDD[5]	G23	VDD2	T3	VSS	N15
RXVDD[10]	W1	TXBVDD[6]	H6	VDD2	U4	VSS	P12
RXVDD[11]	AD1	TXBVDD[7]	J5	VDD2	Y3	VSS	P14
RXVSS[0]	W21	TXBVDD[8]	J1	VDD2	AA3	VSS	P25
RXVSS[1]	V22	TXBVDD[9]	U2	VDD2	AB5	VSS	R2
RXVSS[2]	U26	TXBVDD[10]	V5	VDDIO33	C6	VSS	R13
RXVSS[3]	M22	TXBVDD[11]	Y5	VDDIO33	C7	VSS	R15
RXVSS[4]	H23	TXBVSS[0]	W20	VDDIO33	C11	VSS	R25
RXVSS[5]	F23	TXBVSS[1]	W22	VDDIO33	C12	VSS	T2
RXVSS[6]	G6	TXBVSS[2]	U25	VDDIO33	C19	VSS	T4
RXVSS[7]	J6	TXBVSS[3]	N23	VDDIO33	C20	VSS	U23
RXVSS[8]	K2	TXBVSS[4]	K22	VDDIO33	D10	VSS	W7
RXVSS[9]	R5	TXBVSS[5]	H21	VDDIO33	D22	VSS	W25
RXVSS[10]	W4	TXBVSS[6]	F4	VDDIO33	E5	VSS	Y2
RXVSS[11]	AA4	TXBVSS[7]	H4	VDDIO33	AB22	VSS	Y19
SCAN_ENABLE	D21	TXBVSS[8]	K1	VDDIO33	AC5	VSS	Y25
SCAN_MODE	F19	TXBVSS[9]	U3	VDDIO33	AC11	VSS	AA2
SCANCLK1	D4	TXBVSS[10]	V6	VDDIO33	AC17	VSS	AA22
SCANCLK2	F6	TXBVSS[11]	Y6	VDDIO33	AD7	VSS	AB4
SCANCLK3	C3	TXVDD[0]	AE25	VDDIO33	AD8	VSS	AB6
TCK	E20	TXVDD[1]	AA24	VDDIO33	AD12	VSS	AC10
TDI	D19	TXVDD[2]	R23	VDDIO33	AD13	VSS	AC16
TDIODE_N	Y7	TXVDD[3]	P26	VDDIO33	AD15	VSS	AC22
TDIODE_P	AE2	TXVDD[4]	H25	VDDIO33	AD16	VSS	AE7
TDO	D18	TXVDD[5]	D26	VDDIO33	AD20	VSS	AE8
TMS	E17	TXVDD[6]	E3	VDDIO33	AD21	VSS	AE12
TN	E6	TXVDD[7]	J3	VSS	A25	VSS	AE13
TRST/	E19	TXVDD[8]	P5	VSS	B1	VSS	AE15
TX[0]+	AC25	TXVDD[9]	V2	VSS	B6	VSS	AE16
TX[0]–	AC24	TXVDD[10]	AB2	VSS	B7	VSS	AE20
TX[1]+	AA26	TXVDD[11]	AD2	VSS	B11	VSS	AE21
TX[1]–	AB26	TXVSS[0]	Y20	VSS	B12	VSS	AE26
TX[2]+	T25	TXVSS[1]	Y22	VSS	B14	VSS	AF2

1. *NC* pins are not connected inside of the package.
2. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy through the Phy Transmit Polarity and Phy Receiver Polarity registers.

Table 5.21 Listing by Pin Number[1, 2]

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
A2	IDDT	C12	VDDIO33	E21	VSS	J22	RXBVDD[4]
A3	NC	C13	NC	E22	NC	J23	TXBVDD[4]
A4	MODE[1]	C14	NC	E23	VSS	J24	RX[3]–
A5	CBL_DET[11]/	C15	NC	E24	RX[4]–	J25	RX[3]+
A6	CBL_DET[10]/	C16	CBL_DET[2]/	E25	RX[4]+	J26	NC
A7	CBL_DET[9]/	C17	NC	E26	NC	K1	TXBVSS[8]
A8	CBL_DET[8]/	C18	NC	F1	NC	K2	RXVSS[8]
A9	NC	C19	VDDIO33	F2	RX[7]+	K3	RXBVDD[8]
A10	NC	C20	VDDIO33	F3	RX[7]–	K4	VSS
A11	NC	C21	NC	F4	TXBVSS[6]	K5	NC
A12	NC	C22	NC	F5	VSS	K22	TXBVSS[4]
A13	NC	C23	NC	F6	SCANCLK2	K23	VDD2
A14	NC	C24	RX[5]–	F7	NC	K24	RXBVSS[3]
A15	NC	C25	RX[5]+	F8	NC	K25	RXBVDD[3]
A16	CBL_DET[5]/	C26	TX[5]–	F9	NC	K26	RXVDD[3]
A17	NC	D1	TX[6]–	F18	UART_TX	L1	NC
A18	NC	D2	NC	F19	SCAN_MODE	L2	RX[8]+
A19	NC	D3	RXVDD[6]	F20	NC	L3	RX[8]–
A20	NC	D4	SCANCLK1	F21	NC	L4	NC
A21	NC	D5	VSS	F22	NC	L5	NC
A22	NC	D6	NC	F23	RXVSS[5]	L22	TXVSS[4]
A23	NC	D7	NC	F24	NC	L23	VSS
A24	NC	D8	NC	F25	VSS	L24	VDD2
A25	VSS	D9	NC	F26	RXVDD[4]	L25	VSS
B1	VSS	D10	VDDIO33	G1	TX[7]+	L26	TX[3]+
B2	RX[6]–	D11	VSS	G2	VSS	M1	TXVSS[8]
B3	NC	D12	NC	G3	NC	M2	VSS
B4	MODE[0]	D13	NC	G4	TXVSS[6]	M3	NC
B5	MODE[3]	D14	CBL_DET[4]/	G5	RXBVSS[7]	M4	NC
B6	VSS	D15	CBL_DET[1]/	G6	RXVSS[6]	M5	NC
B7	VSS	D16	NC	G7	RXBVSS[6]	M12	VSS
B8	CBL_DET[6]/	D17	VSS	G8	VSS	M13	VDD2
B9	NC	D18	TDO	G19	NC	M14	VSS
B10	PLLVDD	D19	TDI	G20	NC	M15	VDD2
B11	VSS	D20	PROCMON	G21	RXBVSS[5]	M22	RXVSS[3]
B12	VSS	D21	SCAN_ENABLE	G22	RXBVDD[5]	M23	TXBVDD[3]
B13	NC	D22	VDDIO33	G23	TXBVDD[5]	M24	VDD2
B14	VSS	D23	NC	G24	NC	M25	VSS
B15	VSS	D24	RXVDD[5]	G25	VSS	M26	TX[3]–
B16	CBL_DET[3]/	D25	NC	G26	TX[4]+	N1	RX[9]–
B17	NC	D26	TXVDD[5]	H1	TX[7]–	N2	VSS
B18	NC	E1	RXVDD[7]	H2	VSS	N3	NC
B19	VSS	E2	NC	H3	NC	N4	NC
B20	VSS	E3	TXVDD[6]	H4	TXBVSS[7]	N5	RXVDD[8]
B21	NC	E4	NC	H5	RXBVDD[7]	N12	VDD2
B22	NC	E5	VDDIO33	H6	TXBVDD[6]	N13	VSS
B23	NC	E6	TN	H7	RXBVDD[6]	N14	VDD2
B24	NC	E7	NC	H20	VSS	N15	VSS
B25	NC	E8	NC	H21	TXBVSS[5]	N22	TXVSS[3]
B26	TX[5]+	E9	NC	H22	TXVSS[5]	N23	TXBVSS[3]
C1	TX[6]+	E10	NC	H23	RXVSS[4]	N24	RX[2]–
C2	RX[6]+	E11	NC	H24	NC	N25	RX[2]+
C3	SCANCLK3	E12	NC	H25	TXVDD[4]	N26	NC
C4	NC	E13	NC	H26	TX[4]–	P1	RX[9]+
C5	MODE[2]	E14	NC	J1	TXBVDD[8]	P2	TX[8]+
C6	VDDIO33	E15	NC	J2	RXBVSS[8]	P3	TX[8]–
C7	VDDIO33	E16	CBL_DET[0]/	J3	TXVDD[7]	P4	RXBVSS[9]
C8	CBL_DET[7]/	E17	TMS	J4	TXVSS[7]	P5	TXVDD[8]
C9	NC	E18	UART_RX	J5	TXBVDD[7]	P12	VSS
C10	PLLVSS	E19	TRST/	J6	RXVSS[7]	P13	VDD2
C11	VDDIO33	E20	TCK	J21	RXBVSS[4]	P14	VSS

