IBM Packet Routing Switch PRS64G

Datasheet

IBM Microelectronics Division
1580 Route 52, Bldg. 504
Hopewell Junction, NY 12533-6351

The IBM home page can be found at http://www.ibm.com
The IBM Microelectronics Division home page can be found at http://www.chips.ibm.com

prs64_title.fm.01
August 31, 2000

Contents

List of Figures

List of Tables

1. General Information

1.1 Features

- Non-blocking, self-routing, single-stage switch
- 32 input ports and 32 output ports
- High Performance:
  - 1.6 Gb/s to 2 Gb/s throughput per port
  - Up to 64 Gb/s single device aggregate throughput
- Speed Expansion:
  - Internal: Support doubling port speed, (reducing the number of ports to 16)
  - External: Up to 128 Gb/s aggregate throughput (with two devices)
  - Combination of internal and external speed expansion provides 16 ports at 8 Gb/s (with two devices)
- Serial data communication from 400 to 500 Mb/s, compliant with the EIA/JEDEC JESD8-6 standard.
- Multicast support without packet duplication in shared memory
- Dynamically shared output buffer (1024 packets of 64 to 80 bytes)
- Configurable number of traffic priorities (one to four) with programmable output queue thresholds and shared memory thresholds
- Physical Port Bundling by four, with internal or external speed expansion to define logical port at 16 Gb/s
- Packet lossless scheduled switchover facility
- Serial processor interface (Serial Host Interface)
- Configurable packet lengths of:
  - 32 to 40 bytes (increment of two)
  - 64 to 80 bytes (increment of four)
  - 128 to 160 (increment of eight) with external speed expansion only
- Packet header of two to five bytes, containing destination bit map, packet priority, and switch redundancy support information, all protected by a parity bit
- Shared output buffer with total capacity of:
  - 1024 packets (64 bytes) for a single chip
  - 2048 packets (64 bytes) with external speed expansion (with two devices)
- Reception on any input port of Control Packets destined to the local processor
- Transmission of Control Packets from the local processor to any output port
- Detection of link liveness by reception of specific packets
- Programmable byte shuffling in outgoing packets
- CMOS6SF (SA-12E) technology (Ldrawn=0.25 µm, Leff=0.18 µm): 2.5 V compliant LVCMOS compatible I/O for low speed signals
- IEEE 1149.1 standard boundary scan to facilitate circuit board testing

1.2 Description

The IBM Packet Routing Switch PRS64G, one of a family of second generation switching devices designed for high performance, non-blocking fixed length packet switching, enables development of scalable switch fabrics of aggregate bandwidth from 64 Gb/s to 512 Gb/s.

The switch receives packets on 32 input ports and routes them to one or more of 32 output ports based on bit map information carried in the packet header. Each port operates at two Gb/s, resulting in a single device throughput of 64 Gb/s. This data speed is achieved by implementing, in one device, two 32 x 32 sub-switch elements, running at one Gb/s per port and organized internally in speed expansion mode. In addition, 500 Mb/s serial data communication provides, over two differential pairs, the necessary bit rate per island.

Quality of service support is provided through four levels of packet priority. The architecture supports flow control, based on a grant mechanism, and provides programmable thresholds, one per priority.

Scalability of speed is achieved via external speed expansion. Two devices operate in parallel (one as master, the other as slave) to form a 32 x 32 switch core at 4 Gb/s per port. Scalability in port speed is achieved through link paralleling by combining four physical ports together into a logical port of 16 Gb/s. Scalability of ports can be provided by single stage port expansion, which allows the number of ports on the switch fabric to be increased.

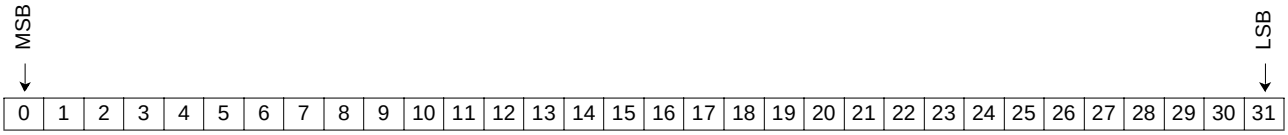
No synchronization is required between input ports. However, packets on a given port are always received or transmitted at a fixed rate according to the packet length.

Table 1: Ordering Information

Part Number	Description	Bandwidth	Throughput
IBM32SW0640DSLCDA250	IBM Packet Routing Switch	64-512 Gb/s	64 Gb/s

1.3 Conventions

Throughout this document, standard IBM notation is used: bits and bytes are numbered in ascending order from left to right. Thus the Most Significant Bit (MSB) has the lowest number and the Least Significant Bit (LSB) has the highest number:



Notation for bit encoding is as follows:

- Hexadecimal values are preceded by x and appear in single quotation marks. For example: x'0B00'.
- Binary values in text are either spelled out (zero and one) or appear in single quotation marks For example: '1010'.
- Binary values in the Default and Description columns of the register sections have no special marking, but are separated from text as in this example:

  0 No action on read access

  1 Auto-reset interrupt request register upon read access

The use of overbars, for example $\overline{\text{DDEL\_OUT}}$, designates signals that are active low.

Note: By default, the bits are positive active.

1.3.1 Definition of Terms

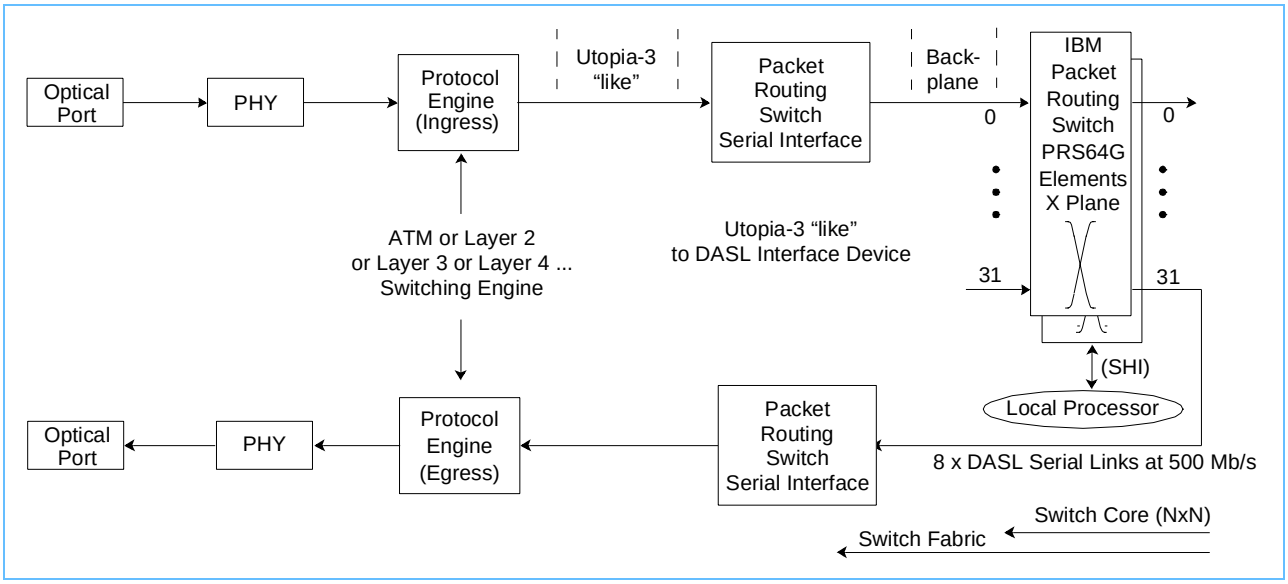
BIST: Built in self test

Local Processor: The microprocessor connected to PRS64G through the Serial Host Interface (SHI).

Logical Unit (LU): The part of the packet that is processed by one island. One packet is made of several LUs of equal length (2, 4, 8).

Packet: The user unit information element made of multiple LUs (header + payload).

Figure 1: System View with IBM Packet Switch Serial Interface



The PRS64G enables the construction of non-blocking scalable switch fabrics through repeated instances of the same switch element. It is designed for a wide variety of applications including: campus, WAN edge, access, and backbone switches. The PRS64G along with the IBM Packet Routing Switch Serial Interface provides a complete redundant switch fabric for the attachment of a 32-bit wide interface to any protocol engine. This supports protocols such as Packet Over Sonet (POS), Gigabit Ethernet Multilayer switching, and ATM.

Similarly, an IBM PowerNP NP4GS3 can attach directly through the DASL links to the PRS64G, allowing a very compact and powerful switching system. For more information regarding the NP4GS3, see the IBM PowerNP NP4GS3 Product Overview.

Figure 2: System View with IBM Network Processor

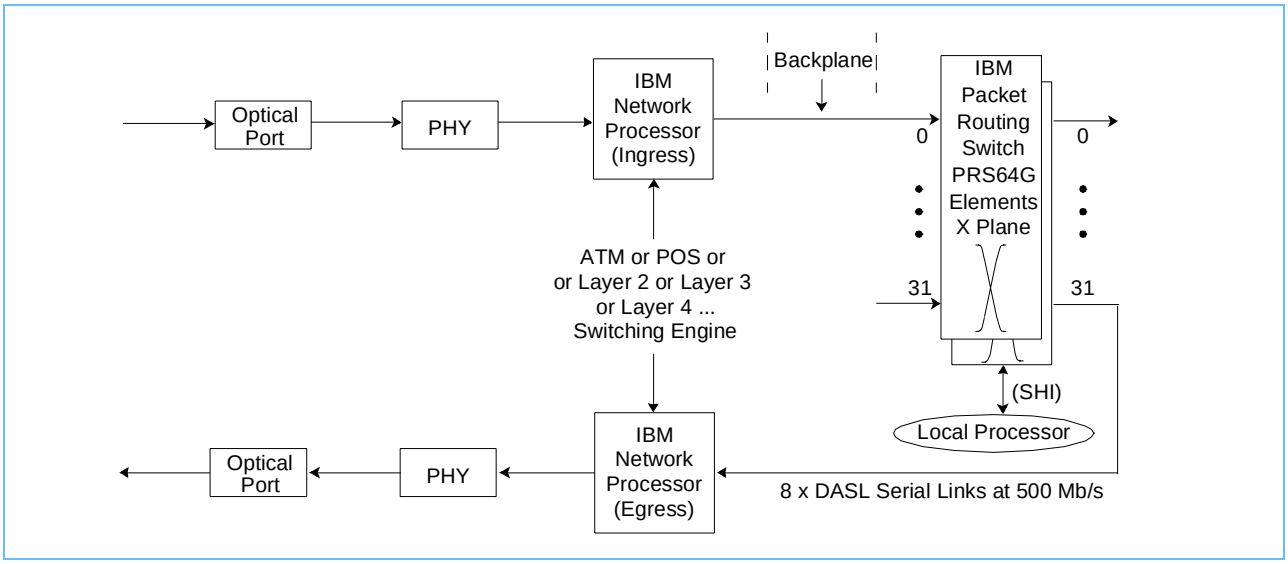
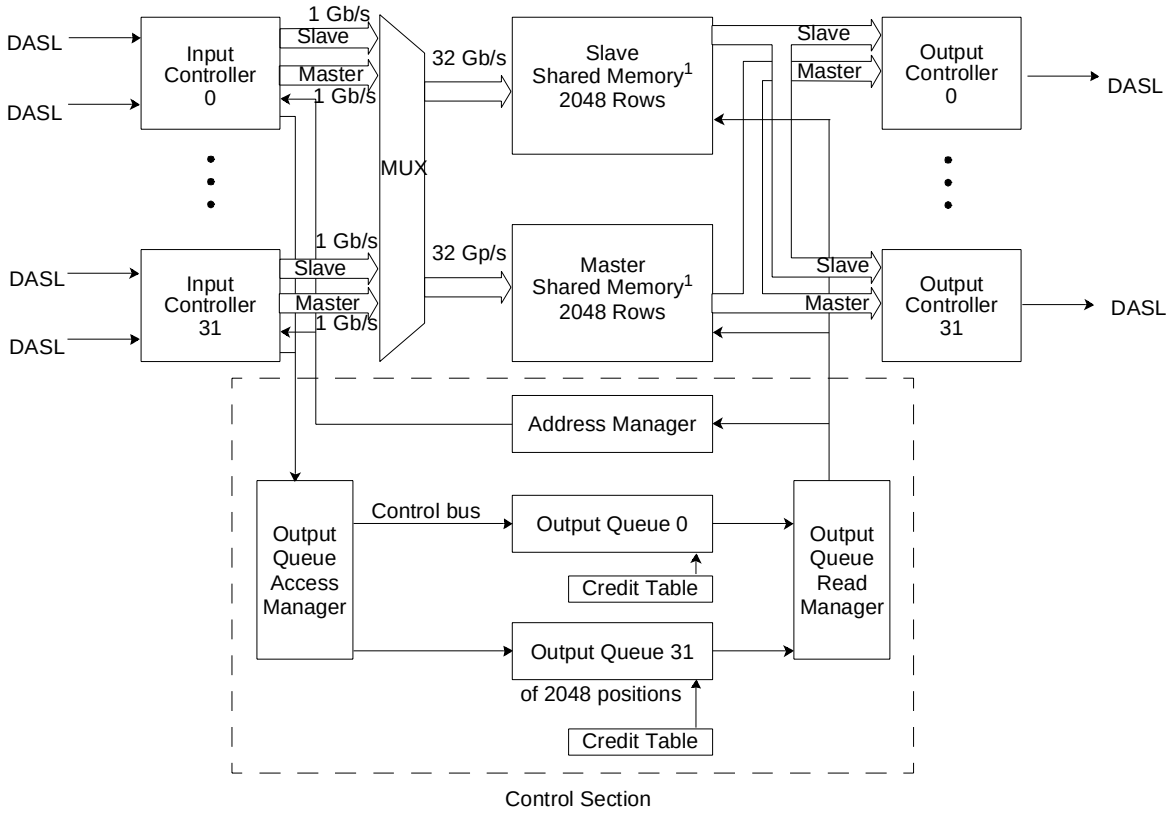


Figure 3: Packet Routing Switch Block Diagram



1. Self-routing sub-switch elements.

2. Architecture

2.1 Basic Data Flow

The PRS64G incorporates two self-routing sub-switch elements, and a control section that is the same. The device is made of two functional islands that operate identically when the switch is in regular (not speed expansion) mode. The switch has a fixed packet length shared memory, and a buffering capacity of 2048 rows of 16 to 20 bytes.

Each 2 Gb/s port carries two data streams, one master and one slave, each at one Gb/s. The master stream carries the Data packet header bytes, followed by packet payload bytes. The slave stream carries only payload bytes.

The input controllers examine the headers of incoming data packets and check the switch header integrity, using a parity bit on the header bytes. Valid data packets are then stored in the shared memory, and their storage addresses, along with the packet priority and bit map, are further processed by a centralized Output Queue Access Manager. These addresses are enqueued into output queues, one per output port and priority, according to the packet priority and bit map field. Data Packets are then transmitted, one at a time according to their place in the output queues, with the restriction that high priority packets always (except if specified differently by the Credit Table) overtake lower priority packets. However, the priority level to be received is selectable.

2.2 Multicast Packets

Multicast packets are processed the same way. A multicast packet is stored only once in Shared Memory, while its address is enqueued in all output queues indicated by its bit map field. A multicast packet is transmitted on the output ports according to the FIFO structure of each destination output queue (not necessarily at the same time on all ports). Multicast packets can only have one priority for all of their destinations.

2.3 Address Management

A central Address Manager maintains a pool of free Shared Memory addresses and provides new store addresses to the Input Controllers. Once a packet is transmitted, its address is returned to the Address Manager. The Address Manager also keeps track of the number of outputs still holding each address, since one address can be copied multiple times for multicast packets. Once this reaches zero, the address is returned to the free address pool.

2.4 Shared Memory Architecture

Shared Memory is organized as two banks, one master and one slave, each consisting of 2048 rows of 20 bytes, with one write port and one read port. Access to Shared Memory is performed one input and output port at a time. Sixteen to 20 bytes are transferred at each access, depending on the packet length. A central sequencer grants Shared Memory access to the input and output ports, cyclically. This sequencer cycle is equal, in byte cycles, to the number of data bytes stored at every access in one memory row. It is defined as an integer between 16 and 20, such that packet length is a multiple of this integer. All cycles have equal

length. Without speed expansion or with speed expansion and packet length greater than 128 bytes, an LU is received in two cycles of equal length. In speed expansion and packet length smaller than 128 bytes, it takes only one cycle to receive an LU.

2.5 Flow Control

Flow control is provided for each output queue and for the entire shared memory using a grant mechanism. Grants are given to the ingress of the attached device to allow packets to enter the switch. Similarly, the egress of the attached device provides grants to each output port to enable packets to leave the switch. On the ingress, the attached device receives the output queue grants, which reflect the status of the output queues, and memory grants that reflect the status of the Shared Memory. One output queue grant is provided per output and per priority.

2.5.1 Output Queuing and Priorities

Queuing is provided for each output port. Packet addresses from each output are organized into four logical queues of different priority. For each logical queue, packet addresses are organized into a first in first out (FIFO) queuing structure.

Packets are sorted by priority when stored, according to a priority flag they carry. The highest priority is represented by zero and the lowest by three. Packets are transmitted on a given output with the highest available priority always overtaking the lower priorities unless the weighted cyclical function has been enabled.

2.5.2 Incoming Flow Process

Each incoming packet contains information about the logical port of its destinations and the logical address (priority) per output (except for Control Packets).