1. *NC* pins are not connected inside of the package.
2. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy through the Phy Transmit Polarity and Phy Receiver Polarity registers.

Table 5.21 Listing by Pin Number[1, 2] (Cont.)

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
P15	VDD2	W6	RXBVSS[11]	AB15	LED_STATUS[7]/	AD20	VDDIO33
P22	RXVDD[2]	W7	VSS	AB16	NC	AD21	VDDIO33
P23	NC	W20	TXBVSS[0]	AB17	LED_STATUS[3]/	AD22	LED_ACT[1]/
P24	NC	W21	RXVSS[0]	AB18	LED_STATUS[0]/	AD23	LED_FAULT[0]/
P25	VSS	W22	TXBVSS[1]	AB19	NC	AD24	GPIO[0]
P26	TXVDD[3]	W23	RXBVDD[1]	AB20	NC	AD25	NC
R1	RXVDD[9]	W24	NC	AB21	LED_FAULT[1]/	AD26	RXVDD[0]
R2	VSS	W25	VSS	AB22	VDDIO33	AE1	NC
R3	VDD2	W26	NC	AB23	NC	AE2	TDIODE_P
R4	RXBVDD[9]	Y1	NC	AB24	RX[0]+	AE3	BSL_SCL
R5	RXVSS[9]	Y2	VSS	AB25	RX[0]–	AE4	EMB_SCL
R12	VDD2	Y3	VDD2	AB26	TX[1]–	AE5	ISTWI_ADDR[0]
R13	VSS	Y4	RXBVDD[11]	AC1	RX[11]–	AE6	LED_ACT[9]/
R14	VDD2	Y5	TXBVDD[11]	AC2	RX[11]+	AE7	VSS
R15	VSS	Y6	TXBVSS[11]	AC3	TX[11]+	AE8	VSS
R22	NC	Y7	TDIODE_N	AC4	TX[11]–	AE9	LED_FAULT[9]/
R23	TXVDD[2]	Y8	NC	AC5	VDDIO33	AE10	NC
R24	NC	Y19	VSS	AC6	NC	AE11	CLK
R25	VSS	Y20	TXVSS[0]	AC7	LED_ACT[10]/	AE12	VSS
R26	RXBVSS[2]	Y21	TXBVDD[0]	AC8	NC	AE13	VSS
T1	NC	Y22	TXVSS[1]	AC9	LED_FAULT[11]/	AE14	NC
T2	VSS	Y23	RXBVSS[0]	AC10	VSS	AE15	VSS
T3	VDD2	Y24	NC	AC11	VDDIO33	AE16	VSS
T4	VSS	Y25	VSS	AC12	LED_FAULT[6]/	AE17	LEDSYNCOUT
T5	TXVSS[9]	Y26	RXVDD[1]	AC13	REFCLK–	AE18	LED_STATUS[6]/
T22	NC	AA1	TX[10]+	AC14	NC	AE19	LED_STATUS[4]/
T23	NC	AA2	VSS	AC15	NC	AE20	VSS
T24	TX[2]–	AA3	VDD2	AC16	VSS	AE21	VSS
T25	TX[2]+	AA4	RXVSS[11]	AC17	VDDIO33	AE22	LED_ACT[2]/
T26	RXBVDD[2]	AA5	TXVSS[11]	AC18	NC	AE23	LED_FAULT[5]/
U1	TX[9]+	AA6	RTRIM	AC19	NC	AE24	GPIO[2]
U2	TXBVDD[9]	AA7	NC	AC20	NC	AE25	TXVDD[0]
U3	TXBVSS[9]	AA8	ISTWI_ADDR[1]	AC21	NC	AE26	VSS
U4	VDD2	AA9	NC	AC22	VSS	AF2	VSS
U5	RXBVSS[10]	AA18	LED_ACT[4]/	AC23	GPIO[3]	AF3	EMB_SD
U22	NC	AA19	NC	AC24	TX[0]–	AF4	EMB_DB_INT
U23	VSS	AA20	LED_FAULT[3]/	AC25	TX[0]+	AF5	LED_ACT[8]/
U24	TXVSS[2]	AA21	GPIO[1]	AC26	NC	AF6	LED_ACT[7]/
U25	TXBVSS[2]	AA22	VSS	AD1	RXVDD[11]	AF7	LED_ACT[6]/
U26	RXVSS[2]	AA23	RXBVDD[0]	AD2	TXVDD[11]	AF8	NC
V1	TX[9]–	AA24	TXVDD[1]	AD3	NC	AF9	LED_STATUS[11]/
V2	TXVDD[9]	AA25	NC	AD4	BSL_SD	AF10	NC
V3	NC	AA26	TX[1]+	AD5	EMB_CRC_INT	AF11	NC
V4	RXBVDD[10]	AB1	TX[10]–	AD6	LED_ACT[11]/	AF12	NC
V5	TXBVDD[10]	AB2	TXVDD[10]	AD7	VDDIO33	AF13	NC
V6	TXBVSS[10]	AB3	NC	AD8	VDDIO33	AF14	NC
V21	TXBVDD[1]	AB4	VSS	AD9	LED_FAULT[10]/	AF15	LEDSYNCIN
V22	RXVSS[1]	AB5	VDD2	AD10	LED_FAULT[7]/	AF16	LED_STATUS[10]/
V23	RXBVSS[1]	AB6	VSS	AD11	NC	AF17	LED_STATUS[9]/
V24	RX[1]+	AB7	NC	AD12	VDDIO33	AF18	LED_STATUS[8]/
V25	RX[1]–	AB8	NC	AD13	VDDIO33	AF19	LED_STATUS[2]/
V26	TXBVDD[2]	AB9	NC	AD14	RESET/	AF20	LED_STATUS[1]/
W1	RXVDD[10]	AB10	NC	AD15	VDDIO33	AF21	LED_ACT[5]/
W2	RX[10]–	AB11	LED_FAULT[8]/	AD16	VDDIO33	AF22	LED_ACT[3]/
W3	RX[10]+	AB12	NC	AD17	NC	AF23	LED_ACT[0]/
W4	RXVSS[10]	AB13	REFCLK+	AD18	LED_STATUS[5]/	AF24	LED_FAULT[4]/
W5	TXVSS[10]	AB14	NC	AD19	NC	AF25	LED_FAULT[2]/

1. *NC* pins are not connected inside of the package.
2. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy through the Phy Transmit Polarity and Phy Receiver Polarity registers.

Figure 5.2 LSISASx12 472–Pin BGA Top View[1]

	A2 IDDT	A3 NC	A4 MODE[1]	A5 CBL_ DET[11]/	A6 CBL_ DET[10]/	A7 CBL_ DET[9]/	A8 CBL_ DET[8]/	A9 NC	A10 NC	A11 NC	A12 NC	A13 NC
B1 VSS	B2 RX[6]–	B3 NC	B4 MODE[0]	B5 MODE[3]	B6 VSS	B7 VSS	B8 CBL_ DET[6]/	B9 NC	B10 PLLVDD	B11 VSS	B12 VSS	B13 NC
C1 TX[6]+	C2 RX[6]+	C3 SCANCLK3	C4 NC	C5 MODE[2]	C6 VDDIO33	C7 VDDIO33	C8 CBL_ DET[7]/	C9 NC	C10 PLLVSS	C11 VDDIO33	C12 VDDIO33	C13 NC
D1 TX[6]–	D2 NC	D3 RXVDD[6]	D4 SCANCLK1	D5 VSS	D6 NC	D7 NC	D8 NC	D9 NC	D10 VDDIO33	D11 VSS	D12 NC	D13 NC
E1 RXVDD[7]	E2 NC	E3 TXVDD[6]	E4 NC	E5 VDDIO33	E6 TN	E7 NC	E8 NC	E9 NC	E10 NC	E11 NC	E12 NC	E13 NC
F1 NC	F2 RX[7]+	F3 RX[7]–	F4 TXBVSS[6]	F5 VSS	F6 SCANCLK2	F7 NC	F8 NC	F9 NC				
G1 TX[7]+	G2 VSS	G3 NC	G4 TXVSS[6]	G5 RXBVSS[7]	G6 RXVSS[6]	G7 RXBVSS[6]	G8 VSS					
H1 TX[7]–	H2 VSS	H3 NC	H4 TXBVSS[7]	H5 RXB- VDD[7]	H6 TXBVDD[6]	H7 RXBVDD[6]						
J1 TXB- VDD[8]	J2 RXBVSS[8]	J3 TXVDD[7]	J4 TXVSS[7]	J5 TXBVDD[7]	J6 RXVSS[7]							
K1 TXBVSS[8]	K2 RXVSS[8]	K3 RXBVDD[8]	K4 VSS	K5 NC								
L1 NC	L2 RX[8]+	L3 RX[8]–	L4 NC	L5 NC								
M1 TXVSS[8]	M2 VSS	M3 NC	M4 NC	M5 NC							M12 VSS	M13 VDD2
N1 RX[9]–	N2 VSS	N3 NC	N4 NC	N5 RXVDD[8]							N12 VDD2	N13 VSS
P1 RX[9]+	P2 TX[8]+	P3 TX[8]–	P4 RXBVSS[9]	P5 TXVDD[8]							P12 VSS	P13 VDD2
R1 RXVDD[9]	R2 VSS	R3 VDD2	R4 RXBVDD[9]	R5 RXVSS[9]							R12 VDD2	R13 VSS
T1 NC	T2 VSS	T3 VDD2	T4 VSS	T5 TXVSS[9]								
U1 TX[9]+	U2 TXBVDD[9]	U3 TXBVSS[9]	U4 VDD2	U5 RXBVSS[10]								
V1 TX[9]–	V2 TXVDD[9]	V3 NC	V4 RXB- VDD[10]	V5 TXB- VDD[10]	V6 TXB- VSS[10]							
W1 RX- VDD[10]	W2 RX[10]–	W3 RX[10]+	W4 RXVSS[10]	W5 TXVSS[10]	W6 RXB- VSS[11]	W7 VSS						
Y1 NC	Y2 VSS	Y3 VDD2	Y4 RXB- VDD[11]	Y5 TXB- VDD[11]	Y6 TXB- VSS[11]	Y7 TDIODE_N	Y8 NC					
AA1 TX[10]+	AA2 VSS	AA3 VDD2	AA4 RXVSS[11]	AA5 TXVSS[11]	AA6 RTRIM	AA7 NC	AA8 ISTWI_ ADDR[1]	AA9 NC				
AB1 TX[10]–	AB2 TXVDD[10]	AB3 NC	AB4 VSS	AB5 VDD2	AB6 VSS	AB7 NC	AB8 NC	AB9 NC	AB10 NC	AB11 LED_ FAULT[8]/	AB12 NC	AB13 REF- CLK+
AC1 RX[11]–	AC2 RX[11]+	AC3 TX[11]+	AC4 TX[11]–	AC5 VDDIO33	AC6 NC	AC7 LED_ ACT[10]/	AC8 NC	AC9 LED_ FAULT[11]/	AC10 VSS	AC11 VDDIO33	AC12 LED_ FAULT[6]/	AC13 REF- CLK–
AD1 RXVDD[11]	AD2 TXVDD[11]	AD3 NC	AD4 BSL_SD	AD5 EMB_ CRC_INT	AD6 LED_ ACT[11]/	AD7 VDDIO33	AD8 VDDIO33	AD9 LED_ FAULT[10]/	AD10 LED_ FAULT[7]/	AD11 NC	AD12 VDDIO33	AD13 VDDIO33
AE1 NC	AE2 TDIODE_P	AE3 BSL_SCL	AE4 EMB_SCL	AE5 ISTWI_ ADDR[0]	AE6 LED_ ACT[9]/	AE7 VSS	AE8 VSS	AE9 LED_ FAULT[9]/	AE10 NC	AE11 CLK	AE12 VSS	AE13 VSS
	AF2 VSS	AF3 EMB_SD	AF4 EMB_ DB_INT	AF5 LED_ ACT[8]/	AF6 LED_ ACT[7]/	AF7 LED_ ACT[6]/	AF8 NC	AF9 LED_ STATUS[11]/	AF10 NC	AF11 NC	AF12 NC	AF13 NC

1. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy.

Figure 5.2 LSISASx12 472–Pin BGA Top View[1] (Cont.)