The island controller allows one packet, corresponding to one input, to be processed and stored at a time. Inputs are visited once per sequencer cycle. When the input on which a packet arrives is visited by the island controller, the packet data is stored once in the shared memory. The address of this location is placed in the logical queues (specified by its priority) of all of its destination outputs according to its priority. At the same time, the shared memory counter and the output queue counters of its destination are incremented.

If an incoming packet is marked as a Control Packet, it is also stored in shared memory. Its address is then placed in a Control Packet queue, and an interrupt is sent to the local processor. An incoming Control Packet can only be received if fewer than 32 Control Packets are present in the Control Packet queue. Otherwise, the packet is discarded and a flag is raised.

2.5.3 Incoming Flow Control

Flow control of incoming packets is provided by grants which are authorizations for the attached adapter to transmit a packet. Grants are provided separately for each output port and for each priority of a port. An output queue grant for a priority is provided whenever the total packet count for an output (regardless of priority) is below the output priority threshold. It is removed whenever this count exceeds the threshold value (there is no hysteresis). Grants are also provided for the shared memory for each priority. The shared memory grant (memory grant) for a given priority is provided whenever the total packet count in shared memory is below the shared memory priority threshold, and it is removed whenever this count exceeds the threshold value (there is no hysteresis).

An adapter is only allowed to transmit a packet when it has the memory grant for the packet priority as well as the output queue grants for the destination logical outputs. For multicast packets, the adapter is allowed to transmit a packet if it has the memory grant for the packet priority, regardless of output queue grants.

Incoming packets also carry a “best effort” flag. This flag allows packet discard at the input of the island whenever the best effort discard threshold is reached due to persistent traffic overspeed for that output port, leading to a congestion in the attached adapter. The watermark counter used by the input controllers to decide to discard a packet is incremented once per received packet for each output port and decremented according to the line period of the attached port adapter.

Finally, an anti-streaming function is provided at the input controllers to detect badly behaving adapters. When an adapter sends a packet to an output priority for which the output queue grant or memory grant have not been given in the past N packet cycles (programmable from 8 to 36 cycles), the packet is discarded and an interrupt is raised. For multicast packets, the same mechanism is applied, but it takes only the memory grants into consideration.

2.5.4 Output Flow

In each of the four logical output queues, Control Packets and Data Packets are transmitted in the following order:

1. Control Packets
2. Priority 0 packets
3. Priority 1 packets
4. Priority 2 packets
5. Priority 3 packets

In normal mode of operation, this order cannot be changed.

Through a weighted cyclical mechanism, it is possible to guarantee minimum bandwidth for any priority. The programmable Credit Table indicates which priority should be transmitted at each packet cycle. When a credit is generated for priority N, then a packet of priority N is sent on that output port, provided that the send grant is active for that priority and the corresponding logical queue is not empty. If a queue is empty, the normal algorithm applies.

2.5.5 Output Flow Control

Output flow control is provided by send grants (one per output), regardless of priority in normal mode of operation. A packet on a given output can only be transmitted if the send grant is provided for that output.

There is also a Send Grant Per Priority Mode which allows the attached device to decide which packet priority is allowed to exit the switch. An antistreaming option prevents a defective attached device from keeping the send grant deasserted indefinitely. The send grant antistreaming option is programmable from 16 to 1024 packet cycles. When this condition occurs a send grant violation interrupt is raised to the attached Local Processor.

2.5.6 Output Queue Manager

The Output Queue Manager maintains a unique counter for each output queue, which indicates the total number of packets enqueued for that output, regardless of priority. The total number of packets in all four logical queues of one output is compared to the threshold value of a given priority, in order to provide flow control for

that output priority. All output counters are compared to those four thresholds once per sequencer cycle. If an output counter value is less than the threshold, the corresponding grant is set. Otherwise, it is cleared. Similarly, a counter keeps track of the total number of packets in Shared Memory.

2.5.7 Shared Memory

Four programmable Shared Memory thresholds are also provided, one for each priority. This counter is permanently compared to those four thresholds to generate the memory grants. An input ingress device is only allowed to transmit a packet when it has received both the output queue grants for the packet's destination and priority, and the memory grant for the packet priority. On the egress, a send grant is processed by each output port, either regardless of priority or per priority.The total number of packets in shared memory is compared to these threshold values for each priority, regardless of the output destination of the packets.

2.6 Control Packets

Control packets are those packets transmitted to and received from a local processor. Packets can be received on any input port, and are passed one at a time to the local processor. A 32-position FIFO queue is provided for incoming Control packets for which priority is not assigned. Control packets can be transmitted by the local processor on any set of output ports. Control packets do not carry a priority and are always transmitted on one output before any other packet present in the shared memory. Due to the slow nature of the local processor access compared to the packet data traffic rate, control packet transmission is infrequent and does not affect the performance of high priority traffic.

Data Packets for which the bit map field (Header Bytes 1:4) is 0 are detected as Control Packets and are passed to the local processor via a dedicated queue. The first bit map byte of an incoming Control Packet is overwritten by the input port number on which the packet is received.

2.7 Serial Host Interface

A Serial Host Interface (SHI) provides a serial interface to a local processor for programming application registers and accessing control packets. Refer to *7.1.2 SHI Interface Signals* on page 107 for an SHI diagram.

2.8 Serial Data Interface

A high speed serial interface is used to minimize the number of pins and to provide direct access over an extended distance. Two pairs of differential lines running at 500 Mb/s are provided for each master and slave LU stream. Therefore, each port is composed of four differential links. On the ingress side, the serial interface provides deserialization of one 2 ns bit stream into an 8 ns nibble stream. On the egress side, an 8 ns nibble stream is serialized into a 2 bit stream. For each stream, two nibbles are grouped to form a byte. Data is transmitted with a known clock such that only bit-phase alignment and packet alignment have to be performed. A picocode mechanism compensates for any skew between nibbles.

Link synchronization is provided by a training sequence of special Idle Packets, called Sync Packets. When sync packets are transmitted on a port, the receiving port can perform bit phase alignment of the incoming data, as well as packet alignment. The format of these Sync Packets is such that they are recognized by the receiving end as valid packets, and provide bit transitions on the physical line to allow for phase recovery. A sync packet LU is entirely composed of x'CC' bytes, except for the last byte which is x'33'.

One data byte is carried over two differential pairs. One pair carries all even numbered bits, and the other one all odd numbered bits. This converts the x‘CC’ sync packet bytes into alternating ‘0’ and ’1’ transitions on the physical lines.

2.9 Controllers

The internal controller is managed by a Sequencer, with a cycle value of 16 to 20 byte clocks (boundaries included). The island controller allows the handling of logical units (LUs) of 16 to 20 bytes (in steps of one byte), or 32 to 40 bytes (in steps of two bytes). An LU is the part of a packet that one island processes. The shared memory organization allows storage of 2048 LUs of 16 to 20 bytes, or 1024 LUs of 32 to 40 bytes.

2.10 Signaling

LU data is passed in and out of the island, one byte per clock cycle, for each input and output port, at a rate of 1 Gb/s. The switch header contains a packet qualifier and bit map.

The Qualifier byte is the first byte of a packet. It contains information about:

- Packet priority
- Packet identification (data, idle)
- Packet filtering support (see *2.10.4 Packet Filtering* on page 20)
- Best effort discard flag
- Parity bit over the entire header (packet qualifier plus bit map field)

2.10.1 Input Header (Bit Map)

For incoming Data Packets, one to four bytes of the packet header provide the address of the packet destination, in the form of a bit map. Each bit of these bytes is associated with a logical output port, consisting of multiple (2, 4, or 8) physical ports, grouped together by either link paralleling or speed expansion into a unique logical port. In packet addressing, there is no difference between ports that are bundled together and individual ports. A bit set to ‘1’ in the bit map indicates that the packet is to be routed to the corresponding logical port. This bit map field can point to multiple outputs.

2.10.2 Output Header (Output Queue Grant)

Depending on the header length, one to four bytes of outgoing packets carry the grant status of all output ports. This grant status is used by the receiver of the other adapter to make a decision on Data Packet transmission, according to the incoming flow control algorithm presented above. This grant status is referred to as an output logical queue grant (OQG).

On a given packet, the output queue grant field carries the status of all logical queues of the same priority. Ports bundled together represent a single logical queue. Consecutive packets carry the grant status for the different priorities, in a cyclic order. The synchronization of this cycle is provided by the packet numbering field (Flywheel priority field) contained in the outgoing Idle Packets.

2.10.3 Service Packets

There are two sets of Service Packets - Time Fill Packets and Link Information Packets.

- *Time Fill Packets* are Idle Packets that contain LU CRC. They result from either a flow control situation or buffer emptiness. When no Data Packets are available on a port, Time Fill Packets are transmitted. During switchover process, Time Fill Packets can be “colored” (red or blue) to indicate that a buffer is empty of all priorities for a given switch plane. This is to maintain a continuous sequencing of the packet clock.
- *Link Information Packets:*
  - *Link Synchronization Packets* are used as a training sequence for the serial links, and they contain the LU CRC.
  - Yellow Idle Packets are used to test the continuity of a link (Link Liveness packets). The reception of a Yellow Idle Packet on a given input is logged into a register accessible by the local processor. Yellow Packets can only be transmitted by the local processor. There are three types of yellow packets: “Yellow1”, “Yellow2”, and “Yellow3”.

2.10.4 Packet Filtering

A packet filtering function is provided on the switch island inputs in order to decide whether to receive packets for certain destinations. According to the packet filtering field in the qualifier byte, the incoming packet bit map is either logically ANDed (Red packets) with a specified mask, its complement (Red compliment packets) value, or not masked at all (Blue packets). For Data Packets, the color (red, red compliment, or blue) is used to determine how the bit map filter is applied to the bit map destination address. There is one mask for all 32 inputs.

2.11 Link Integrity

Link integrity is provided by a CRC field, placed in the last byte of all Idle Packet LUs. The 8-bit CRC is the checksum of all bits carried over a port since the previous Idle Packet ended. Thus, when multiple Data Packets are transmitted, followed at some time by an Idle Packet, the CRC of this Idle Packet covers all previous Data Packets. This provides a measure of the link quality for fault isolation.

2.12 Receive Grants

A receive grant function is provided on device pins as a way to block packet reception for specific outputs. This function allows the implementation of packet lossless switchover for simple switch systems. It cannot be combined with the packet lossless switchover mechanism provided by filtering mechanisms. When a receive grant is asserted, reception of Data Packets in the corresponding output queue is enabled. When a receive grant is deasserted, no incoming packets are stored in the corresponding output queue, regardless of their bit map setting.

2.13 Look-Up Table

The Look-Up Table is a facility that allows shuffling of the bytes of outgoing packets. Only the first 16 bytes can be exchanged among each other or overwritten by one of the 16 bytes. One table is provided for all output ports. There is one table per island.

2.14 Side Communication Channel

The 4-bit side communications channel (SCC) allows two-way communication between the adapters and the switch. It is assumed that the switching factor of this information is very low. On the path from the adapters to the switch, the information is transferred inband in the sixth byte of the Idle cells master logical unit. SCC information is carried in bits 0:3 and 4:7. When the switch receives the idle cell, it extracts and compares bits 0:3 and 4:7. If the value is the same, an internal register that holds the information is refreshed. The information is then made available through the read-only register. On the path from the switch to the adapters, the information is taken from four input pins (SCCIn 0:3).

The SCC information is inserted into all the outgoing idle cells; all the output ports send the same information. In order to guarantee a minimum time to propagate the information in case of information change, the switch automatically generates an idle cell to all ports on SCC input pins edge detection.

2.15 Expansion Modes

When multiple islands (of the same device or of two devices) are running in speed expansion, one is master and the others are slaves. The master island generates the byte stream containing the qualifier byte, bit map, and output queue grant. The master island also performs packet routing and queuing. The slave islands only receive data bytes, and do not perform any packet routing and queuing. The address of packets in shared memory is provided by the master to the slave on a special input address bus. Furthermore, the slave island's internal sequencers are synchronized on the master sequencer in order to transmit the LUs at the same time on a given port.

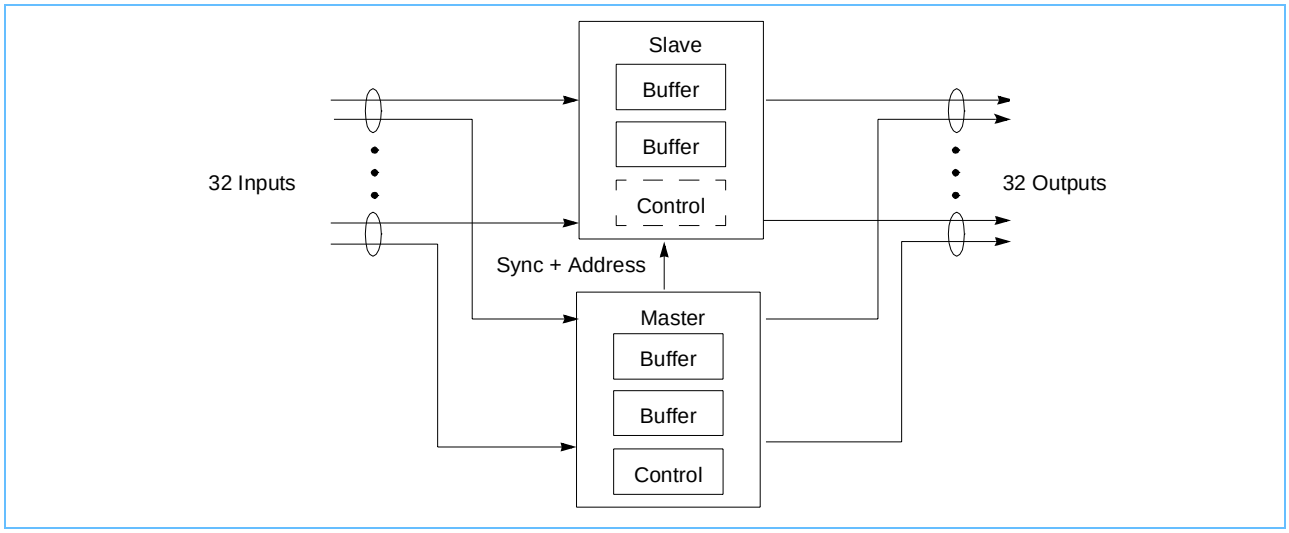
2.15.1 External Speed Expansion

Speed expansion allows multiple devices to be connected in parallel to increase the port speed while keeping the number of ports constant, as shown in the Speed Expansion Block Diagram below.

When two devices are in external speed expansion, ports of the same number are grouped together.

Only two devices can be combined in external speed expansion.

Figure 4: External Speed Expansion Block Diagram



2.15.2 Internal Speed Expansion

Ports of one island can also be paired to double the port speed while reducing the number of input and output ports to 16. Each pair of ports consists of one master port and one slave port, which have exactly the same functions as the ports of master or slave islands in external speed expansion.

Figure 5: Internal Speed Expansion Block Diagram

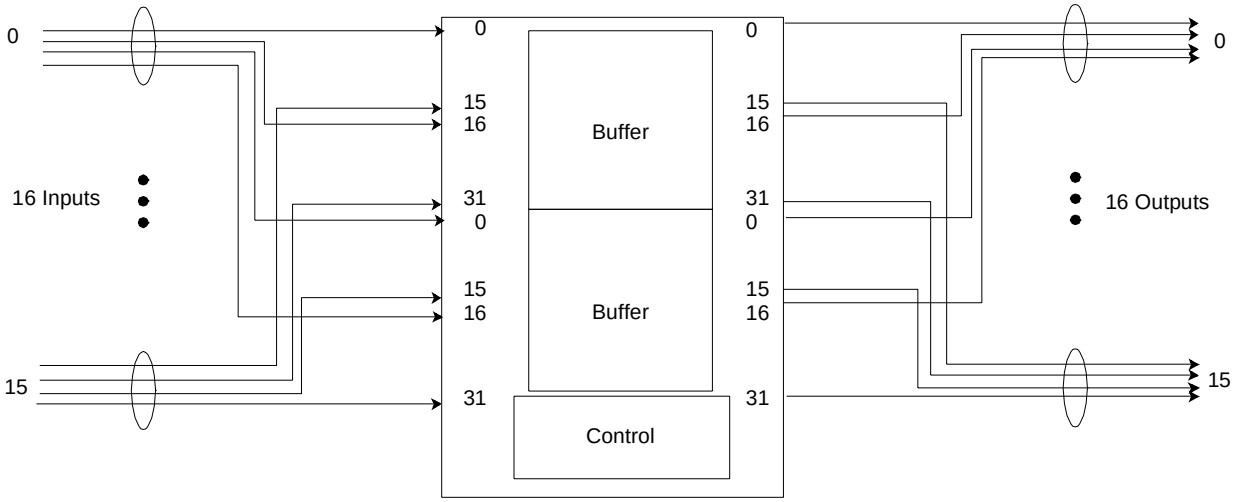


Table 2: Speed Expansion Shared Memory Buffering Capacities

Speed Expansion Mode	Switch Configuration	Gb/s port	Packet Length (bytes)	LU Size (bytes)	Buffering (packets)	Speed Expansion Factor
Single device, no internal speed expansion	32 x 32	2	64 - 80	32 - 40	1 K	2
			32 - 40	16 - 20	2 K	
Single device, with internal speed expansion	16 x 16	4	64 - 80	16 - 20	1 K	4
			128 - 160	32 - 40	512	
Two devices speed expanded, no internal speed expansion	32 x 32	4	64 - 80	16 - 20	2 K	4
			128 - 160	32 - 40	1 K	
Two devices speed expanded, no internal speed expansion	16 x 16	8	128 - 160	16 - 20	1 K	8
			256 - 320	32 - 40	512	

Note: In each case, the LU size seen by one island (either the master or the slave island of one device) is equal to the packet length divided by the speed expansion factor (port speed divided by the island port speed of one Gb/s).