A14 NC	A15 NC	A16 CBL_DET[5]/	A17 NC	A18 NC	A19 NC	A20 NC	A21 NC	A22 NC	A23 NC	A24 NC	A25 VSS	
B14 VSS	B15 VSS	B16 CBL_DET[3]/	B17 NC	B18 NC	B19 VSS	B20 VSS	B21 NC	B22 NC	B23 NC	B24 NC	B25 NC	B26 TX[5]+
C14 NC	C15 NC	C16 CBL_DET[2]/	C17 NC	C18 NC	C19 VDDIO33	C20 VDDIO33	C21 NC	C22 NC	C23 NC	C24 RX[5]–	C25 RX[5]+	C26 TX[5]–
D14 CBL_DET[4]/	D15 CBL_DET[1]/	D16 NC	D17 VSS	D18 TDO	D19 TDI	D20 PROCMON	D21 SCAN_ENABLE	D22 VDDIO33	D23 NC	D24 RXVDD[5]	D25 NC	D26 TXVDD[5]
E14 NC	E15 NC	E16 CBL_DET[0]/	E17 TMS	E18 UART_RX	E19 TRST/	E20 TCK	E21 VSS	E22 NC	E23 VSS	E24 RX[4]–	E25 RX[4]+	E26 NC
				F18 UART_TX	F19 SCAN_MODE	F20 NC	F21 NC	F22 NC	F23 RXVSS[5]	F24 NC	F25 VSS	F26 RXVDD[4]
					G19 NC	G20 NC	G21 RXBVSS[5]	G22 RXBVDD[5]	G23 TXB-VDD[5]	G24 NC	G25 VSS	G26 TX[4]+
						H20 VSS	H21 TXBVSS[5]	H22 TXVSS[5]	H23 RXVSS[4]	H24 NC	H25 TXVDD[4]	H26 TX[4]–
							J21 RXBVSS[4]	J22 RXBVDD[4]	J23 TXB-VDD[4]	J24 RX[3]–	J25 RX[3]+	J26 NC
								K22 TXBVSS[4]	K23 VDD2	K24 RXBVSS[3]	K25 RXBVDD[3]	K26 RXVDD[3]
								L22 TXVSS[4]	L23 VSS	L24 VDD2	L25 VSS	L26 TX[3]+
M14 VSS	M15 VDD2							M22 RXVSS[3]	M23 TXBVDD[3]	M24 VDD2	M25 VSS	M26 TX[3]–
N14 VDD2	N15 VSS							N22 TXVSS[3]	N23 TXBVSS[3]	N24 RX[2]–	N25 RX[2]+	N26 NC
P14 VSS	P15 VDD2							P22 RXVDD[2]	P23 NC	P24 NC	P25 VSS	P26 TXVDD[3]
R14 VDD2	R15 VSS							R22 NC	R23 TXVDD[2]	R24 NC	R25 VSS	R26 RXBVSS[2]
								T22 NC	T23 NC	T24 TX[2]–	T25 TX[2]+	T26 RXBVDD[2]
								U22 NC	U23 VSS	U24 TXVSS[2]	U25 TXBVSS[2]	U26 RXVSS[2]
							V21 TXBVDD[1]	V22 RXVSS[1]	V23 RXBVSS[1]	V24 RX[1]+	V25 RX[1]–	V26 TXBVDD[2]
						W20 TXBVSS[0]	W21 RXVSS[0]	W22 TXBVSS[1]	W23 RXBVDD[1]	W24 NC	W25 VSS	W26 NC
					Y19 VSS	Y20 TXVSS[0]	Y21 TXBVDD[0]	Y22 TXVSS[1]	Y23 RXBVSS[0]	Y24 NC	Y25 VSS	Y26 RXVDD[1]
				AA18 LED_ACT[4]/	AA19 NC	AA20 LED_FAULT[3]/	AA21 GPIO[1]	AA22 VSS	AA23 RXB-VDD[0]	AA24 TXVDD[1]	AA25 NC	AA26 TX[1]+
AB14 NC	AB15 LED_STATUS[7]/	AB16 NC	AB17 LED_STATUS[3]/	AB18 LED_STATUS[0]/	AB19 NC	AB20 NC	AB21 LED_FAULT[1]/	AB22 VDDIO33	AB23 NC	AB24 RX[0]+	AB25 RX[0]–	AB26 TX[1]–
AC14 NC	AC15 NC	AC16 VSS	AC17 VDDIO33	AC18 NC	AC19 NC	AC20 NC	AC21 NC	AC22 VSS	AC23 GPIO[3]	AC24 TX[0]–	AC25 TX[0]+	AC26 NC
AD14 RESET/	AD15 VDDIO33	AD16 VDDIO33	AD17 NC	AD18 LED_STATUS[5]/	AD19 NC	AD20 VDDIO33	AD21 VDDIO33	AD22 LED_ACT[1]/	AD23 LED_FAULT[0]/	AD24 GPIO[0]	AD25 NC	AD26 RXVDD[0]
AE14 NC	AE15 VSS	AE16 VSS	AE17 LEDSYNC-OUT	AE18 LED_STATUS[6]/	AE19 LED_STATUS[4]/	AE20 VSS	AE21 VSS	AE22 LED_ACT[2]/	AE23 LED_FAULT[5]/	AE24 GPIO[2]	AE25 TXVDD[0]	AE26 VSS
AF14 NC	AF15 LEDSYNC-IN	AF16 LED_STATUS[10]/	AF17 LED_STATUS[9]/	AF18 LED_STATUS[8]/	AF19 LED_STATUS[2]/	AF20 LED_STATUS[1]/	AF21 LED_ACT[5]/	AF22 LED_ACT[3]/	AF23 LED_ACT[0]/	AF24 LED_FAULT[4]/	AF25 LED_FAULT[2]/	

1. Boot load options configure the polarity of the RX+/RX– signals and TX+/TX– signals for each phy.

5.4 Package Diagram

The LSISASx12 is packaged in a 472-EPBGA-T package with a 27 mm x 27 mm footprint and 1.0 mm ball pitch. The package code is UO. The package drawing number is JZ02-000015-00. Figure 5.3 provides the package diagram for the LSISASx12.

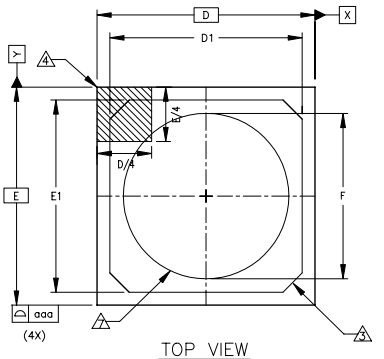
Figure 5.3 472-Pin EPBGA-T (UO) Mechanical Drawing (Sheet 1 of 3)

REVISIONS

APPROVAL	DATE	DESCRIPTION	REV
C.FISHLEY	01–24–02	INITIAL RELEASE	0
C.FISHLEY	03–20–02	REVISE DIMENSION (ddd)	1
R.MENDOZA	01–29–03	ADD PACKAGE (4E)	2
C.FISHLEY	06–04–03	ADD PACKAGE (6i)	3
C.FISHLEY	06–25–03	ADD PACKAGE (61)	4
WK Liew	10–06–03	ADD PACKAGE (4X) (5E) (HT)	5
L. MORA	03–24–04	ADD PACKAGE (5Y)	6

REVISIONS

APPROVAL	DATE	DESCRIPTION	REV
WK Liew	07–02–04	ADD PACKAGE (8C)	7
C. Fishley	08–13–04	ADD PACKAGE (uo, 3w, ow, 82)	8
L. Mora	01–24–05	ADD PACKAGE (gz)	9
L. Mora	05–26–05	Changed "F" min from 19.5 to 17.5.	10
L. Mora	08–22–05	Add Package (aa2au)	11



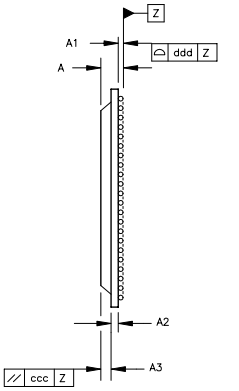


TABLE 1: COMMON DIMENSIONS

SYM	MIN	NOM	MAX
A	–	–	2.50
A1	0.40	–	–
A2	0.25	–	0.70
A3	–	–	1.25
b	0.50	–	0.70
D		27.00 BSC	
D1	23.00	–	25.00
E		27.00 BSC	
E1	23.00	–	25.00
e		1.00 BSC	
F	17.5	–	21.50

TABLE 2: TOLERANCES OF FORM AND POSITION

DIMENSION	e = 1.00
aaa	0.20
ccc	0.35
ddd	0.15
eee	0.25
fff	0.10

NOTES:

1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M–1994.
2. ALL DIMENSIONS IN MM.
3. CORNER DETAIL IS LSI LOGIC CORPORATION OPTION.
4. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL, AND MAY CONSIST OF INK, LASER MARK OR METALLIZED MARKING, BUT MUST BE LOCATED WITHIN THE ZONE INDICATED
5. JEDEC CODE MS–034.
6. PRIMARY DATUM Z AND SEATING PLANE ARE DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
7. HEAT SPREADER OUTLINE.

LSI LOGIC CORPORATION
1551 McCARTHY BLVD. MILPITAS, CA. 95035

TITLE: 27x27mm EPBGA–T, EPBGA–HP
1.00 BALL PITCH

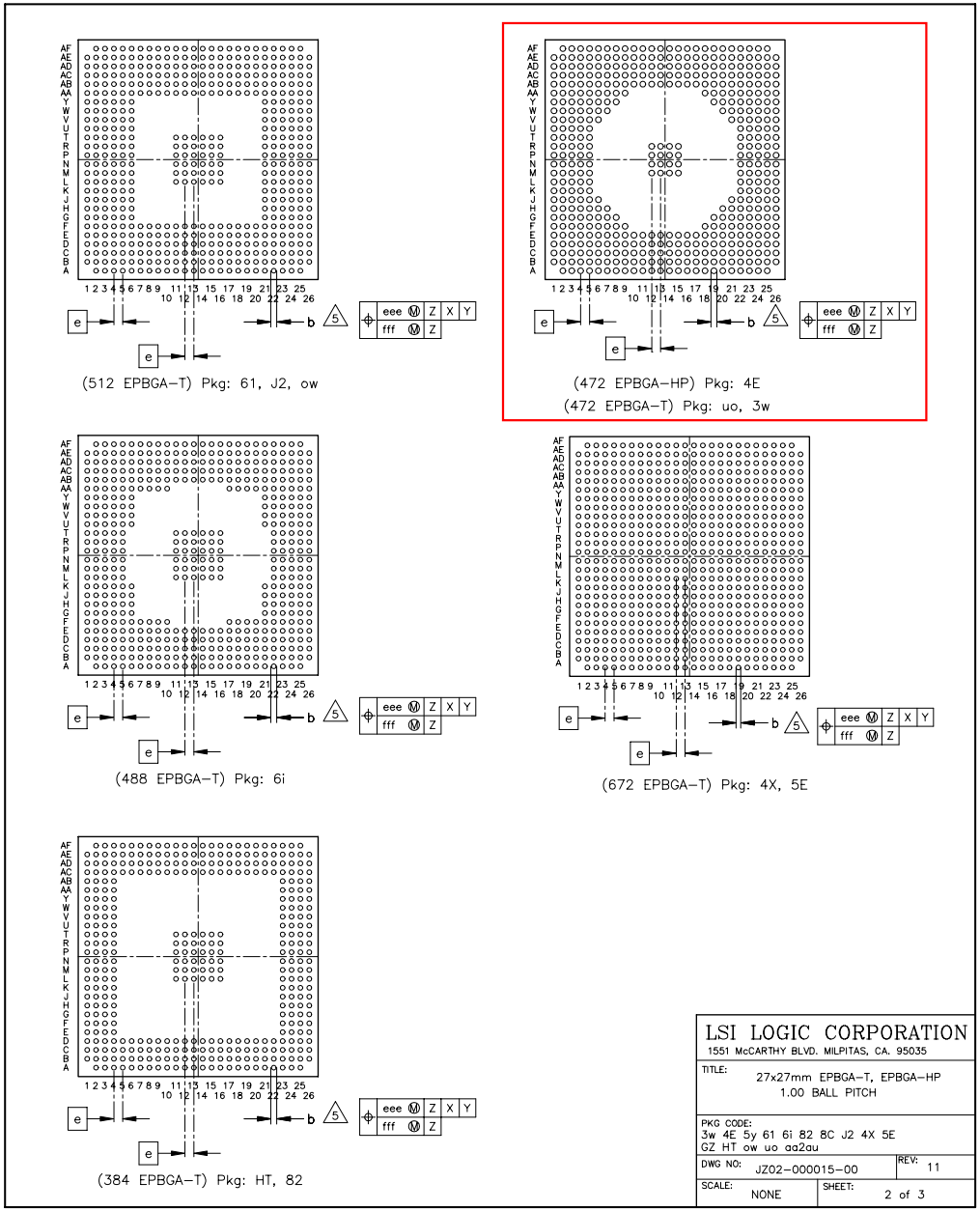
PKG CODE: 3w 4E 5y 61 6i 82 8C J2 4X 5E GZ HT ow uo aa2au

DWG NO: JZ02–000015–00 REV: 11

SCALE: NONE SHEET: 1 of 3

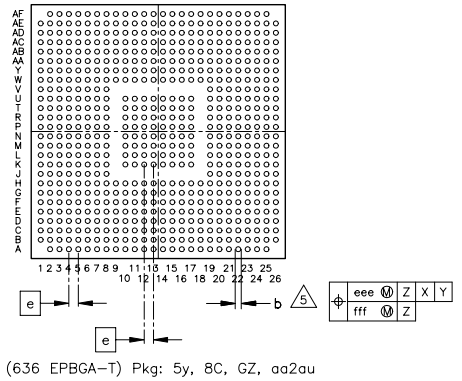
Important: **For board layout and manufacturing, obtain the most recent engineering drawings from your LSI Logic marketing representative by requesting the outline drawing for package code UO.**

Figure 5.3 472-Pin EPBGA-T (UO) Mechanical Drawing (Sheet 2 of 3) (Bottom View)



Important: **For board layout and manufacturing, obtain the most recent engineering drawings from your LSI Logic marketing representative by requesting the outline drawing for package code UO.**

Figure 5.3 472-Pin EPBGA-T (UO) Mechanical Drawing (Sheet 3 of 3) (Bottom View)



(636 EPBGA–T) Pkg: 5y, 8C, GZ, aa2au

LSI LOGIC CORPORATION 1551 McCARTHY BLVD. MILPITAS, CA. 95035	
TITLE: 27x27mm EPBGA–T, EPBGA–HP 1.00 BALL PITCH	
PKG CODE: 3w 4E 5y 61 6i 82 8C J2 4X 5E GZ HT ow uo aa2au	
DWG NO: JZ02–000015–00	REV: 11
SCALE: NONE	SHEET: 3 of 3

Important: **For board layout and manufacturing, obtain the most recent engineering drawings from your LSI Logic marketing representative by requesting the outline drawing for package code UO.**

Appendix A
Register Summary

Table A.1 through Table A.9 provide a register summary.

Table A.1 Configuration Manager Config Register Map

Register Name	Offset	Page
LSISASx12 Expander SAS Address High	0x0000	4-6
LSISASx12 Expander SAS Address Low	0x0004	4-6
Vendor Identifier High	0x0008	4-7
Vendor Identifier Low	0x000C	4-8
Product Identifier 3	0x0010	4-8
Product Identifier 2	0x0014	4-9
Product Identifier 1	0x0018	4-9
Product Identifier 0	0x001C	4-10
Product Revision	0x0020	4-10
Component Vendor ID High	0x0024	4-11
Component Vendor ID Low	0x0028	4-11
Component ID and Revision	0x002C	4-11
Vendor Specific Dword 1	0x0030	4-12
Vendor Specific Dword 0	0x0034	4-12
Report General 1	0x0038	4-13
Report General 2	0x003C	4-13
Spin-up Control	0x0040	4-14
Phy Configuration	0x0044	4-15

Table A.1 Configuration Manager Config Register Map (Cont.)