2.15.3 Port Expansion

Port expansion allows multiple islands to be interconnected in parallel, in a single stage, in order to increase the number of physical ports, while keeping the port speed constant.

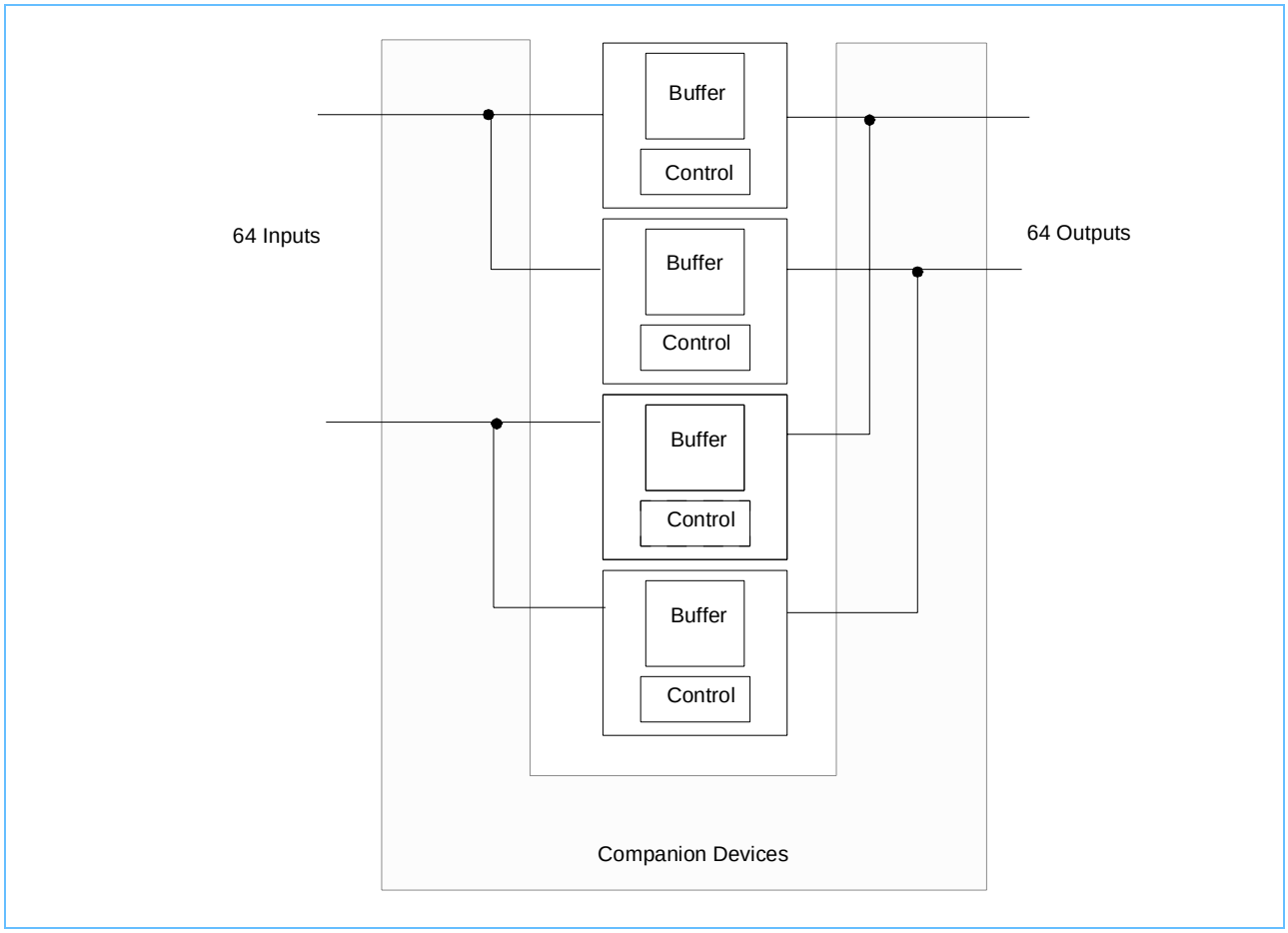
An external function must be provided to:

- Duplicate incoming packets and insert the correct bit map used by each island

- Merge traffic from different islands

Port expansion can be combined with speed expansion (internal and/or external) to increase port speed and the number of ports at the same time.

Figure 6: Single Stage Port Expansion Block Diagram



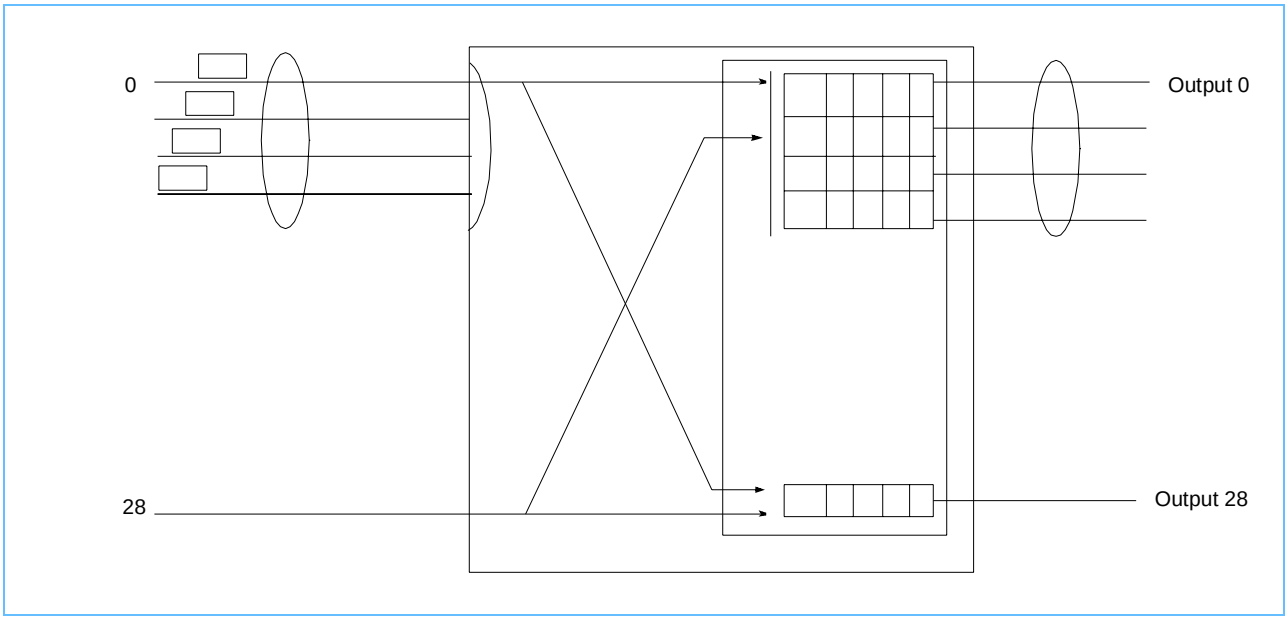
2.15.4 Physical Port Bundling

Link paralleling is the grouping of physical ports to form a unique logical port. The only difference between the ports is in the packet ordering. For packets belonging to the same flow (i.e. input/output/priority combination), packet sequence must be maintained through the switch fabric, and packet ordering must be guaranteed when transmitting and receiving packets. Packet order is based on time of transmission. Packets of ports of a common link are sent one after another, with an offset of four clock cycles (32 ns). Therefore, a port 1 packet sent 32 ns after a packet on port 0 comes after the packet on port 0 in the sequence order.

Link paralleling of four ports is only possible in the following combinations: (0,1,2,3), (4,5,6,7), (8,9,10,11), (12,13,14,15), (16,17,18,19), (20,21,22,23), (24,25,26,27), and (28,29,30,31). Not all groups have to be used at the same time. This is configurable via the Input and Output Link Paralleling field in *5.6.1 Configuration 0 Register* on page 78.

The figure below shows output link paralleling of ports (0,1,2,3), with a unique link list built in OQ 0. When sending a packet to output link 0, only the output 0 bit is set in the bit map. Addresses from OQ 0 are then distributed to the output ports 0,1,2, and 3: The OQ reads from OQ 0 when the internal sequencer indicates to process output 0, 1, 2, or 3.

Table 3: Link Paralleling Block Diagram



Link Paralleling is a logical grouping of ports. Each port or link transports independent packets. Packet ordering is such that a packet carried on a port of lower number is processed before a packet carried at the same time on a port of a higher number (for instance, if ports 0, 1, 2, and 3 are linked, a packet on port 0 comes, and thus is processed (enqueued), before packets on port 1, 2, or 3).

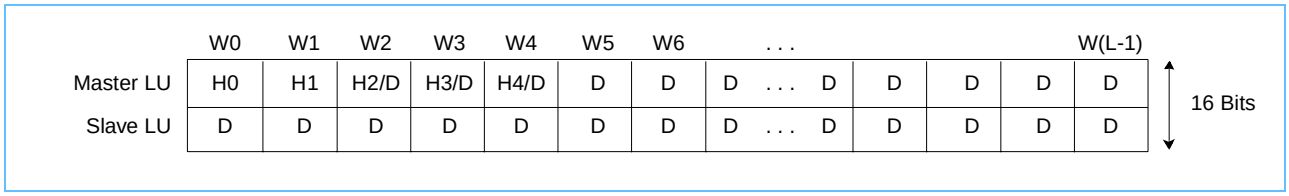
3. Functional Description

3.1 Logical Interface

As described in *2.15 Expansion Modes* on page 21, the switch consists of two switch islands running in speed expansion. Data Packets are carried over an input or output port in two byte streams of one Gb/s each, one to or from the master island, and the other to or from the slave island. Each byte stream carries Logical Units (LU). Depending on the expansion mode and packet length, an LU has a length of 16 to 20 bytes or 32 to 40 (in increments of two) bytes.

As represented in the figure below, the master LU always carries the packet routing information, or header bytes, indicated by H0, H1 through H4 (depending on the header length). Depending on the speed expansion mode and the number of ports operating in link paralleling, the number of addressable output ports can be 8, 16, 24, or 32.

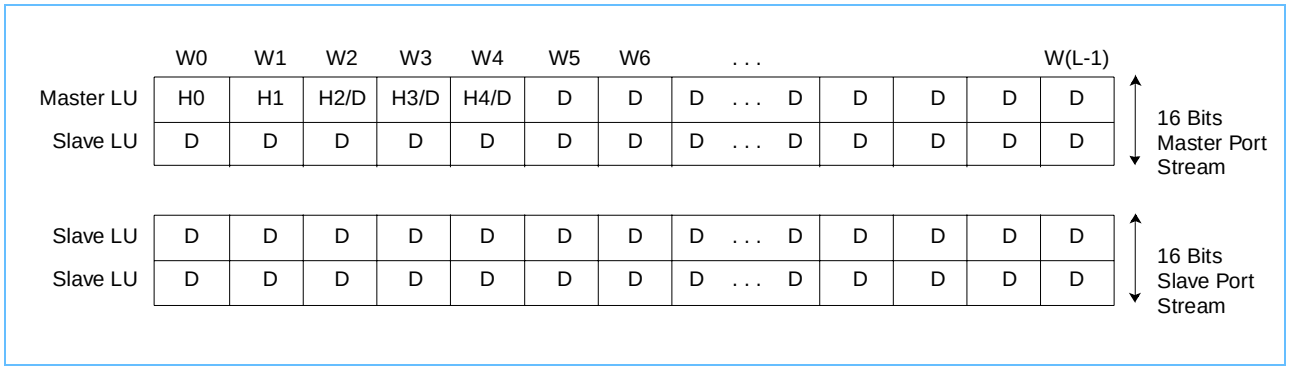
Figure 7: Packet Format 2 Gb/s Port



When running in either internal or external speed expansion, two ports are grouped to form a 4 Gb/s port. Depending on the port and chip configuration, a 2 Gb/s port can run either as a master port or a slave port:

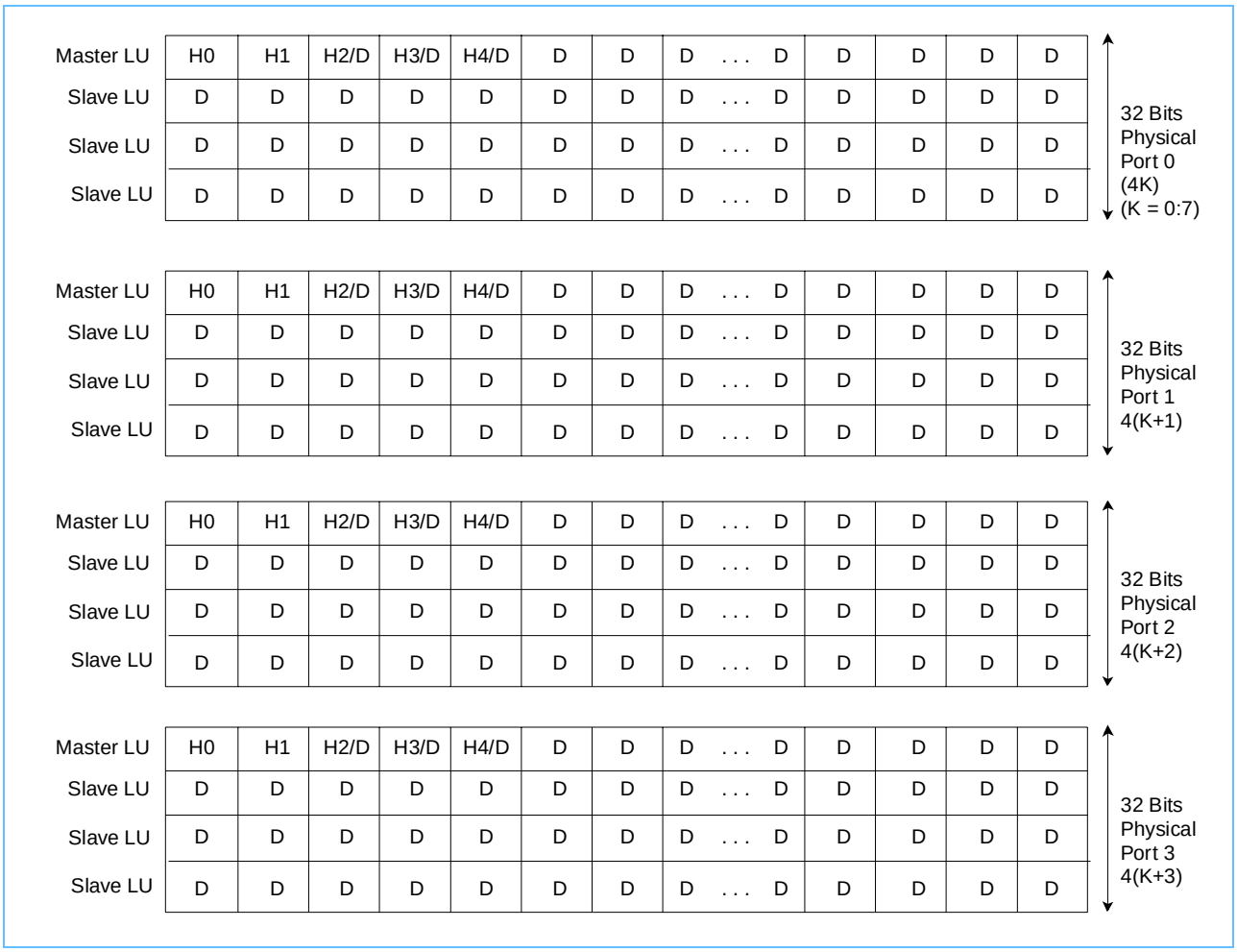
- The master port is composed of two streams, one master, which carries the packet header information and data bytes, and one slave, which carries data bytes only.
- The slave port is composed of two slave streams, which carry data bytes only.

Figure 8: Packet Format for 4 Gb/s Port



The LUs of a packet are always transmitted or received at the same time on both streams of a port. The LUs of successive packets are transported one after the other, with no gap between packets.

Figure 9: Packet Format for 16 Gb/s Port



In the case of link paralleling the internal sequencer of device processes one physical port at a time, there is an interval of 32 (4 clock cycles x 8ns)ns between each physical port of a logical port. In order to guarantee that packets are received and processed according to their order of arrival, they must be transmitted to the input physical ports with the same pattern as the physical port processing (offset of 32 ns between ports). This is the responsibility of the attached device.

3.2 Physical Interface

Within the device, an LU is transported in a stream of bytes at 125 Mb/s. Externally, it is transported over two bit streams of 500 Mb/s each. At the device-pin level, each bit stream interface is differential and complies with the JEDEC JESD8-6 standard (HSTL).

Data bits are transferred across devices at a known frequency and no companion clock is required. However, bit-phase alignment is performed during the link synchronization phase.

The interface between the 8 ns and the 2 ns domains is handled by a data line interface, the Data Aligned Serial Link (DASL).

- On the input side, the DASL deserializes the two bit streams into one byte stream. It also performs the link synchronization through training sequence to allow bit phase alignment and packet alignment. Afterwards, it automatically compensates for any variation in temperature, voltage, etc.
- On the output side, the DASL serializes the byte stream into two bit streams.

Two internal picoprocessors (shared DASL controller: SDC) perform the synchronization of all lines on all ports. One is dedicated to ports 0 to 15 and the other to ports 16 to 31. A synchronization algorithm running on the picoprocessor performs the bit-phase alignment and packet alignment of all ports during the synchronization phase (see *6.3 DASL Initialization and Operation* on page 100). The synchronization algorithm is downloaded into an internal instruction memory and is delivered along with the device.