Register Name	Offset	Page
Phy Transmit Polarity	0x004C	4-15
Phy Receiver Polarity	0x0050	4-16
Boot Control and Status	0x0090	4-16
Reserved	–	–
API2C Global Control	0x4000	4-17
API2C Interrupt Status	0x4008	4-18
API2C Interrupt Enable	0x4010	4-19
API2C Wait Timer Control	0x4018	4-19
API2C $t_{TIMEOUT}$ Control	0x4020	4-20
API2C t_{LOW} Control	0x4028	4-21
API2C Timer Clock Divider Control	0x4038	4-22
API2C Monitor	0x4040	4-23
API2C Soft Reset	0x4048	4-24
API2C Master Command	0x4050	4-25
API2C Receive Transfer Length	0x4058	4-26
API2C Transmit Transfer Length	0x4060	4-27
API2C Address Register 1	0x4068	4-28
API2C Address Register 2	0x4070	4-29
API2C Data	0x4078	4-29
API2C Transmit FIFO Status	0x4080	4-30
API2C Receive FIFO Status	0x4088	4-30
API2C Master Interrupt Enable	0x4090	4-31
API2C Master Interrupt Status	0x4098	4-32
API2C Transmit Bytes Transferred	0x40A0	4-35
API2C Receive Bytes Transferred	0x40A8	4-36

Table A.1 Configuration Manager Config Register Map (Cont.)

Register Name	Offset	Page
API2C SCL High Period	0x4100	4-36
API2C SCL Low Period	0x4108	4-37
API2C Spike Filter Control	0x4110	4-38
API2C SDA Setup Time	0x4118	4-39
API2C SDA Hold Time	0x4120	4-40
Reserved	–	–
LED Group Control	0x8000	4-41
Group 0 GPIO Control	0x8004	4-42
Group 1 GPIO Control	0x8008	4-43
Group 2 GPIO Control	0x800C	4-43
Group 3 GPIO Control	0x8010	4-44
Group 4 GPIO Control	0x8014	4-44
Group 0 GPIO Value	0x8018	4-45
Group 1 GPIO Value	0x801C	4-45
Group 2 GPIO Value	0x8020	4-46
Group 3 GPIO Value	0x8024	4-46
Group 4 GPIO Value	0x8028	4-47
LED Blinker Definition 1	0x8030	4-47
LED Blinker Definition 2	0x8034	4-48
LED Blinker Definition 3	0x8038	4-49
Group 0 Override	0x8040	4-50
Group 1 Override	0x8048	4-51
Group 2 Override	0x8050	4-52
Group 3 Override	0x8058	4-53
Group 4 Override	0x8060	4-54

Table A.1 Configuration Manager Config Register Map (Cont.)

Register Name	Offset	Page
Reserved	–	–
SIO Configuration	0xC000	4-55
SIO Receive 0	0xC004	4-57
SIO Receive 1	0xC008	4-58
SIO General Purpose Receive 0	0xC00C	4-58
SIO General Purpose Receive 1	0xC010	4-59
SIO Output Data Control 0	0xC014	4-59
SIO Output Data Control 1	0xC018	4-63
SIO Output Data Control 2	0xC01C	4-64
SIO General Purpose Output Data 0	0xC020	4-66
SIO General Purpose Output Data 1	0xC024	4-66
Reserved	–	–
SIO Pattern Definition 0	0xC040	4-67
SIO Pattern Definition 1	0xC044	4-67
SIO Pattern Definition 2	0xC048	4-68
SIO Pattern Definition 3	0xC04C	4-69
Activity Stretch Control	0xC050	4-69
SIO Adapter Control	0xC054	4-71

Table A.2 SMP Target Register Map

Register Name	Offset	Page
LSISASx12 Expander SAS Address High	0x00000	4-73
LSISASx12 Expander SAS Address Low	0x00004	4-73
Remote Routing Table Configuration	0x00020–0x0004C	4-73

Table A.3 SPhynx Link Register Map

Register Name	Offset	Page
Identify Address Information	0x0004	4-76
Identify SAS Address High	0x0010	4-78
Identify SAS Address Low	0x0014	4-78
Discover Information 1	0x0044	4-78
Discover Information 2 – SAS Address High	0x0048	4-80
Discover Information 3 – SAS Address Low	0x004C	4-80
Discover Information 4 – Attached SAS Address High	0x0050	4-81
Discover Information 5 – Attached SAS Address Low	0x0054	4-81
Discover Information 6 – Attached Phy Identifier	0x0058	4-81
Discover Information 7 – Reserved	0x005C	4-82
Discover Information 8 – Link Rates	0x0060	4-82
Discover Information 9 – Routing Attributes	0x0064	4-83
Discover Information 10 – Vendor Specific	0x0068	4-83
Report SATA Phy ID	0x0070	4-84
SATA Target Address High	0x0078	4-84
SATA Target Address Low	0x007C	4-85
SATA Dev2Host FIS0	0x0080	4-85
SATA Dev2Host FIS1	0x0084	4-86
SATA Dev2Host FIS2	0x0088	4-86
SATA Dev2Host FIS3	0x008C	4-87
Affiliated STP Initiator Address High	0x0098	4-87
Affiliated STP Initiator Address Low	0x009C	4-88
Phy Control Link Rates	0x00A0	4-88

Table A.3 SPhynx Link Register Map (Cont.)

Register Name	Offset	Page
Phy Control PPTOV	0x00A4	4-89
Connection Information	0x00AC	4-89
Broadcast Change High	0x00C0	4-90
Broadcast Change Low	0x00C4	4-90
Broadcast SES High	0x00C8	4-91
Broadcast SES Low	0x00CC	4-91
SATA Nexus Loss Timeout	0x0108	4-92
SATA Connection Mode	0x0120	4-93

Table A.4 Phy Register Map

Register Name	Offset	Page
Reserved	–	–
SMP Command	0x0010	4-94
Reserved	–	–
Error Log Invalid Word	0x0018	4-95
Error Log Display Error	0x001C	4-95
Error Log Loss of Sync	0x0020	4-96
Error Log Reset Sequence Fail	0x0024	4-96
Test Register	0x0028	4-97
Reserved	–	–
Custom Jitter	0x0034	4-98
Reserved	–	–

Table A.5 STP Target Register Map

Register Name	Offset	Page
Reserved	–	–
Expander SAS Address High	0x0010	4-101
Expander SAS Address Low	0x0014	4-101
Reserved	–	–
Discover Information	0x0044	4-101
Discover Information 2 – SAS Address High	0x0048	4-103
Discover Information 3 – SAS Address Low	0x004C	4-104
Discover Information 4 – Attached SAS Address High	0x0050	4-104
Discover Information 5 – Attached SAS Address Low	0x0054	4-104
Discover Information 6 – Attached Phy Identifier	0x0058	4-105
Discover Information 7 – Reserved	0x005C	4-105
Discover Information 8 – Link Rates	0x0060	4-105
Discover Information 9 – Routing Attributes	0x0064	4-106
Discover Information 10 – Vendor Specific	0x0068	4-107
Reserved	–	–
Phy SATA Phy ID	0x0070	4-107
Reserved	–	–
STP Target SAS Address High	0x0078	4-108
STP Target SAS Address Low	0x007C	4-108
SATA Dev2Host Reset FIS0	0x0080	4-108
SATA Dev2Host Reset FIS1	0x0084	4-109
SATA Dev2Host Reset FIS2	0x0088	4-110
SATA Dev2Host Reset FIS3	0x008C	4-110

Table A.5 STP Target Register Map (Cont.)

Register Name	Offset	Page
Reserved	–	–
Affiliated STP Initiator Address High	0x0098	4-111
Affiliated STP Initiator Address Low	0x009C	4-111
Phy Control Link Rates	0x00A0	4-111
Phy Control PPTOV	0x00A4	4-112
Reserved	–	–
Connection Information	0x00AC	4-112
Reserved	–	–
SATA Nexus Loss Timeout	0x0108	4-113
Reserved	–	–
Last H2D Register FIS Received Dword 0	0x0200	4-115
Last H2D Register FIS Received Dword 1	0x0204	4-115
Last H2D Register FIS Received Dword 2	0x0208	4-116
Last H2D Register FIS Received Dword 3	0x020C	4-116
Reserved	–	–
STP Connection Control	0x0220	4-117
Reserved	–	–
SMP Phy Operation	0x0310	4-117
Reserved	–	–
Error Log – Invalid DWord	0x0318	4-118
Error Log – I^2C CRC Error Count	0x031C	4-119
Error Log – Unexpected SYNC Count	0x0320	4-120
Error Log – R_ERR Transmitted/Received Count	0x0324	4-120

Table A.5 STP Target Register Map (Cont.)

Register Name	Offset	Page
Reserved	0x328	–
STP Error Status	0x032C	4-121
Reserved	0x330–3FF	–

Table A.6 ECM Phy Register Map

Register Name	Offset	Page
Phy 0 ECM Config Register	0x00000	4-127
Phy 0 Remote Bank Enable	0x00004	4-128
Reserved	0x00008–0x0000C	–
Phy 1 ECM Config Register	0x00010	4-127
Phy 1 Remote Bank Enable	0x00014	4-128
Reserved	0x00018–0x0001C	–
Phy 2 ECM Config Register	0x00020	4-127
Phy 2 Remote Bank Enable	0x00024	4-128
Reserved	0x00028–0x0002C	–
Phy 3 ECM Config Register	0x00030	4-127
Phy 3 Remote Bank Enable	0x00034	4-128
Reserved	0x00038–0x0003C	–
Phy 4 ECM Config Register	0x00040	4-127
Phy 4 Remote Bank Enable	0x00044	4-128
Reserved	0x00048–0x00042C	–
Phy 5 ECM Config Register	0x00050	4-127
Phy 5 Remote Bank Enable	0x00054	4-128
Reserved	0x00058–0x0005C	–
Phy 6 ECM Config Register	0x00060	4-127

Table A.6 ECM Phy Register Map (Cont.)

Register Name	Offset	Page
Phy 6 Remote Bank Enable	0x00064	4-128
Reserved	0x00068–0x0006C	–
Phy 7 ECM Config Register	0x00070	4-127
Phy 7 Remote Bank Enable	0x00074	4-128
Reserved	0x00078–0x0007C	–
Phy 8 ECM Config Register	0x00080	4-127
Phy 8 Remote Bank Enable	0x00084	4-128
Reserved	0x00088–0x0008C	–
Phy 9 ECM Config Register	0x00090	4-127
Phy 9 Remote Bank Enable	0x00094	4-128
Reserved	0x00098–0x0009C	–
Phy 10 ECM Config Register	0x000A0	4-127
Phy 10 Remote Bank Enable	0x000A4	4-128
Reserved	0x000A8–0x000AC	–
Phy 11 ECM Config Register	0x000B0	4-127
Phy 11 Remote Bank Enable	0x000B4	4-128
Reserved	0x00058–0x0005C	–
STP Target ECM Config Register	0x000C0	4-127
STP Target Remote Bank Enable	0x000C4	4-128
Reserved	0x00058–0x0005C	–
SMP Target ECM Config Register	0x000D0	4-127
SMP Target Remote Bank Enable	0x000D4	4-128
Reserved	0x000D8–0x01FFF	–
Global Config	0x02000	4-129

Table A.7 provides the offsets of each remote bank in the expander connection manager register map. The register space for each Remote Bank [NN] Config is a 96-byte space with the format defined in Table A.8. The format and register definitions for each remote bank are identical with the exception of a differing offset.

Table A.7 ECM Remote Bank Register Map

Register Name	Offset	Page
Remote Bank 00 Config	0x10000	4-129
Remote Bank 01 Config	0x100C0	
Remote Bank 02 Config	0x10180	
Remote Bank 03 Config	0x10240	
Remote Bank 04 Config	0x10300	
Remote Bank 05 Config	0x103C0	
Remote Bank 06 Config	0x10480	
Remote Bank 07 Config	0x10540	
Remote Bank 08 Config	0x10600	
Remote Bank 09 Config	0x106C0	
Remote Bank 10 Config	0x10780	
Remote Bank 11 Config	0x10840	
Remote Bank 12 Config	0x10900	
Reserved	0x109C0–0x1FFFF	

Table A.8 ECM Remote Bank Register Map

Register Name	Offset	Page
Remote Bank NN SAS Address High 0	0x00	4-132
Remote Bank NN SAS Address Low 0	0x04	4-132
Remote Bank NN Config 0	0x08	4-133
Reserved	0x0C	–

Table A.8 ECM Remote Bank Register Map (Cont.)

Register Name	Offset	Page
Remote Bank NN SAS Address High 1	0x10	4-132
Remote Bank NN SAS Address Low 1	0x14	4-132
Remote Bank NN Config 1	0x18	4-133
Reserved	0x1C	–
Remote Bank NN SAS Address High 2	0x20	4-132
Remote Bank NN SAS Address Low 2	0x24	4-132
Remote Bank NN Config 2	0x28	4-133
Reserved	0x2C	–
Remote Bank NN SAS Address High 3	0x030	4-132
Remote Bank NN SAS Address Low 3	0x34	4-132
Remote Bank NN Config 3	0x38	4-133
Reserved	0x3C	–
Remote Bank NN SAS Address High 4	0x40	4-132
Remote Bank NN SAS Address Low 4	0x44	4-132
Remote Bank NN Config 4	0x048	4-133
Reserved	0x4C	–
Remote Bank NN SAS Address High 5	0x50	4-132
Remote Bank NN SAS Address Low 5	0x54	4-132
Remote Bank NN Config 5	0x58	4-133
Reserved	0x5C	–
Remote Bank NN SAS Address High 6	0x60	4-132
Remote Bank NN SAS Address Low 6	0x64	4-132
Remote Bank NN Config 6	0x68	4-133
Reserved	0x6C	–
Remote Bank NN SAS Address High 7	0x70	4-132

Table A.8 ECM Remote Bank Register Map (Cont.)

Register Name	Offset	Page
Remote Bank NN SAS Address Low 7	0x74	4-132
Remote Bank NN Config 7	0x78	4-133
Reserved	0x7C	–
Remote Bank NN SAS Address High 8	0x80	4-132
Remote Bank NN SAS Address Low 8	0x84	4-132
Remote Bank NN Config 8	0x88	4-133
Reserved	0x8C	–
Remote Bank NN SAS Address High 9	0x90	4-132
Remote Bank NN SAS Address Low 9	0x94	4-132
Remote Bank NN Config 9	0x98	4-133
Reserved	0x9C	–
Remote Bank NN SAS Address High 10	0xA0	4-132
Remote Bank NN SAS Address Low 10	0xA4	4-132
Remote Bank NN Config 10	0xA8	4-133
Reserved	0xAC	–
Remote Bank NN SAS Address High 11	0xB0	4-132
Remote Bank NN SAS Address Low 11	0xB4	4-132
Remote Bank NN Config 11	0xB8	4-133
Reserved	0xBC	–

.