During serialization and deserialization, a one byte stream is split into two bit streams, one carrying the even bits; the other the odd bits. A port consists of four differential pairs. The information and bit order that each pair carries is shown in the table below.

Table 4: Physical Bit Organization of a Port

Differential Pair	Information Carried	Bit Order
Data_0_Q and Data_0_QN	even number bits of the **slave** byte stream	b0 b2 b4 b6 of **slave** byte
Data_1_Q and Data_1_QN	odd number bits of the **slave** byte stream	b1 b3 b5 b7 of **slave** byte
Data_2_Q and Data_2_QN	even number bits of the **master** byte stream	b0 b2 b4 b6 of **master** byte
Data_3_Q and Data_3_QN	odd number bits of the **master** byte stream	b1 b3 b5 b7 of **master** byte

This bit grouping guarantees bit transitions to perform the phase alignment during synchronization on SYNC packets.

3.3 Packet Type

A packet can be one of the following types:

- A Data Packet that contains user data to be switched from input to output.
- A Control Packet that is used to communicate with the local processor.
- An Idle Packet that does not contain user data. Idle Packets are sent on a link when no Data Packets are available, or to perform link synchronization.
- Link Information Packet: Yellow Idle used to test the continuity of the link.

3.3.1 Data Packets and Control Packets

Data packets carry user data and Control packets carry information for local processor communication. *Figure 7: Packet Format 2 Gb/s Port* on page 25 shows the format of a Data and Control packet without speed expansion. A packet consists of 16-bit words, divided into a master stream and a slave stream.

Figure 8: Packet Format for 4 Gb/s Port on page 25 shows the format of a data or a control packet with speed expansion, either internal or external. In those configurations, the packet is transported over two different ports, either in the same device or in separate devices, in four LUs. The master port receives the master LU and a slave LU, and the slave port receives two slave LUs.

Data packets have a priority that ranges from ‘0’ (highest) to ’3’ (lowest). In addition, they also carry filtering (color) information used for switchover support, and a “best effort discard” flag.

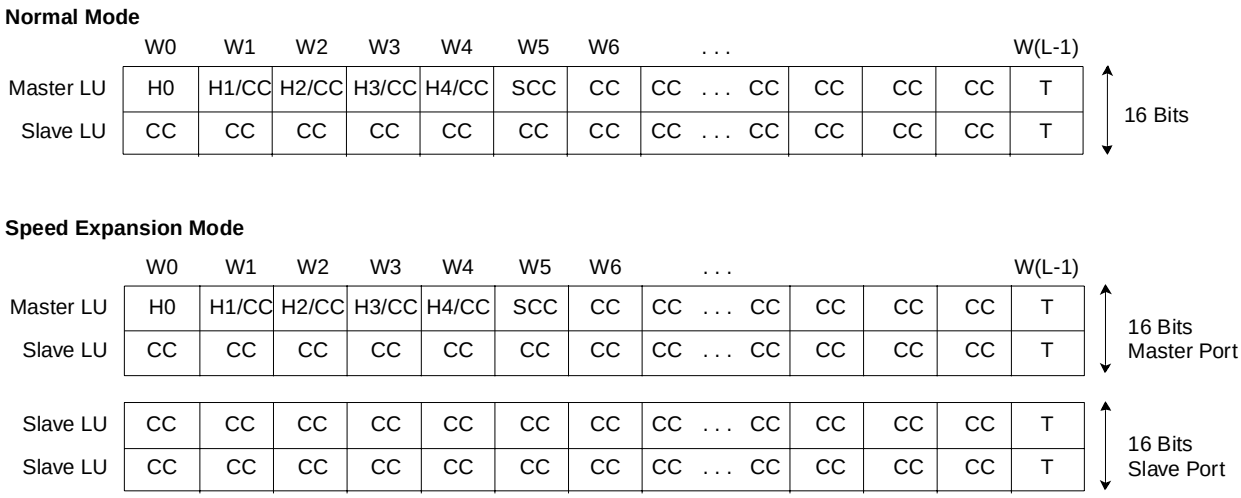
Control Packets do not have priority.

3.3.2 Idle Packets

The figure below shows the format of an egress Idle Packet with and without speed expansion. In normal mode, the format is identical to the Data Packet format for 2 Gb/s ports, except that an Idle packet contains two trailer bytes in the last word of the packet. The ingress Idle packet has the same content as the egress Idle Packet except byte H1 through H4 contain ‘CC’.

Speed expansion mode is similar to the Data packet format for 4 Gb/s ports, except for the trailer bytes at the end of the packet. The ingress Idle packet has the same content as the egress packet except bytes H1 through H4 contain ‘CC’.

Figure 10: Idle Packet Format



Note: SCC field - Bits 0:3 contain SCC information. Bits 4:7 are a copy of bits 0:3.

Like Data Packets, Idle Packets also carry color information. However, in this case the color (filtering) information is for signaling (such as liveness messages, or to identify link synchronization packets, called Sync packets), in addition to switchover support.

3.3.3 Sync Packets

Sync packets are special types of Idle Packets, which allow link synchronization by providing bit transition and packet delineation.

The bit organization by the physical interface provides on a given differential pair, a sequence of ‘A', followed by a ‘5'. This bit sequence provides the necessary bit transition while the ‘A to 5' transition allows for packet delineation.

3.4 Header Format

3.4.1 Header Byte H0 - Packet Qualifier

Table 5: H0 for "TimeFill" packets (Idle Packets)

Packet Type	Bit Position					
	0	1	2	3	4 and 5	6 and 7
	FlyWheel Extender	Parity	0	0	Color	Grant Priority bits
Blue Idle Packet	FWE	r	0	0	00	gg
Red Idle Packet	FWE	r	0	0	01	gg

r Even parity bit on entire header.

gg Value of grant sync bits (ignored for received packets - value = '00'. Grant synchronization is a flywheel mechanism that allows the device to indicate to the attached chip the priority level of the output queue grant information contained in header bytes H1 through H4.

Table 6: H0 for "Synchronization" Packets

Packet Type	Bit Position							
	0	1	2	3	4	5	6	7
Synchronization Packet	1	1	0	0	1	1	0	0

Table 7: H0 for "Yellow" Packets

Packet Type	Bit Position							
	0	1	2	3	4	5	6	7
YellowPacket Type 1	0	r	0	0	1	0	x	x
YellowPacket Type 2	0	r	0	0	1	1	x	x
YellowPacket Type 3	1	r	0	0	1	0	x	x

r Even parity bit on entire header.

x x value = 0 or 1.

Table 8: H0 for Data Packet and Control Packet (Bits 2 and/or 3 ≠ 0)

Packet Type	Bit Position						
	0	1	2	3	4	5	6 and 7
	EBM (1)	Parity (1)	Active Bit (1)	Backup Bit (1)	Best Effort Bit (1)	Reserved (1)	Priority (2)
Red compliment Data/Control Packet	x	r	0	1	be	x	pp

Key:

EBM extended bit map

x value = '0' or '1'

r even parity bit on entire header

be best effort

pp packet priority

Table 8: H0 for Data Packet and Control Packet (Bits 2 and/or 3 ≠ 0)

Packet Type	Bit Position						
	0	1	2	3	4	5	6 and 7
	EBM (1)	Parity (1)	Active Bit (1)	Backup Bit (1)	Best Effort Bit (1)	Reserved (1)	Priority (2)
Red direct Data/Control Packet	x	r	1	0	be	x	pp
Blue Data/Control Packet	x	r	1	1	be	x	pp

Key:

EBM	extended bit map
x	value = ‘0’ or ‘1’
r	even parity bit on entire header
be	best effort
pp	packet priority

Table 9: H0 Fields

Field Name	Description	Settings
Parity Bit	The parity bit is even parity calculated on the entire packet header including the reserved fields. Even parity is such that the resulting number of ‘1’ in the packet header is even. This ensures that the SYNC cell has a valid header.	
Reserved Bits	Reserved bits pass through the device unmodified.	
Data Packet Priority bits	The priority of a Data packet distinguishes between four levels of priority. When the packet is a Control packet, it is always transmitted before any other Data packets, and the priority field is ignored. Data Packets with higher priority will always be transmitted before those with lower priorities. There is a weighted cyclical mechanism based on a credit table which makes it possible to give precedence to a lower priority packet on a per port basis.	’00’ Highest priority ’01’ Medium - High ’10’ Medium - Low ’11’ Lowest priority
Active and Backup Bits	The Active and Backup bits are used to determine how the Bit Map Filter Mask is applied to the Bit Map Destination Address (header bytes H1 through H4) and to determine the traffic type (Red or Blue) of Data Packets or Idle Packets. The Bit Map Filter is a programmable register. The resulting masked destination bit map is used by the device to route the packets to the appropriate destination(s), or to ignore the packet if the resulting bit map is all zeros. Control packet detection is performed before the Bit Map Filter is applied to the bit map contained in the packet header.	Active/ Backup Bit Map FilterColor ’00’ Idle Packetn/a ’01’ Packet bit map is bit wiseRed ANDed with bit wise complement of bit map filter ’10’ Packet bit map is bit wiseRed ANDed with bit map filter ’11’ Packet bit map is usedBlue unfiltered
Best Effort Discard Bit	Incoming packets with the “best effort discard” bit set will be dropped by the device when a threshold of four up/down counters (one per priority) representing the inverse of the switch speedup compared to the attached line is reached. By monitoring the packets destined to an output queue it is possible to determine up to which best effort priority is to be discarded to ease the congestion at the convergence of the output port.	’01’ Packet bit map is bit wise ANDed with bit wise complement of bit map filter

Table 9: H0 Fields (Continued)

Field Name	Description	Settings
gg Bits	The grant priority bits (gg) are defined only for Idle Packets transmitted by the switch. They are used to synchronize the grant priority fly wheel of the attached device. For received packets, this field is not examined (its value is 'xx'). These bits encode the priority of the inband output queue grant information stored in header bytes H1 through H4. The coding of bits gg is identical to the Data packet priority.	'00' Highest priority '01' Medium - High '10' Medium - Low '11' Low priority
EBM: Extended BitMap	The purpose of this bit is to address in one cell header, half of the ports, in order to reduce the bitmap field length. When the "Extended bitmap function" is enabled, PRS64G will address 16 or 32 ports with a bitmap field of respectively 1 or 2 bytes.	'0' The bitmap field addresses the ports 0 to 7 for a 1 byte bitmap and the ports 0 to 15 for a 2 byte bitmap. '1' The bitmap field addresses the ports 8 to 15 for a 1 byte bitmap and the ports 16 to 31 for a 2 byte bitmap.
FWE: FlyWheel Extender	The purpose of this bit is to provide the inband OQ grant for half the ports in one cell header, in order to reduce the bitmap field length. When the "Extended bitmap function" is enabled, PRS64G will address 16 or 32 ports with a bitmap field of respectively 1 or 2 bytes.	FWE is always a toggle bit. When 0, it indicates that the grant field is related to ports 0 to 7 for a 1 byte bitmap and to ports 0 to 15 for a 2 byte bitmap. When 1, it indicates that the grant field is related to ports 8 to 15 for a 1 byte bitmap and to ports 16 to 31 for a 2 byte bitmap. Therefore, 2 consecutive cells with FWE equal to 0 then 1 carry the grant information for one priority.

Note: All the H0 bits are transported without modification except the Switch header parity which is refreshed. When the Best effort traffic is disabled, bit 4 has no effect. Bits 2, 3, 6 and 7 are always processed by the PRS64G. Therefore, they must be set by the user to the appropriate value.

3.4.2 Header Byte H1, H2, H3, and H4

The following chapter describes the contents of header bytes H1:H4 when the "Extended BitMap" function is disabled. When the function is enabled, the content of the header bytes H1:H4 depend on the value of the EBM and FWE bits. See descriptions for EBM and FWE in *Table 9: H0 Fields* on page 30 for more information.

3.4.2.1 Received Packet: Bitmap

Header Bytes H1 through H4 of a received packet carry the destination bit map of the packet. Header Byte H1 carries destination information for "logical" output ports 0 through 7. Header Byte H2 carries destination information for output ports 8 through 15. Header Byte H3 carries destination information for output ports 16 through 23. Header Byte H4 carries destination information for output ports 24 through 31.

For Control Packets, the entire bit map field is set to '0'.

Table 10: Header Byte 1 through 4 and Incoming Packet Bitmap

	Bit:	0	1	2	3	4	5	6	7
Header Byte H1	Port:	0	1	2	3	4	5	6	7
Header Byte H2	Port:	8	9	10	11	12	13	14	15
Header Byte H3	Port:	16	17	18	19	20	21	22	23
Header Byte H4	Port:	24	25	26	27	28	29	30	31

3.4.2.2 Transmitted Packet: Inband Output Queue Grant Information

Depending on the setting of the Flow Control Enable bit in *5.6.1 Configuration 0 Register* on page 78, Header Bytes H1 through H4 provide inband access to the output queue grant information.

When output queue grant insertion is enabled, Header Bytes H1 through H4 carry the output queue grant. The output queue grant is the grant information given to devices connected to the input ports. This information indicates the status of the output queues and controls whether or not to transmit packets to the switch. This information is carried for all 32 output ports at the same time for a given priority. Consecutive Idle and Data Packets carry a different priority, cycling from zero to the number of priority minus one. For instance, when two priorities are enabled, it takes two packets to transmit the output queue grant information.

Table 11: Header Byte 1 through 4 and Output Queue Grant

	Bit	0	1	2	3	4	5	6	7
Header Byte H1	Output Queue:	0	1	2	3	4	5	6	7
Header Byte H2	Output Queue:	8	9	10	11	12	13	14	15
Header Byte H3	Output Queue:	16	17	18	19	20	21	22	23
Header Byte H4	Output Queue:	24	25	26	27	28	29	30	31

When all four priorities are enabled, the output queue grant information is transmitted in a cycle of four packets. However, if the Three Threshold Enable bit is set in *5.6.2 Configuration 1 Register* on page 80, the cycle is reduced to 3, and grants for priorities 0, 1 and 2 only are sent. It is assumed that thresholds 2 and 3 are programmed with the same value.

To synchronize the input interface device with the inband output queue grant, the priority value for which the output queue grant is transmitted is carried in the Grant Priority bits of the Qualifier Byte of Idle Packets. Since inband grant priority is transmitted in cycles of consecutive packets, priority synchronization (through a fly wheel counter) is performed on Idle Packets only, and this information is not provided for Data Packets but the fly wheel counter continues to update itself synchronously per packet. However, grant information is always transmitted for both idle and data packets.

Table 12: Example of Fly Wheel Counter Update

Counter Value	3	0	1	2	3	0	1
Packet type	Data	Idle H0 bit gg=00	Data	Data	Idle H0 bit gg= 11	Data	Data

When the “output queue grant insertion enable” bit of *5.6.2 Configuration 1 Register* on page 80 is not set, bytes H1 through H4 are identical to bytes H1 through H4 of the received header.

3.5 Idle Packet Trailer Format

The trailer byte of ingress and egress 'timefill' Idle packets contains an eight-bit CRC checksum value which is calculated over all the bytes sent over that stream since the last Idle packet CRC byte. The trailer byte of ingress and egress Data and Control packets contains user data. Note that, depending on the module programming, the CRC is also checked on the ingress yellow packets. This is done in order to insure the compatibility with the IBM 25G Packet Routing switch.

The CRC encoding is defined by the generating polynomial X8+X4+X3+X2+1. The initial value for CRC calculation is programmable via an 8-bit CRC Init field in *5.6.2 Configuration 1 Register* on page 80. The initial CRC value is chosen so that the resulting trailer CRC of a Sync packet equals x'33' for all LU lengths. The CRC Init values are given in the chapter describing *5.6.2 Configuration 1 Register* on page 80.

When an Idle packet is received, the CRC is verified. CRC errors, if not masked, are reported via an interrupt.

3.6 Packet Reception

3.6.1 Master Input Port Operation

Packets are received on a particular input asynchronously with the packets on the other inputs, except for the ports in link paralleling. When a start of packet is received, a master input port performs the following actions:

- The packet header is analyzed and the various fields are extracted from the header. The entire packet is ignored if the header parity is incorrect. The error is reported via a Header Error interrupt and, if not masked, the main interrupt is asserted. Also, the global Header Error counter is incremented.
- If the header parity is correct and the received packet is not a sync packet, the packet color is extracted. When the packet type is yellow, bit n in the *5.6.15 Color Packet Received Register* on page 88 is set (where n is the number of the input port).

Further actions depend on the packet type and are described in the following sections.

3.6.1.1 Idle Packet

In external speed expansion, the slave is informed that an Idle packet has arrived. The Trailer Byte is verified. If there is a CRC error, it is reported via the CRC Error interrupt and, if not masked, the main interrupt is asserted. The global CRC Error counter is also incremented. No further action is taken for Idle Packets (not stored in shared memory).

3.6.1.2 Data Packet

- The packet is ignored if:
  - The packet destination output ports are all disabled, or
  - The packet cannot be stored in the packet memory. This is a flow control error and will be reported via the Flow Control Violation interrupt. This error can only occur if the adapter does not follow the memory grant information. If the interrupt is not masked, the main interrupt is asserted.

3.6.1.3 Control Packet

- The packet is only received if the number of currently enqueued Control Packets does not exceed the fixed threshold of 32; otherwise the Control packet is ignored. When 32 Control packets are already enqueued, the next Control packet is discarded, and an interrupt is raised if not masked.

3.6.2 Slave Input Port Operation

When a start of packet is received, a slave input port receives control information from the master port. For each received packet this can be one of the following:

- An Idle packet is received. The Trailer Byte is verified. If a CRC error is detected, it is reported via the CRC Error interrupt. The CRC Error Counter is also incremented. If the error is not masked, the main interrupt is asserted.
- The incoming Data packet is received. The packet is then stored in the packet memory at the address received from the master port.

If both the slave and master port receive an Idle packet, and the master port informs the slave port to ignore the Idle Packet as a result of a header parity error, the slave will ignore the Trailer Byte in the Idle packet.

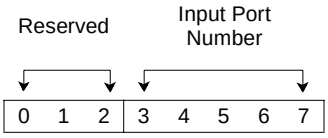
3.6.3 Parity and CRC Errors

The detection of header parity errors is reported via the *5.6.10 Header Parity Error Port ID Register* on page 84. Trailer CRC errors are reported via the CRC Error interrupt, and if not masked, generate a main interrupt. Each type of error has an individual error counter.

These interrupts are generated by the device that detects the error. In external speed expansion, the master device reports header parity errors and trailer CRC errors for data on its input port; the slave device reports trailer CRC errors for data on its input ports.

3.6.4 Address Insertion

The input port number is inserted in header byte H1 of incoming Control packets as follows:



The input port number is encoded on five bits x'8000' in the physical bit map, which corresponds to a value of X'00' in the input port number.

3.7 Input Flow Control

The switch continuously maintains shared memory and output queue status information. This status information is periodically transmitted via the memory grants and output queue grants for flow control to adapters which are connected to the switch inputs.

Flow control is used to signal the adapter that it should stop transmitting packets when the packet memory threshold is exceeded for the packet priority, or when an output queue threshold is exceeded for the packet priority.

3.7.1 Memory Threshold Exceeded Condition

There are four programmable memory-full thresholds, one for each packet priority. These thresholds can be used to prevent packets of a particular priority from using the entire packet memory. When the total number of allocated memory locations exceeds the threshold value, the corresponding memory grant signal is cleared. It is reset when the total number of allocated locations is less than the threshold value.

The four memory-full thresholds must be programmed in a decreasing manner. The memory-full threshold for priority 0 must be greater than or equal to the memory-full threshold for priority 1, which in turn must be greater than or equal to the memory-full threshold for priority 2, and so on. Consequently, when memory-full threshold 0 is exceeded, the other memory-full thresholds are also exceeded.

The memory-full information is available on chip pins (MEM_GRANT).

3.7.2 Programming the Memory-Full Thresholds

The memory-full threshold 0 should be programmed to:

NumberOfPackets - 64 - (32 * MasterGrantDelay) - ControlPacketReception - ControlPacket Transmission.

NumberOfPackets is the total packet storage available in shared memory, given in *Table 2: Speed Expansion Shared Memory Buffering Capacities* on page 22.

64 addresses are reserved by the Input Controller and should be subtracted from the total number of packets (either 1024 or 2048 depending whether speed expansion is off or on).

MasterGrantDelay, calculated in LUs, is equal to the transmission and processing time of the Memory Grant by the adapter + 1.

ControlPacketReception represents the 32 locations that are reserved for Control Packet reception, if necessary.

ControlPacketTransmission represents the location that is reserved for the Control Processor to send a packet.

3.7.3 Output Queue Threshold Exceeded Condition

There are four programmable output queue-full thresholds, one for each packet priority. All output queues use the same threshold per priority. These thresholds can be used to:

- Prevent packets of a certain priority and destined to a specific output from using the entire packet memory.
- Prevent packets of a certain priority from consuming too much output queue memory space (in relation to packets of a higher priority).

When the total number of addresses in an output queue exceeds a priority threshold, regardless of the priorities, the corresponding output queue grant is cleared. It is set whenever the total number of addresses in the output queue is below the threshold value.

3.7.4 Packet Reception Fairness

Two fairness mechanisms are implemented to guarantee that, on average, each input has an equal chance of receiving a packet:

- **Output queue-full fairness**. When multiple inputs receive packets destined for the same set of outputs, then on average, each input has the same chance to receive its packet.
- **Memory-full fairness**. When multiple inputs receive packets with the same priority and the output queue(s) are not full, then on average, each input has the same chance to receive its packet.

This is accomplished by updating the memory grant and output queue grant values once per sequencer cycle.

3.8 Output Queue Grant Signaling

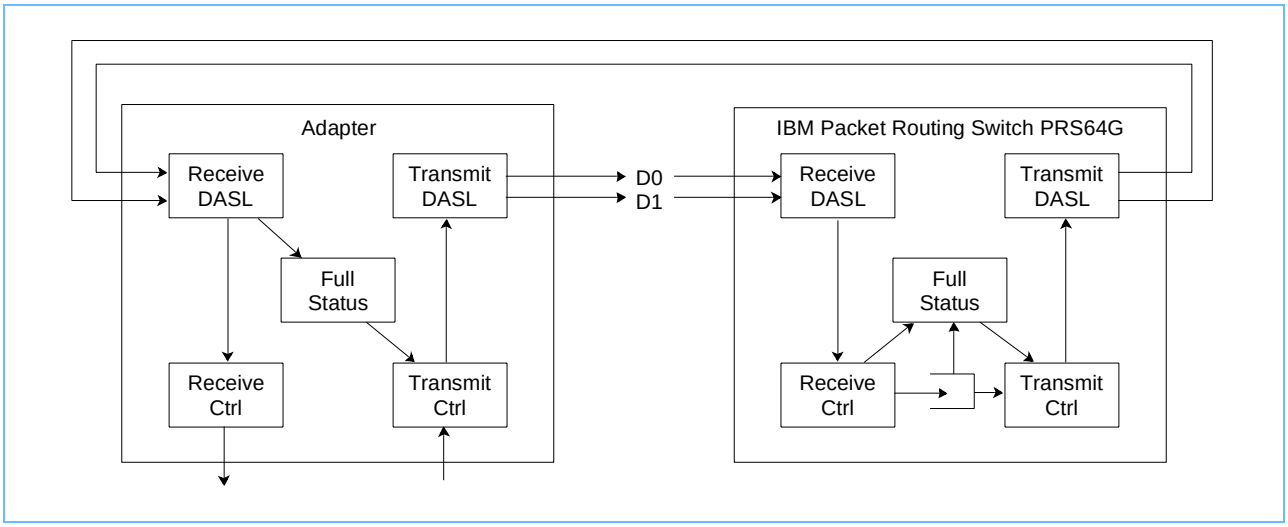
This section describes how the output queue grant flow-control status information described in the previous section is transmitted to the adapters that are connected to the switch inputs, and how these adapters should use the flow control information.

The memory grants are directly available on device pins. However, the output queue-full information is available only inband in the header of packets transmitted to the adapters if the Output Queue Grant Insertion Enable bit is set in *5.6.2 Configuration 1 Register* on page 80.

It is the responsibility of the adapter transmitting packets to the switch to determine if the packet can be accepted by the switch. The figure below shows the use of the inband output queue grant information. Inside the switch, only one input port and one output port are shown.

The adapter maintains an output queue grant status, which is a copy of the switch output queue grant status. Due to the multiplexing of the inband output queue grant of all priorities, this copy can be outdated by P * LU + 2*LU time from the switch full status, plus the latency of the attached device in processing the information, plus the DASL link latency, where P is the number of enabled priorities. See *3.4.2 Header Byte H1, H2, H3, and H4* on page 31 for a description of the inband output queue grant transmission.

Figure 11: Input-side Grant Operation



3.8.1 Adapter Transmission Rules

The adapter maintains output queue grant status, which contains the following information:

- An output queue grant bit for each packet routing switch output i and for each priority p, that is set when the adapter is allowed to transmit packets of priority p to output i. When the adapter receives a packet from the packet routing switch, it copies the 32 bits of header bytes one through four into its internal status table.
- Four memory grant bits, which override all output queue grant bits.

A rule to determine when an adapter may transmit a packet destined for a set of outputs with priority p is:

- The memory grant bit of the appropriate priority in the adapter must be set, and
- If the packet is unicast, the output queue grant bits in the adapter must be set for the destination output.

For multicast packets, it is recommended that only the memory grant bits be examined.

3.8.2 Flow Control Error

3.8.2.1 Shared Memory Overrun

As mentioned above, a packet routing switch input can always receive a packet, regardless of the full status of the output queues and shared memory, as long as it has a packet storage address. However, if a packet is received when the input does not have an address, the packet is discarded and a Flow Control Violation interrupt is set. If not masked, the main interrupt is asserted. This error can only occur if the Shared Memory Threshold bits have not been programmed correctly, or if the adapter does not respond to the memory grant pin information.

3.8.2.2 Flow Control Violation

An error can also occur if an adapter transmits packets regardless of the output queue grant and/or memory grant being deasserted. When an input receives a packet destined to an output for which the output queue grant or memory grant have not been given in the past N LU cycles (flow control latency N is programmable from 8 to 36 by increment of 2), a Flow Control Violation interrupt is set and, if not masked, the main interrupt is asserted and the packet is discarded.

When the incoming packet is multicast, this anti streaming mechanism is only based on the memory grant, and does not take the output queue grants into consideration.

3.9 Output Queues and Output Queue Priorities

The addresses of packets in the packet memory are stored in one output queue (or in multiple output queues when the packet is a multicast packet). There are 32 output queues, one queue for each output. Each output queue can store the maximum number of addresses.

In addition, each output queue is logically divided into four independent queues, one queue for each packet priority. When reading the output queue, addresses in a higher priority queue have precedence over addresses in a lower priority queue. Consequently, higher priority packets will overtake lower priority packets. However, the Credit Table allows alteration of priority scheduling, and it is possible to give the lower priorities a minimum amount of bandwidth to avoid starvation.

When the output SND_GRANT is active, the corresponding output queue is emptied at the rate at which packets are transmitted. This operation is also performed when the output queue is disabled (slow flush), regardless of the SND_GRANT value.

In addition, if an adapter keeps the send grant for a period longer than N LU cycles (N programmable from 16 to 1024), a send grant violation interrupt is raised and the slow flush mechanism is enabled.

3.10 Shared Memory

3.10.1 Organization

The shared memory is organized as two dual port large RAMs, each containing 2048 rows of 20 bytes. The two RAMs operate in parallel. The first RAM stores all the bytes in the master LU; the second RAM stores all the bytes in the slave LU.

Each island receives, byte after byte, the content of an LU and stores the entire row at a time in the shared memory. Similarly, each output reads an entire row at a time from the packet memory and transmits it to the output byte after byte.

3.10.2 Shared Memory Access by Local Processor

The local processor has full read and write access to the entire packet memory through the application registers via the *5.6.19 Shared Memory Pointer Register* on page 90 and *5.6.20 Shared Memory Data Register* on page 91. For example, this function is used when creating or reading Control Packets.

For an LU of length 17 to 20 and 34 to 40, the access time to the shared memory is guaranteed to be no more than one LU cycle. For an LU of length 16 and 32, the access time can only be guaranteed if the Control Packet Access Priority bit is set in the *5.6.2 Configuration 1 Register* on page 80.

3.11 Packet Transmission

The data stream format at the switch output side is identical to the format at the input side. However, packet transmit start times are synchronized with an internal switch packet sync clock, via the SEQ_CLK. The internal switch packet sync clock is either generated internally or can be synchronized with an external packet sync clock. This is required for a slave device in external speed expansion mode, where the sync clock of the slaves must be synchronized with the sync clock of the master. The programming of the SEQ_CLK usage is performed via the Sequencer Sync Pin Mode bit in the *5.6.2 Configuration 1 Register* on page 80.

3.11.1 Output Port Servicing

The transmission of a packet on a certain output starts at a fixed point in time, as defined in the table below.

Table 13: Packet Transmission Time

Output Port Number	Time Relationship with SEQ_CLK [byte cycle][1]	Transmission Time without Internal Speed Expansion [byte cycle]	Transmission Time with Internal Speed Expansion [byte cycle]
0 (reference), 16	21	0	0
1, 17	25	4	4
2, 18	29	8	8
3, 19	33	12	12
4, 20	22	1	1
5, 21	26	5	5
6, 22	30	9	9
7, 23	34	13	13
8, 24	23	2	-
9, 25	27	6	-
10, 26	31	10	-
11, 27	35	14	-
12, 28	24	3	-
13, 29	28	7	-
14, 30	32	11	-
15, 31	36	15	-

1. The time relationship with SEQ_CLK is the number of clock cycles between a low to high transition on the SEQ_CLK pin and the start of transmission of the first byte of a packet.

An switch output will always transmit a packet. If no Control packet or Data packet is available to transmit, or no packet has a transmission grant, the output will transmit an Idle Packet. Otherwise the switch will transmit the packet.

In external speed expansion, slave outputs start to transmit packets at the same time as master outputs.

3.11.1.1 Look-Up Table

Byte ordering of the outgoing packets can be modified via the Look-Up Tables. It is also possible to repeat the same byte several times. These tables, one for the master stream and one for the slave stream, identify which data byte of the packet to send at a specific byte time. Only the first 16 data bytes of each stream can be rearranged, and master and slave bytes cannot be mixed. These tables are common for all output ports.

Table 14: Example of Byte Reordering via the Look-Up Table

Table Entry Value	Byte Row before Ordering	Byte Row after Ordering
3	Byte 0	Byte 3
4	Byte 1	Byte 4
5	Byte 2	Byte 5
3	Byte 3	Byte 3

Table 14: Example of Byte Reordering via the Look-Up Table

Table Entry Value	Byte Row before Ordering	Byte Row after Ordering
4	Byte 4	Byte 4
5	Byte 5	Byte 5
15	Byte 6	Byte 15
14	Byte 7	Byte 14
13	Byte 8	Byte 13
12	Byte 9	Byte 12
11	Byte 10	Byte 11
10	Byte 11	Byte 10
9	Byte 12	Byte 9
8	Byte 13	Byte 8
7	Byte 14	Byte 7
6	Byte 15	Byte 6
-	Byte 16	Byte 16

3.11.2 Idle Packet Transmission

3.11.2.1 Normal Idle Packet

Time fill Idle Packets are transmitted when no Data Packets are available or when the SND_GRANT is low, according to the following rules:

- If the Color Force bit is ‘1' in the *5.6.1 Configuration 0 Register* on page 78, an Idle packet of the color equal to the Color bit of the Mode Register is sent.
- If the Color Force bit is ‘0', an Idle packet of color equal to the Expected Color bit of the *5.6.1 Configuration 0 Register* on page 78 is sent, if
  - A packet of color equal to the Expected Color has been received on all active inputs since the Color Clear command was last sent via the *5.6.21 Command Register* on page 91, and
  - The corresponding output queue is empty.
- Otherwise, an Idle packet of color opposite to the Expected Color is sent.

3.11.2.2 Yellow Idle Packet

Yellow Idle Packets can only be generated by the Local Processor. The local processor must build the yellow Idle packet in shared memory and transmit it to the desired outputs as for any other Control Packet. Therefore on the egress port the yellow Idle packet does not carry the LU CRC or the grant priority flywheel bits.

3.12 Send Grant

There are two modes of Send Grant operation - a single indication independent from the priority, and a mode where the attached device has the capability to determine which priority it want to receive.

3.12.1 Send Grant without Priority Indication

Flow control is provided to the output ports via 32 send grant SND_GRANT pins. Data or Control Packets are only transmitted on a given output if its SND_GRANT is active high. Otherwise, Idle Packets are transmitted as long as the SND_GRANT is low.

These signals are treated asynchronously to the internal device logic. Therefore, if the transitions on the SND_GRANT pins are not synchronized with the output packet flow, an exact control of the number of Data or Control Packets transmitted cannot be bounded to two packets. For example, when SND_GRANT is high for N * LU cycles, N-1 to N+1 Data or Control Packets will be transmitted (if available). Similarly, when SND_GRANT is low for N * LU, N-1 to N+1 Idle Packets will be transmitted.

However, if the SND_GRANT transitions are synchronized with the output packet flow, N Data or Control Packets (if available) will be transmitted if SND_GRANT is high for N * LU, and N Idle Packets will be transmitted when SND_GRANT is low for N * LU. In order to achieve this, the synchronization requirement is that the SND_GRANT signal should be stable at the device pin during its sampling window, defined in relationship with the beginning of the transmission of the corresponding packet on device pins (See I/O Timing section).

This SND_GRANT sampling window also defines the latency of the packet transmission reaction to changes of the SND_GRANT signal both when it is asserted or deasserted.

3.12.2 Send Grant per Priority

SND_GRANT can also be operated per priority. In this mode, the SND_GRANT interface line is multiplexed per priority over one line (one per port).

In this mode of operation the attached device provides serial information timed by a clock at half the LU byte clock and based on a unique header followed by the priority bits.

The pattern for edge detection and multiplexing of the SND_GRANTs (P0, P1, P2 and P3) is shown below. It is composed of 9 x 16ns intervals.

Note: This pattern therefore, can be padded by ‘0’s, if its length needs to be adjusted (to LU size for instance).

0	0	0	0	1	P0	P1	P2	P3

The ‘00001’ pattern cannot be reproduced by any combination of the SND_GRANTs.

Phase alignment and pattern delineation is performed by edge detection (through oversampling) on the first ‘1’ following four ‘0’s by oversampling the SND_GRANT interface line. The alignment is performed for every new set of SND_GRANT values, which provides an insensitivity to long term clock phase jitter.

The attached device provides a priority bit map which will allows the switch device to send any packet with the granted priority(ies) and to hold the other(s).