Table A.9 EMB Slave Register Map

Register Name	Offset	Page
Data FIS RAM	0x000–0x41F	–
Reserved	0x420–0x4FF	–
STP - SEP Doorbell	0x500	4-134
SEP - STP Doorbell	0x501	4-137
STP - SEP Transfer Length	0x502	4-138
SEP - STP Transfer Length	0x503	4-138
Transfer Attributes	0x504	4-139
Command Status	0x505	4-139
Error Detection Control	0x506	4-140
Error Status	0x507	4-141
Actual CRC	0x508	4-142
Expected CRC	0x509	4-143
Reserved	0x508–0x5FF	–
RR_Command0 - RR_Command3	0x600 - 0x603	4-143
RR_Data0 - RR_Data7	0x604 - 0x60B	4-144
RR_Control	0x60C	4-144
Reserved	0x60D–0xFFF	–

Appendix B Example Code for API2C Interface and Serial EEPROM

The following code examples demonstrate how to initialize the API2C interface, read from the serial EEPROM, and write to the serial EEPROM. The read and write examples perform a 32-byte page read and 32-byte page write. This code assumes the use of a serial EEPROM that supports 16-bit addressing and is able to write a 32-byte page.

B.1 Initialize API2C Interface

This section provides example code for initializing the API2C interface.

```
// reset
write(SOFT_RESET, 0x1);
do {
   temp = read(SOFT_RESET);
} while (temp & 0x1);

// Flush TX FIFO
write(MASTER_COMMAND, 0x7);
write(MASTER_COMMAND, 0xF);

// Enable Master
write(GLOBAL_CONTROL, 0x1);

// Enable Master Interrupt
write(INTERRUPT_ENABLE, 0x1);

// Enable individual interrupt sources
write(MASTER_INTERRUPT_ENABLE, 0xFFF0);

// Set the clock timings
write(SCL_HIGH_PERIOD, 0x0176);
write(SCL_LOW_PERIOD, 0x0176);
write(SDA_SETUP_TIME, 0x002D);
write(SDA_HOLD_TIME, 0x0026);
write(SPIKE_FILTER_CONTROL, 0x0005);
```

```
write(WAIT_TIMER, 0xCE4D);
write(TIMER_CLOCK_DIVIDER_CONTROL, 0x000D);

// Flush any data in the RX FIFO
depth = read(RX_FIFO_STATUS);
for (i=0; i<depth; i++)
{
   read(DATA);
}1
```

B.2 Write a 32-Byte Page to the Serial EEPROM

This section provides example code for writing a 32-byte page to the serial EEPROM.

```
// transferring a total of 34 (decimal) bytes
// 2 for 16-bit serial EEPROM address plus 32 bytes
// of data
Write(TRANSMIT_TRANSFER_LENGTH, 34);
Write(RECEIVE_TRANSFER_LENGTH, 0x00);

// Control byte for serial EEPROM
Write(ADDRESS_REGISTER_1, 0xA0);

// set to automatic transfer mode
Write(MASTER_COMMAND, 0x01);
Write(MASTER_COMMAND, 0x09);

// write upper EEPROM address, this example is writing
// to address 0x00 in serial EEPROM
Write(DATA, 0x00);

// write lower EEPROM address, this example is writing
// to address 0x00 in serial EEPROM
Write(DATA, 0x00);

bytesLeft = 32;
while (bytesLeft) {
   depth = Read(TRANSMIT_FIFO_STATUS);
   while (depth > 0 && bytesLeft) {
          Write(DATA, *ptrData);
          ptrData++;
          bytesLeft--;
          depth--;
   }
```

```
}

// wait for Transfer Stopped Interrupt
Do {
    temp = Read(MASTER_INTERRUPT_STATUS);
} while ((temp & 0x08) == 0x00);

// clear interrupt
Write(INTERRUPT_STATUS, 0x1);
```

B.3 Read a 32-Byte Page from the Serial EEPROM

This section provides example code for reading a 32-byte page from the serial EEPROM.

```
// write 2 bytes to select eeprom address
Write(TRANSMIT_TRANSFER_LENGTH, 2);
Write(RECEIVE_TRANSFER_LENGTH, 0x00);

// Control byte for serial EEPROM
Write(ADDRESS_REGISTER_1, 0xA0);

// set to automatic transfer mode
Write(MASTER_COMMAND, 0x01);
Write(MASTER_COMMAND, 0x09);

// write upper EEPROM address, this example is writing
// to address 0x00 in serial EEPROM
Write(DATA, 0x00);

// write lower EEPROM address, this example is writing
// to address 0x00 in serial EEPROM
Write(DATA, 0x00);

// wait for Transfer Stopped Interrupt
Do {
    temp = Read(MASTER_INTERRUPT_STATUS);
} while ((temp & 0x08) == 0x00);

// clear interrupt
Write(INTERRUPT_STATUS, 0x1);

// read 32 bytes
Write(TRANSMIT_TRANSFER_LENGTH, 0x00);
```

```
Write(RECEIVE_TRANSFER_LENGTH, 32);

// Control byte for serial EEPROM and set read bit
Write(ADDRESS_REGISTER_1, 0xA1);

// set to automatic transfer mode
Write(MASTER_COMMAND, 0x01);
Write(MASTER_COMMAND, 0x09);

bytesLeft = 32;
while(bytesLeft)
   depth = Read(RECEIVE_FIFO_STATUS);
   for (i=0; i<depth; i++) {
          temp = Read(DATA);
          *ptrData = (temp & 0xFF);
          ptrData++;
          bytesLeft--;
   }
}

// wait for Transfer Stopped Interrupt
Do {
   temp = Read(MASTER_INTERRUPT_STATUS);
} while ((temp & 0x08) == 0x00);

// clear interrupt
Write(INTERRUPT_STATUS, 0x1);
```

Index

D

E

F

G

I

J

L

M

N

O

P

R

S

T

U

V

W

Customer Feedback

We would appreciate your feedback on this document. Please copy the following page, add your comments, and fax it to us at the number shown.

If appropriate, please also fax copies of any marked-up pages from this document.

Important: Please include your name, phone number, fax number, and company address so that we may contact you directly for clarification or additional information.

Thank you for your help in improving the quality of our documents.

Reader's Comments

Fax your comments to: LSI Logic Corporation
Technical Publications
M/S AF-198
Fax: 408.433.4333

Please tell us how you rate this document: *LSISASx12/LSISASx12A 3.0 Gbit/s SAS/SATA Expander Technical Manual, Version 3.0.* Place a check mark in the appropriate blank for each category.

	Excellent	Good	Average	Fair	Poor
Completeness of information	____	____	____	____	____
Clarity of information	____	____	____	____	____
Ease of finding information	____	____	____	____	____
Technical content	____	____	____	____	____
Usefulness of examples and illustrations	____	____	____	____	____
Overall manual	____	____	____	____	____

What could we do to improve this document?

If you found errors in this document, please specify the error and page number. If appropriate, please fax a marked-up copy of the page(s).

Please complete the information below so that we may contact you directly for clarification or additional information.

Name _____ Date _____

Telephone _____ Fax _____

Title _____

Department _____ Mail Stop _____

Company Name _____

Street _____

City, State, Zip _____