3.13 Receive Filter

The receive grant permits the filtering of incoming packets based on their destination. It prevents the reception of Data Packets by selected outputs. The 32 RCV_GRANT bits act as a filter of the incoming packet bit map:

- When RCV_GRANT(i) is high, reception of Data Packets in the output queue i is enabled,

- When RCV_GRANT(i) is low, no incoming packets are stored in output queue i, regardless of their bit map setting.

The RCV_GRANT signals are treated asynchronously to the internal logic. However, it is possible to synchronize all 32 RCV_GRANT signals with the input packet flows in order to guarantee reception, or rejection, of specific packets. To this end, the RCV_GRANT signals must be stable on the device pins during the sampling window, defined in relationship with the beginning of the reception of the corresponding packet on the device pins (See *7.1 I/O Timing* on page 107).

The RCV_GRANT function is actually implemented as an extension of the Bit Map Filter on device pins (RCV_GRANTs are ANDed with the Bit Map Filter) and therefore has the exact same effect. When using the RCV_GRANT function:

- The Color mechanism and the Bit Map Filter function should not be used
- Only Data Packets with header Active bit set to ‘1'b and Backup bit set to ‘0’ can be transmitted, and
- The Bit Map Filter has to be set to x‘FFFF'

3.14 Port Disabling

An input port and the corresponding output port can only be disabled by programming the *5.4.4 Input Port Enable Register* on page 69 (x'0B') and *5.4.2 Output Port Enable Register* on page 68 (x’09’).

3.15 Address Manager and Address Corruption (Book Keep Mechanism)

The address manager controls packet addresses by maintaining a pool of free addresses and occupancy counters for each address. It provides the input controllers with addresses from the free address pool. When an address is used, the occupancy counters are initialized to the number of destinations of the corresponding packet. Each time a packet is transmitted, the occupancy counter of its address is decremented by one. When the counter reaches zero, the address is returned to the free address pool.

Since addresses are continuously used as packets are being received and transmitted, it is important to detect address corruption scenarios. Address corruption can lead not only to corruption of packets, but also to a loss of available addresses, which can decrease performance.

Therefore, two error detection mechanisms are in place to detect corruption:

- Each output queue is protected by parity. When an address is written to an output queue, parity is generated and stored with the address and it is checked upon reading this location.
- The address manager detects the following error scenarios:
  - A counter is initialized to a value while it is not zero,
  - An address is freed by an output controller when its counter is zero.

Each of these errors sets the Address Corruption interrupt and, if not masked, the main interrupt is asserted.

3.16 Control Packets

It is possible to address a local processor attached to a packet routing switch by means of Control packets. A Control packet is similar to a normal Data packet, except that its memory addresses are stored in a special Control packet queue.

For external speed expansion, Control packet received and transmitted interrupts are only generated by the master. Also, the Load Next Control Packet Address command, and Free Control Packet Address command, are only issued to the master. No action is taken if those commands are issued to the slaves.

Whenever performing a shared memory access to receive or transmit a Control packet, the access time to the memory is guaranteed by the sequencer to be at most one LU unit length, except for LUs of length 16 and 32 bytes. This also is true for both the master and slave in speed expansion.

For LUs of 16 and 32 bytes, the Control Packet Access Priority Enable bit (*5.6.1 Configuration 0 Register* on page 78) must be set in order to guarantee a memory access time of at most three LU unit lengths (this depends on the latency in reacting to the Memory Grant mechanism by the attached device). This rule applies only to single device configurations or to the master in speed expansion. In speed expansion, a master operation follows a slave operation as described in the sections below. In this case, it is guaranteed that the slave memory access is completed when the master memory access is completed.

3.16.1 Control Packet Reception

Control Packets can be received on all input ports and are stored as any other packet in the shared memory. The shared memory locations of the Control Packets are written into a Control packet queue, which allows for multiple Control Packets to be received. The Control packet queue has 32 locations. This provides fairness among all inputs. Each input can receive a Control Packet at the same time and all packets can be processed. If the queue is full when a Control packet arrives, it is discarded. When a Control packet is accepted, a Control Packet Received interrupt is generated and, if not masked, the main interrupt is asserted. The local processor can then access the Control packets via the *5.6.18 Shared Memory Access Registers* on page 89.

The number of Control packets currently enqueued is provided via the Control Packet Counter. Upon reception of a Control Packet Received Interrupt, the following tasks are performed:

1. A Load Next Control Packet Address command is issued via the *5.6.21 Command Register* on page 91. This loads the first address of the Control Packet queue into the *5.6.19 Shared Memory Pointer Register* on page 90.
2. The *5.6.19 Shared Memory Pointer Register* on page 90 is read. This is address A:

External Speed Expansion	Internal Speed Expansion	Packet Size (No. of Bytes)	Addresses to read in master and slave memory banks
0	0	32 to 40	A
0	1	32 to 40	A
1	0	32 to 40	A
0	0	64 to 80	A and A+1
0	1	64 to 80	A and A+1024
1	0	64 to 80	A
1	0	128 to 160	A and A+1
0	1	128 to 160	A and A+1 A+1024 and A+1025
1	1	128 to 160	A and A+1024
1	1	256 to 320	A and A+1 A+1024 and A+1025

3. All the rows of the Control packet are read via the *5.6.19 Shared Memory Pointer Register* on page 90 (Master LU and slaves LU) and *5.6.20 Shared Memory Data Register* on page 91. A Free Current Control Packet Address command is issued via the *5.6.21 Command Register* on page 91 to free up the first address in the Control Packet Queue.
4. If the Control Packet Counter is not zero and/or if another Control Packet Received Interrupt is issued, steps one through four are performed again.

In external speed expansion, the shared memory start address of the Control packet for the slaves is the same as for the master. Therefore, after issuing the Load Next Control Packet Address command to the master, the local processor must read the master *5.6.19 Shared Memory Pointer Register* on page 90 and write its content to the slave *5.6.19 Shared Memory Pointer Register* on page 90. According to how the control packet access priority enable bit (configuration register 1 @x'21') has been set (mainly in the case that the LU size is 16 or 32 and the switch is loaded at 100%), the local processor must then issue a Read Command to the slave, followed by a Read Command to the master. Only then can the data from the slave and the master be read if the Control Packet Access Priority Enable bit in *5.6.2 Configuration 1 Register* on page 80 has been set. Once the first row is completely read in both master and slave, the same sequence of a Read Command to the slave and the master has to be issued in order to read the second data row in both devices. This is necessary to guarantee the access time to the slave shared memory. All rows in both the master and the slave must be read before issuing the Free Current Control Packet Address command to the master.

Note: The reference of the input port which received the Control packet is inserted in the header position H1.

3.16.2 Control Packet Transmission

The Local Processor has write access to the entire packet memory. There is an address reserved as indicated in the following table. The contents of these rows can be transmitted upon command from the Local Processor.

Table 15: Shared Memory Reserved Address for Control Packets

External Speed Expansion	Internal Speed Expansion	Packet Size (No. of Bytes)	Shared Memory Reserved Address
0	0	32 to 40	0
0	0	64 to 80	0 and 1
0	1	64 to 80	0 and 1024
1	0	64 to 80	0
1	0	128 to 160	0 and 1

Control Packets are sent one at a time. Note that Control Packets can only start at shared memory location zero. In order to transmit a Control Packet, the following tasks are performed:

1. The first row of the Control packet in the Memory Row Register is built via the *5.6.20 Shared Memory Data Register* on page 91.
2. A Memory Row Write at address 0 command is issued via the *5.6.19 Shared Memory Pointer Register* on page 90.
3. Items one and two are repeated for all rows of the Control Packet.
4. The output ports from which the Control packet must be transmitted are specified by loading the *5.6.22 Control Packet Destination Register* on page 92.

5. A Wait for a Control Packet Transmitted Interrupt (an interrupt will be raised when all raws are transmitted).

In external speed expansion, the local processor must write the rows to the slave and the master devices before issuing the Control Packet Transmit command. The local processor must first write a row to the slave, followed by a row to the master. This sequence is necessary to guarantee access time to the slave memory.

Note: The *5.6.22 Control Packet Destination Register* on page 92 must be specified only for the master chip.

3.16.3 Packet Reception Window for Speed Expansion

Packets belonging to a group of related speed-expanded inputs must arrive within a specific window. The reference point (the input that defines packet arrival time Ta) is the master in a speed expanded configuration. All other related inputs must receive packets between Ta - 1 byte clock and Ta + 1 byte clock.

3.17 Speed Expansion

The purpose of speed expansion is to increase the physical port speed of a switch. Two methods are supported: external speed expansion and internal speed expansion.

3.17.1 External Speed Expansion

One master device and one slave device can be used in so called external speed expansion mode. The master device performs all the address handling. That is, it maintains the output queues and the address manager. These components are idle in the slave device. Packet addresses, used by the inputs, are sent from the master device to the slave device on a special input address bus. The timing in a master device is such that master inputs receive new addresses at the same time as all the slave inputs.

Packet addresses used by the outputs are sent in a similar fashion from the master device to the slave device. Again, the timing in the master device is such that a master output receives a new retrieve address at the same time as the slave output. A master output and the corresponding slave output always start the transmission of a packet simultaneously.

When two devices are in external speed expansion, ports of the same number are grouped together.

Only two devices can be combined in external speed expansion.

3.17.2 Internal Speed Expansion

It is also possible to connect two ports on the same device together in speed expansion. This allows for a four Gb/s 16 x 16 switch. The table below shows which inputs and outputs are combined.

Internal and external speed expansion may not be combined together.

Table 16: Port Combination in Internal Speed Expansion

Internal Speed Expansion Mode (configuration 1 register)	External Speed Expansion Mode (configuration 1 register)	Configuration	Port Speed	Port Combination
0	0	32 x 32	2 Gb/s	none

Table 16: Port Combination in Internal Speed Expansion

Internal Speed Expansion Mode (configuration 1 register)	External Speed Expansion Mode (configuration 1 register)	Configuration	Port Speed	Port Combination
1	0	16 x 16	4 Gb/s	0 and 16, 1 and 17, ... 15 and 31
1	1	16 x 16	8 Gb/s	0 and 16, 1 and 17, ... 15 and 31

3.17.3 Synchronization of Slave Device with Master Device

The master device and the slave device in an external speed expansion configuration must be synchronized. This is done with SEQ_CLK sync pin, which can operate in one of the following modes:

- When the PRS64G is configured as a master device in external speed expansion, or in single device operation, the SEQ_CLK pin is programmed as an output. The device generates an internal sync signal that is used for its internal timing. This internal sync signal appears on the SEQ_CLK output.

  Note: It is possible to configure the SEQ_CLK of the master as an input and to externally generate this signal. However, this signal must be distributed to the master and the slave synchronously to the device internal byte clocks.

- When the PRS64G is configured as a slave device in external speed expansion, the SEQ_CLK must be programmed as input. The sync signal that is used for the internal timing is taken from the master.

Each PRS64G contains a sequencer. The SEQ_CLK sync signal is used to synchronize the sequencer in the slave device with the sequencer in the master device. Both the master and slave sequencer are fully synchronized. Therefore the SEQ_CLK is fully synchronous to the device internal logic.

3.17.4 Master Slave Address Communication

Two busses, one for the input section and one for the output section, are used to transfer addresses from the master to the slave:

- Each bus is synchronous with the internal byte clock.
- When external speed expansion is disabled, the busses are tri-state.
- When external speed expansion is enabled, and the device is programmed as a master, the busses are configured as output.
- When external speed expansion is enabled, and the device is programmed as a slave, the busses are configured as input.
- Each bus has a parity signal:
  - When the slave detects a parity error on the input bus, the address is ignored, the packet is not stored, and a speed expansion error interrupt is generated.
  - When the slave detects a parity error on the output bus, the address is ignored, the slave transmits Idle Packet LUs, and a speed expansion error interrupt is generated.

Therefore, the slave does not report parity errors directly to the master, but does interrupt the local processor.

- The maximum external speed expansion factor is two (which logically represents four-way speed expansion). That is, there can be one master and one slave.

3.18 Link Paralleling Bitmap Mapping

With four ports operating in link paralleling, only one bit is consumed in the incoming bitmap. With many ports operating in link paralleling, the full 32-bit bitmap may be useless and a shorter bitmap header can be used to optimize the header/payload ratio. This increases the total payload bandwidth available for each port. An example is the packet routing switch operating in 8x8 mode, which only requires an 8-bit bitmap.

Any physical port in multiple of four (0, 4, 8, 12,16, 20, 24, 28) can operate in one or four-way link paralleling. Therefore LP physical ports are in fixed groups of four and the lowest physical port is the one used to identify the physical link paralleling.

The following table is an example of the mapping between a logical |bit of the bitmap (with lowest port numbers operating in link paralleling) and its associated physical queue within the chip. Queues that can operate in link paralleling mode appear in bold. Numbers appearing with a smaller font indicate the physical queues which are used (no more available) by this queue which operates in link paralleling mode. The port in link paralleling is always referenced to the lowest physical port. The table shows the LP ports in ascending order but the LP ports can be setup in any order. The table also shows the associated possible bit map length (8, 16, 24, 32) recommended for each configuration.

Table 17: Link Paralleling Configuration Example with Link Paralleling in Ascending Order

Logical Bitmap	No LP 32 ports	1 LP 29 ports	2 LP 26 ports	3 LP 23 ports	4 LP 20 ports	5 LP 17 ports	6 LP 14 ports	7 LP 11 ports	8 LP 8 ports
0	00	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**	**00 (01,02,03)**
1	01	04	**04 (05,06,07)**	**04 (05,06,07)**	**04 (05,06,07)**	**04 (05,06,07)**	**04 (05,06,07)**	**04 (05,06,07)**	**04 (05,06,07)**
2	02	05	08	**08 (09,10,11)**	**08 (09,10,11)**	**08 (09,10,11)**	**08 (09,10,11)**	**08 (09,10,11)**	**08 (09,10,11)**
3	03	06	09	12	**12 (13,14,15)**	**12 (13,14,15)**	**12 (13,14,15)**	**12 (13,14,15)**	**12 (13,14,15)**
4	04	07	10	13	16	**16 (17,18,19)**	**16 (17,18,19)**	**16 (17,18,19)**	**16 (17,18,19)**
5	05	08	11	14	17	20	**20 (21,22,23)**	**20 (21,22,23)**	**20 (21,22,23)**
6	06	09	12	15	18	21	24	**24 (25,26,27)**	**24 (25,26,27)**
7	07	10	13	16	19	22	25	28	**28 (29,30,31)**
8	08	11	14	17	20	23	26	29	
9	09	12	15	18	21	24	27	30	

Note: Queues operating in link paralleling mode appear in bold. Numbers appearing with a smaller font indicate the physical queues which are “consumed” (no more available) by this queue which operates in link paralleling mode.

Table 17: Link Paralleling Configuration Example with Link Paralleling in Ascending Order (Continued)

Logical Bitmap	No LP 32 ports	1 LP 29 ports	2 LP 26 ports	3 LP 23 ports	4 LP 20 ports	5 LP 17 ports	6 LP 14 ports	7 LP 11 ports	8 LP 8 ports
10	10	13	16	19	22	25	28	31	
11	11	14	17	20	23	26	29		
12	12	15	18	21	24	27	30		
13	13	16	19	22	25	28			
14	14	17	20	23	26	29			
15	15	18	21	24	27	30			
16	16	19	22	25	28				
17	17	20	23	26	29				
18	18	21	24	27	30				
19	19	22	25	28	31				
20	20	23	26	29					
21	21	24	27	30					
22	22	25	28	31					
23	23	26	29						
24	24	27	30						
25	25	28	31						
26	26	29							
27	27	30							
28	28	31							
29	29								
30	30								
31	31								

Note: Queues operating in link paralleling mode appear in bold. Numbers appearing with a smaller font indicate the physical queues which are “consumed” (no more available) by this queue which operates in link paralleling mode.

There is also a multiplexer of 32 by 32 ports, and each logical port can be mapped to any physical port. For example, physical port 1 can be addressed by logical bit map 0 (instead of logical port 1 corresponding to physical port 1 as shown in the table above).

3.19 Best Effort Discard

The PRS64G implements a subflow discard mechanism for applications in which the lowest priorities are prevented from accessing an output port. This is done because in the higher priorities some subflows are less important than some of the lowest priority subflows. The packet routing switch implements virtual queue monitoring and an optional packet discard to prevent those best effort subflows from jeopardizing the quality of service of the guaranteed priority subflows. This allows the packet routing switch to perform a best effort burst discard when all flow control capability has been exhausted. The discard is done in burst to minimize the number of frames that are affected. As with lossless operation, when the packet routing switch is operating as

a lossy switch, flow control is activated when the amount of traffic for an output port exceeds the port's capacity. The packet routing switch discards traffic flagged as best effort when the aggregate load for that output port exceeds the port's capacity for a significant amount of time.

The PRS64G assumes that the guaranteed bandwidth traffic is never discarded and therefore congestion for this traffic is managed only through the normal flow control mechanism.

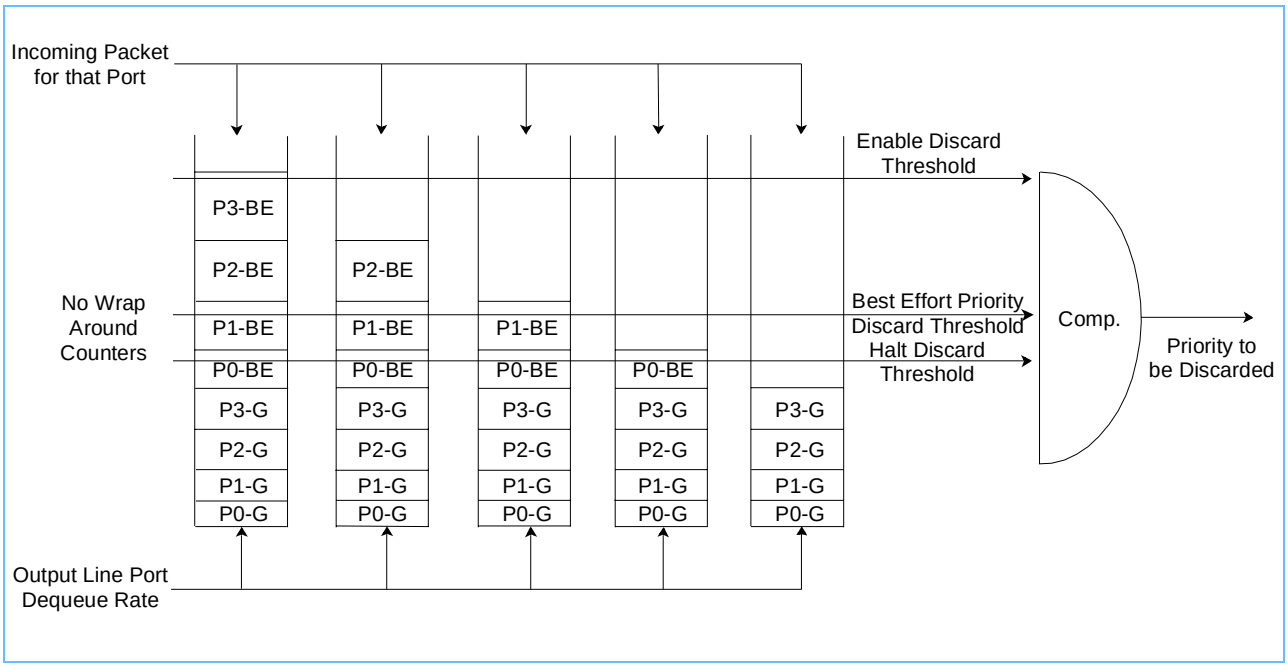
The operation of the lossy switch is based on a set of five counters per port that are decremented at a packet speed equivalent to the throughput of the port to which it is attached (the line rate is provided by a register defined at the initialization of the port). The five counters are updated as follows:

- The main counter represents the total traffic for a given output port. It is incremented for each packet destined to the output port it is monitoring.
- The second counter counts all incoming packets for a given output port except the packets which are of lowest priority (Priority 3) and which are best effort bandwidth. The guaranteed traffic of priority 3 is also counted.
- The third counter counts all incoming packets except best effort packets of priority 3 and 2.
- The fourth counter counts all incoming packets except best effort packets of priority 3, 2, and 1.
- The fifth counter counts only those packets corresponding to the guaranteed bandwidth traffic.

Associated with the main counter is an enable discard threshold (EDT) which, when reached, will trigger the discard mechanism. There is also a halt discard threshold (HDT) based on the value of the main counter used to end the discard process. The other four counters have a single common threshold, the Best Effort Priority Discard Threshold (BEPDT), which determines which subflow(s) have to be discarded. The assumption is that if the value of any of the counters exceeds the threshold, the discard of that traffic will not be sufficient to ease the flow of the guaranteed traffic bandwidth. The PRS64G immediately knows which priority of best effort subflow traffic has to be discarded based on past history. It discards only the best effort bandwidth traffic corresponding to the priority creating the problem as well as the best effort traffic corresponding to the lower priorities. The BEPDT threshold is analyzed only once when the discard process is started. Afterwards, only the HDT threshold is used to determine when to stop the discard process.

The global assumption is that the guaranteed bandwidth should be engineered so that it never exceeds the discard threshold. The objective is that, on average, the guaranteed bandwidth is engineered to dequeue without having the need for flow control. However, there might be situations where the nature of the traffic changes before the HDT is reached. In this case, the discard mechanism does not help to ease the congestion. So, as an option when the EDT threshold is reached, the algorithm analyzes the BEPDT every packet cycle to determine if there is not a new best effort priority which is impairing the guaranteed bandwidth.

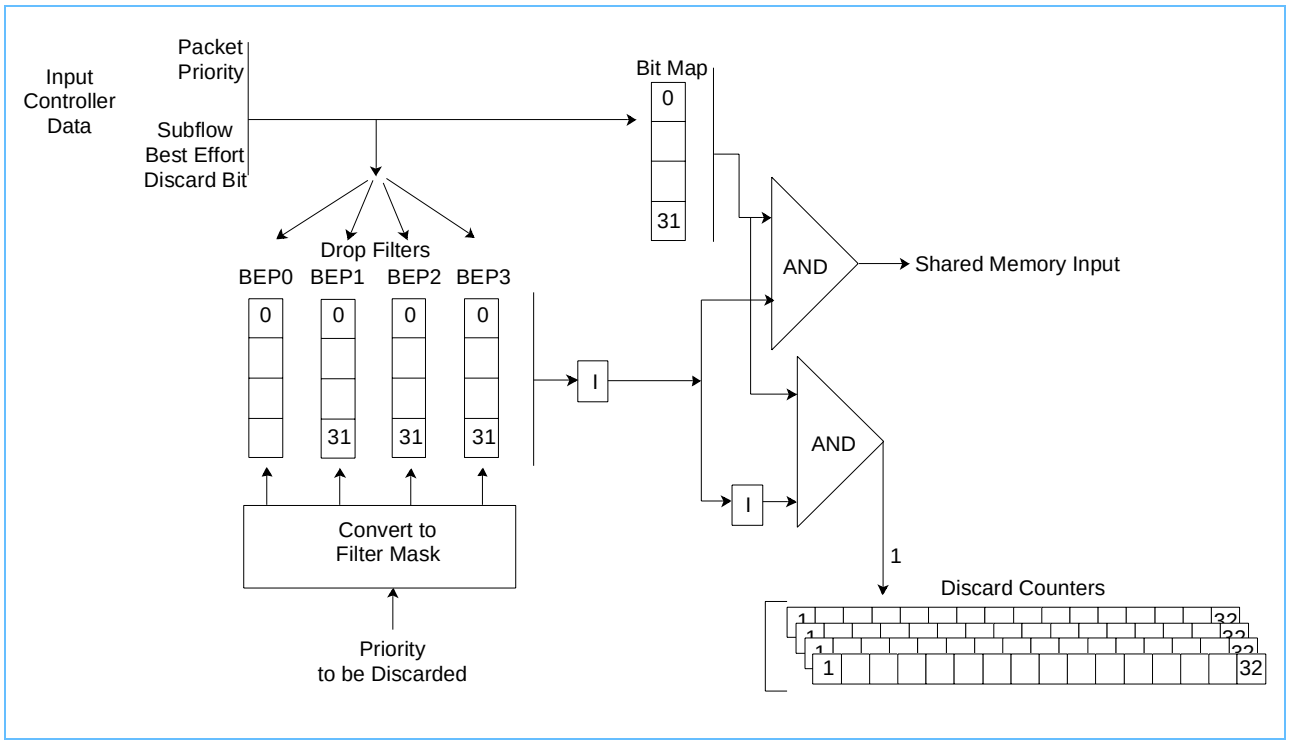
Figure 12: Best Effort Burst Discard



3.19.1 Subflow Best Effort Discard Filter

When an incoming packet arrives at the switch input, its priority bits, the subflow best effort discard bit, and the destination Bit Map are provided to the best effort filter logic. The priority selects the appropriate BEPx Drop Filter. The Filter is used as a destination Bit Map Mask (Drop = 0). So, if the combination of the incoming packet bit mask and the filter bit mask is ‘0’, the packet is not queued and the corresponding destination(s) discard counters are updated. There is a 16-bit discard counter for each combination of port and priority. When a packet is discarded, the appropriate counter is updated. This will provide visibility on the discard intensity and level of discard.

Figure 13: Subflow Best Effort Discard Mechanism within the Input Controller



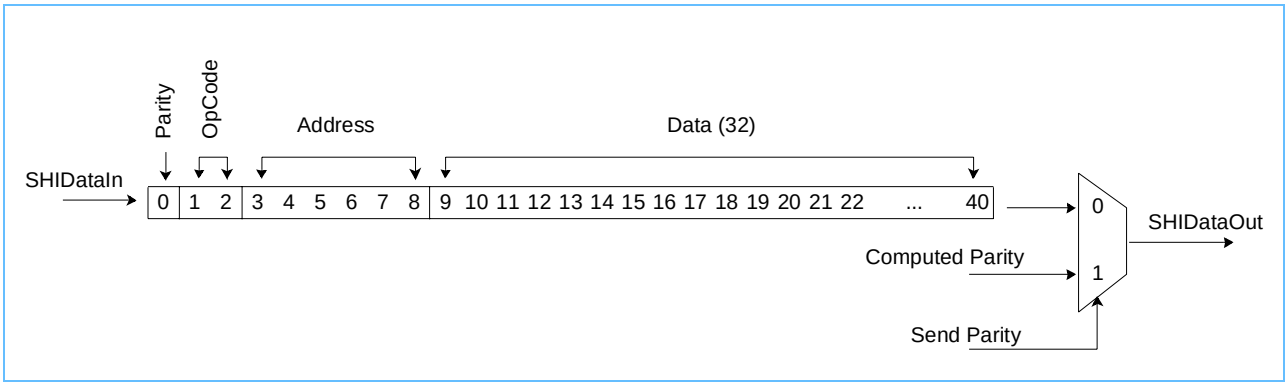
4. Programming Interface

The programming interface is provided by the Serial Host Interface (SHI). The SHI provides an access to all Switch Element internal resources through four signals:

- SHI_Clock: Switch Element input
- $\overline{\text{SHI\_Select}}$: Switch Element input
- SHI_Data_In: Switch Element input
- SHI_Data_Out: Switch Element output

The SHI interface and the internal SHI logic are synchronized to the SHIClock. This clock operates at a lower frequency than the system clock.

4.1 SHI Instruction Register



The instruction scanned into the SHI is decoded into four fields as shown in above: a Data field (32-bit), an Address field (6-bit), an OpCode field (2-bit) and a Parity bit. The *4.1 SHI Instruction Register* on page 52 checks for odd parity of incoming instructions. The Parity bit is contained in the MSB (bit 0) of the *4.1 SHI Instruction Register* on page 52, and is the last bit scanned.

Table 18: SHI OpCode Commands

OpCode	Command	Command Description
00	NOOP	Execute no operation. Used to push the data out of the SHI Instruction Register after a Read or ReadStatus command has been issued.
01	READ STATUS	Allows the reading of the *5.1 Status Register* on page 54. The Status Register content is loaded into the Data field of the SHI Instruction Register. At the same time the Status Register is cleared. NOOP operation is then required to get back the SHI instruction Data field.
10	WRITE REGISTER	Data contained in the Data field of the SHI Instruction Register is written into the register specified by the Address field of the Instruction Register.
11	READ REGISTER	The register content specified by the Address field of the SHI Instruction Register is loaded into the Data field of the SHI Instruction Register. NOOP operation is then required to get back the SHI instruction Data field.

4.2 SHI Parity Error Checking

SHI error checking consists of one bit of parity protection for each instruction scanned into the *4.1 SHI Instruction Register* on page 52. If a parity error is detected on a received instruction, the execution of that instruction is inhibited and the SHI Parity Error bit is set in the *5.1 Status Register* on page 54. All bits of the SHI command are protected by the parity bit (*i.e.* if SHI_Select is active during N number of SHI_Clock cycles, the parity is checked on N bits).

4.3 SHI Parity Generation

Both incoming and outgoing data carry odd parity. This parity is computed at each SHI_Clock cycle as long as the SHI_Select signal is active. The computed parity is sent on SHI_Data_Out when the SHI_Select line is deactivated.

4.4 SHI Operational Protocol

An SHI instruction operation is invoked when the SHI_Select input is b'0'. Data transferred over SHI_Data_In is shifted into the *4.1 SHI Instruction Register* on page 52. Serial data shifted in must begin with the Least Significant Bit (LSB) of the instruction and end with the Most Significant Bit (MSB) of the instruction to be executed. This scan operation is synchronized with the SHI_Clock line. Instructions always execute one cycle after the SHI_Select changes from an active to an inactive state.

SHI Command	OpCode	Operation(s)[1]	Description
Write	10	One single SHI operation of 41 bits.	The Address field specifies which internal register has to be loaded with the content of the Data field.
Read	11	9-bit READ	Address field specifies the register number to read.
	00	33-bit NOOP	Read result + parity is sent over SHIDataOut signal
Read Status	01	3-bit READ STATUS	
	00	33-bit NOOP	*5.1 Status Register* on page 54 content + parity is sent over SHIDataOut signal.

1. The Read and Read Status operations each consists of two SHI operations.

5. Application Registers

5.1 Status Register

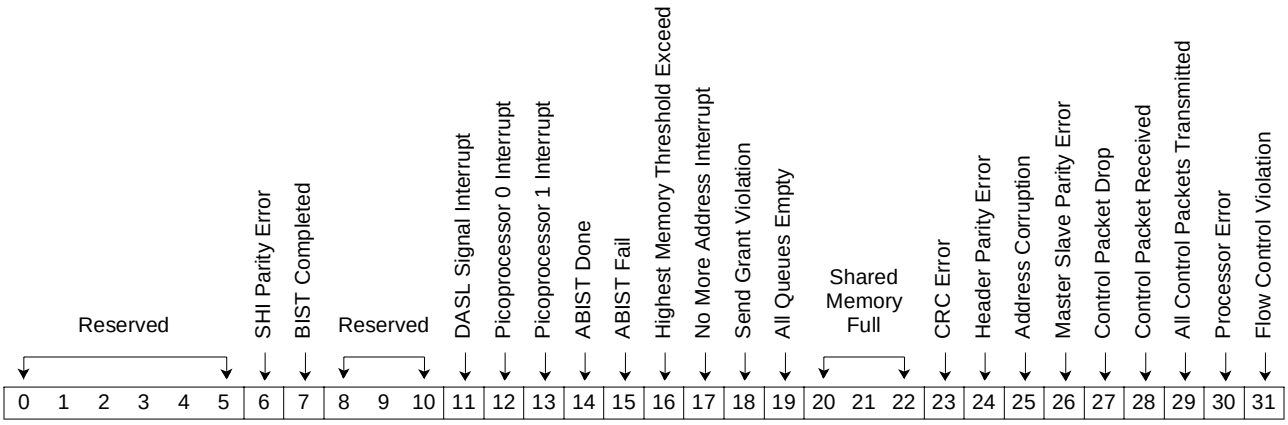
The *5.1 Status Register* on page 54 is accessed when an SHI READ STATUS command is performed.

Almost all bits can generate an interrupt. All the interrupts are maskable by setting the appropriate bit in the *5.3.3 Interrupt Mask Register* on page 61. When a mask bit is set to '1' the corresponding event sets the appropriate bit in the *5.1 Status Register* on page 54. This does not result in the activation of the $\overline{\text{Interrupt}}$ output signal, which is used as an interrupt to the microprocessor.

Note that the $\overline{\text{Interrupt}}$ signal can only be activated if the Global Interrupt Mask bit is not set in the *5.3.2 Reset Register* on page 59 and the OCD Enable bit is set in the same register.

Reset Value ‘0000 0000’

Access Type Read Only and Read/Clear



Bit(s)	Description
0:5	Reserved.
6	SHI Parity Error. Set to '1' when the SHI detects a parity error in the SHI instruction.
7	BIST Completed. Set to ‘1’ when the BIST controller has ended the internal process after a BistRequest command. *5.1 Status Register* on page 54 bits 8 to 31 are forced to ‘0’ while the BIST is running. This bit does not generate an interrupt.
8:10	Reserved.
11	DASL Signal Interrupt. Generated whenever a bit is the *5.4.5 DASL Signal Lost Register* on page 69 changes.
12	Picoprocessor 0 Interrupt. When set to ‘1’b, indicates that the internal processor which handles ports 0 to 15 has generated an interrupt.
13	Picoprocessor 1 Interrupt. When set to ‘1’b, indicates that the internal processor which handles ports 16 to 31 has generated an interrupt.
14	ABIST Done. Set to ‘1’b when the ABIST controller has ended an internal process after an ABIST Start command. This bit does not generate an interrupt.
15	ABIST Fail. Indicates that, after completion of the ABIST process, at least one ABIST check failed on one RAM. This bit is only valid if the ABIST Done bit is asserted. This bit does not generate an interrupt.

Bit(s)	Description
16	Highest Memory Threshold Exceeded. Indicates that shared memory occupancy has crossed the Priority 0 threshold. This is an event (not a status) which occurs when the threshold is exceeded. It does not indicate when the shared memory becomes below the threshold.
17	No More Address Interrupt. This interrupt is generated whenever a packet is received on an input port while no store address is available.
18	Send Grant Violation. Set every time an adapter removes the SND_GRANT for the time defined in the Send Grant Antistreaming field of *5.6.1 Configuration 0 Register* on page 78, see *5.6.16 Send Grant Violation Register* on page 88. This function is enabled only if the link is synchronized and if the Output Queue is enabled.
19	All Queues Empty. Set by an edge detection of all output queues being empty.
20:22	Shared Memory Full. Indicates maximum level of Shared Memory reached between two consecutive read operations. 000: None. 001: Priority 3 full. 010: Priorities 2 and 3 full. 011: Priorities 1, 2, and 3 full. 100: Priorities 0, 1, 2, and 3 full. others Reserved.
23	CRC Error. Set when a data trailer CRC error is detected on an input port. The port is identified via the *5.6.9 CRC Error Port ID Register* on page 84. The number of CRC errors for all ports is reported in the field Trailer CRC Error Count of the .
24	Header Parity Error. Set when a parity error is detected in an incoming packet. The port is identified via the *5.6.10 Header Parity Error Port ID Register* on page 84. The number of parity errors for all ports is reported in the Header Parity Error Count field of the .
25	Address Corruption. Set every time an address corruption is detected. When an address is corrupted, one or more addresses are actually lost from the address space available to store packets.
26	Master/Slave Parity Error. Set when a parity error has been detected on the speed expansion bus which connects a master to a slave device.
27	Control Packet Drop. Set when a Control packet is dropped on an input as a result of a Control packet queue full condition.
28	Control Packet Received. Set when a new Control packet is received. The number of Control packets currently in the Control Receive Queue is indicated by the “Control Packet Counter” register.
29	All Control Packets Transmitted. Set when a Control packet has been successfully transmitted to all the selected destinations.
30	Processor Error. This interrupt is generated when the local processor initiates a new command or operation while the chip internal logic is not ready. This interrupt can be generated in two situations: • Write *5.6.19 Shared Memory Pointer Register* on page 90 while either a Read or Write operation is pending. • Write/Read the *5.6.20 Shared Memory Data Register* on page 91 while either a Read or Write operation is pending.
31	Flow Control Violation. For unicast packets, this interrupt is generated either when a packet is destined to outputs for which no output queue grant has been given in the past N packet cycles (see description of the Flow Control Latency field in the *5.6.1 Configuration 0 Register* on page 78) or when no MemoryGrant has been given in the past N packet cycles. For a multicast packet, only the MemoryGrant information is taken into account for the violation detection. The violating ports are identified by the corresponding bits in the *5.6.12 Flow Control Violation Port ID Register* on page 85.

5.2 Register Definitions

The Internal Application Registers (described below) allow the chip configuration to be specified and status information to be reported. All bits in the Application Registers are set to zero during a Flush operation, unless otherwise specified.

All **Reserved** bits/addresses correspond to non-implemented bits/addresses. This means they are always read as '0' and the write value is “don't care”.

All **Spare** bits are implemented but not used. This means that these bits can be Write/Read but they have no effect on the chip configuration.

Note: Registers 0 to 6 are directly implemented in the SHI internal logic. There is no need to start the PLL and release the flush to access those registers.

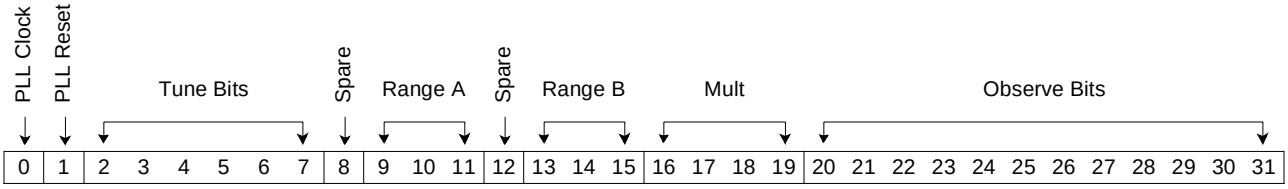
Register Address (hex)	Access	Functional Description
SHI Internal Registers: 00 to 07		
00	Read / Write	PLL Programming Register
01	Read / Write	Reset Register
02	Read / Write	Interrupt Mask Register
03	Read / Write	BIST Counter Register
04	Read / Write	BIST Data Register
05	Read / Write	BIST Select Register
06	Read / Write	Debug Bus Select Register
07		Reserved
DASL Programming Register: 08 to 1D		
08	Read / Write	DASL Output Driver Enable Register
09	Read / Write	Output Port Enable Register
0A	Read / Write	Synchronization Packet Transmit Register
0B	Read / Write	Input Port Enable Register
0C	Read Only	DASL Signal Lost Register
0D	Read / Write	SDC RLOS Enable Register
0E	Read / Write	DASL Synchronization Hunt Register
0F	Read Only	DASL Synchronization Status Register
10	Read / Write	Picoprocessors Instruction Memory Access Register
11	Read / Write	DASL Configuration Register
12	Read Only	DASL Port Error Register
13	Read / Write	DASL Port Quality Mask Register
14	Read Only	DASL Port Quality Register
15	Read / Write	SDC_0 Resources Address Register
16	Read / Write	SDC_0 Resources Control Register
17	Read / Write	SDC_0 Resources Data Register
18	Read Only	SDC_0 Status Register

Register Address (hex)	Access	Functional Description
19	Read / Write	SDC_1 Resources Address Register
1A	Read / Write	SDC_1 Resources Control Register
1B	Read / Write	SDC_1 Resources Data Register
1C	Read Only	SDC_1 Status Register
Flow Control Pins Status Registers: 1D to 1F		
1D	Read / Write	Send Grant Per Priority
1E	Read Only	Send Grant Status Register
1F	Read Only	Receive Grant Status Register
Functional Registers: 20 to 3F		
20	Read / Write	Configuration 0 Register
21	Read / Write	Configuration 1 Register
22	Read / Write	Output Queue Enable Register
23	Read / Write	Input Controller Enable Register
24	Read / Write	Color Detection Disable Register
25	Read / Write	Send Grant Enable Register
26	Read / Write	Force Send Grant Register
27	Read Only	Expected Color Received Register
28	Read Only	CRC Error Port ID Register
29	Read Only	Header Parity Error Port ID Register
2A	Read Only	Error Counters Register
2B	Read Only	Flow Control Violation Port ID Register
2C	Read Only	Control Packet Counter Register
2D	Read Only	Output Queues 00 to 07 Status Register
2E	Read Only	Output Queues 08 to 15 Status Register
2F	Read Only	Output Queues 16 to 23 Status Register
30	Read Only	Output Queues 24 to 31 Status Register
31	Read Only	Color Packet Received Register
32	Read Only	Send Grant Violation Register
33	Read Only	Occupancy Counters Register
34	Read / Write	Shared Memory Pointer Register
35	Read / Write	Shared Memory Data Register
36	Read / Write	Command Register
37	Read / Write	Control Packet Destination Register

Register Address (hex)	Access	Functional Description
38	Read / Write	Bit Map Filter Register
39	Read / Write	Threshold Access Register
3A	Read / Write	Credit Table Access Register
3B	Read / Write	Look Up Tables Access Register
3C	Read / Write	Bitmap Mapping Register
3D	Read / Write	Best Effort Resources Access Register
3E	Read Only	Best Effort Discard Alarm
3F	Read Only	Side Communication Channel Input Reporting

5.3 SHI Internal Registers

5.3.1 PLL Programming Register



Reset Value 'u100 0000 0000 0000 0000 uuuu uuuu uuuu' where 'u' = undefined

Address x'00'

Access Type Read / Write

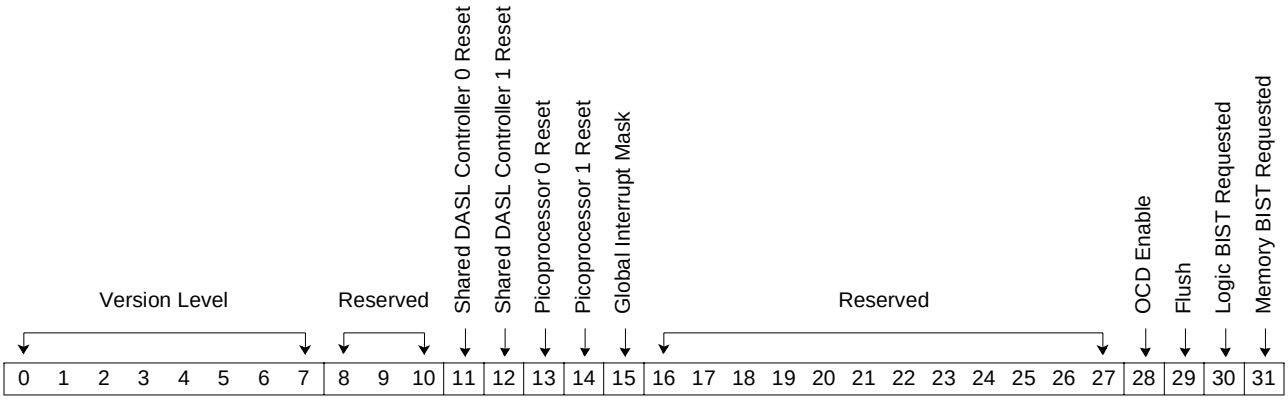
Bit(s)	Description
0	PLL Clock. Read-Only field which indicates that the feedback clock is in phase with the reference clock.
1	PLL Reset. Holds the PLL under reset. Can not be released until the reference clock is stable and the PLL is correctly programmed.
2:7	Tune Bits. Used to optimize the PLL stability and jitter. Must be set to '010101'.
8	Spare.
9:11	Range A. Used to choose the PLL output frequency. Must be set to '111'.
12	Spare.
13:15	Range B. Not used. Must be set to '111'.
16:19	Mult. Defines the PLL feedback divider. Must be set to '0010'.
20:31	Observe Bits. 12 Read-Only bits for testing purposes.

5.3.2 Reset Register

Reset Value ‘0000 0010 0001 1111 0000 0000 0000 0100’

Address x’01’

Access Type Read / Write



Bit(s)	Description
0:7	Version Level. (Read-Only)
8:10	Reserved.
11	Shared DASL Controller 0 Reset. When this bit is set, the DASL logic attached to ports 0 to 15 is forced to reset state. This bit has to be kept asserted until the *5.4.10 DASL Configuration Register* on page 72 is fully programmed.
12	Shared DASL Controller 1 Reset. When this bit is set, the DASL logic attached to ports 16 to 31 is forced to reset state. This bit has to be kept asserted until the *5.4.10 DASL Configuration Register* on page 72 is fully programmed.
13	Picoprocessor 0 Reset. When this bit is set, the internal picoprocessor attached to ports 0 to 15 is forced into a reset state. This bit has to be kept asserted until the corresponding instruction memory is fully programmed.
14	Picoprocessor 1 Reset. When this bit is set, the internal picoprocessor attached to ports 16 to 31 is forced into a reset state. This bit has to be kept asserted until the corresponding instruction memory is fully programmed.
15	Global Interrupt Mask. When asserted, no interrupt is generated to the local processor. The device interrupt pin (active low) is tri-stated and tied up with an external pull-up resistor. However, the *5.1 Status Register* on page 54 bits are still asserted whenever the corresponding event or error occurs.
16:27	Reserved.
28	OCD Enable. When set, all device drivers are enabled if no other configuration disables them. When not set, all drivers are disabled (tri-stated), except for the SHI_Data_Out driver.
29	Flush. When active, all the device logic, except the SHI internal logic, is kept under reset.
30	Logic BIST Requested. When active, the BIST controller starts to run the internal logic BIST as soon as the Flush bit is deactivated. This bit can only be asserted while the Flush bit is active. The BIST completion is reported through the *5.1 Status Register* on page 54.
31	Memory BIST Requested. When active, the memory BIST controller starts to execute memory test as soon as the Flush bit is deactivated. This bit can only be asserted while the Flush bit is active. The memory BIST completion and result is reported in the *5.1 Status Register* on page 54.

5.3.2.1 BIST Execution Sequence

This sequence must be executed after card power up.

1. Activate the $\overline{\text{NotPowerOnReset}}$ input pin and start the SHI clock.
2. Deactivate the $\overline{\text{NotPowerOnReset}}$ input pin after at least three SHI clock cycles.
3. Write the PLL programming register with the correct value, release PLL reset.
4. Wait 500 µs.
5. Set the Logic BIST Requested bit in the *5.3.2 Reset Register* on page 59.
6. Set the PRPG/MISR Data field of the *5.3.5 Bist Data Register* on page 63 with a defined value.
7. Load the Bist Counter register.
8. Set the Shift Speed field in the BIST Select *5.3.6 BIST Select Register* on page 63, as well as the scan chain length.
9. Remove the Flush bit in the *5.3.2 Reset Register* on page 59.
10. Pool the *5.1 Status Register* on page 54 until BIST Completed bit is set.
11. Read the MISRs result in the PRPG/MISR Data field of the *5.3.5 Bist Data Register* on page 63.
12. Set the Flush bit and reset the LogicBIST Requested bit in the *5.3.2 Reset Register* on page 59.

5.3.2.2 Memory BIST Execution Sequence

1. Activate the $\overline{\text{PowerOnReset}}$ input pin and start the SHI clock.
2. Deactivate the $\overline{\text{PowerOnReset}}$ input pin after at least three SHI clock cycles.
3. Write the PLL programming *PLL Programming Register* with the correct value, release PLL reset.
4. Wait 500 µs.
5. Set the “Memory BIST Requested” bit in the *Reset Register*.
6. Remove the Flush bit in the *Reset Register*.
7. Pool the *Status Register* until “ABIST Done” bit is set.
8. Check that “ABIST Fail” is off.
9. Set the Flush bit and reset the “Memory BIST Requested” bit in the *Reset Register*.

5.3.2.3 Reset Sequence

This sequence must be executed after a switch card power up.

1. Activate the $\overline{\text{PowerOnReset}}$ input pin and start the SHI clock.
2. Deactivate the $\overline{\text{PowerOnReset}}$ input pin after at least three SHI clock cycles.
3. Write the PPL Programming *5.3.1 PLL Programming Register* on page 58 with the correct value and release the PLL reset.
4. Wait 500 µs.
5. Set the “MemoryBIST Requested” bit in the *5.3.2 Reset Register* on page 59.
6. Remove the Flush bit in the *5.3.2 Reset Register* on page 59 .
7. Pool the *5.1 Status Register* on page 54 until “ABIST Done” bit is set.
8. Check that the “ABIST Fail” is off.
9. Set the Flush bit and reset the “Memory BIST Requested” bit in the *5.3.2 Reset Register* on page 59.
10. Deactivates the Picoprocessor 0 and 1 Reset bits in the *5.3.2 Reset Register* on page 59.
11. Read Status Register to clear all interrupts.
12. If needed, release the Global Interrupt Mask in the *5.3.2 Reset Register* on page 59.
13. Set the OCD Enable bit in the *5.3.2 Reset Register* on page 59.

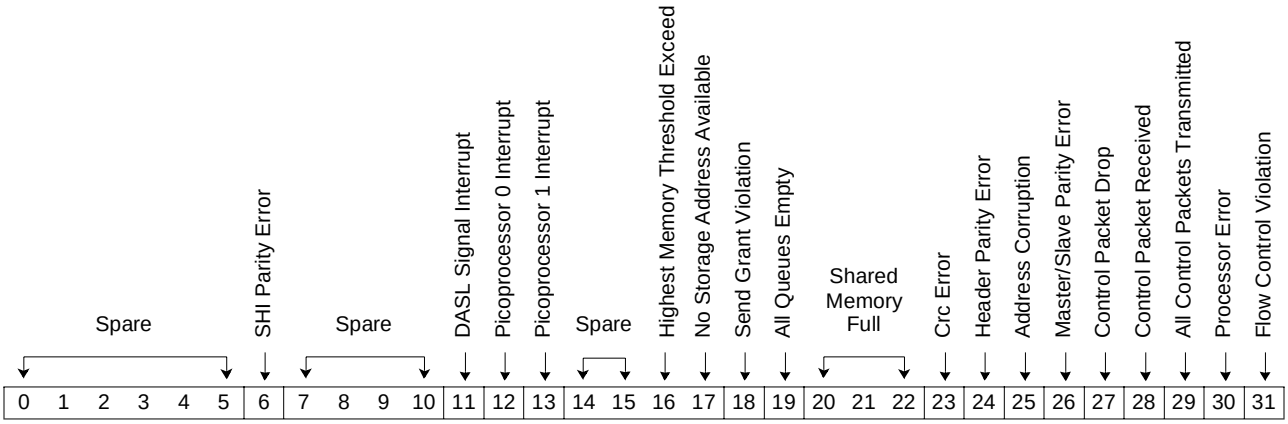
5.3.3 Interrupt Mask Register

Sets masks for the application bits of the *5.1 Status Register* on page 54. When a mask bit is set, the corresponding status bit is still asserted whenever an event is detected. No interrupt is generated.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'02'

Access Type Read / Write



Bit(s)	Description
0:5	Spare.
6	SHI Parity Error. Set to '1' when the SHI detects a parity error in the SHI instruction.
7:10	Spare
11	DASL Signal Interrupt. Generated whenever a bit is the *5.4.5 DASL Signal Lost Register* on page 69 changes.
12	Picoprocessor 0 Interrupt. When set to '1'b, indicates that the internal processor which handles ports 0 to 15 has generated an interrupt.
13	Picoprocessor 1 Interrupt. When set to '1'b, indicates that the internal processor which handles ports 16 to 31 has generated an interrupt.
14:15	Spare.
16	Highest Memory Threshold Exceeded. Indicates that shared memory occupancy has crossed the Priority 0 threshold. This is an event (not a status) which occurs when the threshold is exceeded. It does not indicate when the shared memory becomes below the threshold.
17	No Address
18	Send Grant Violation. Set every time an adapter removes the SND_GRANT for the time defined in the Send Grant Antistreaming field of *5.6.1 Configuration 0 Register* on page 78, see *5.6.16 Send Grant Violation Register* on page 88. This function is enabled only if the link is synchronized and if the Output Queue is enabled.
19	All Queues Empty. Set by an edge detection of all output queues being empty.

Bit(s)	Description
20:22	Shared Memory Full. Indicates for which priority the total number of packets in shared memory has crossed the threshold value. 000: None. 001: Priority 3 full. 010: Priorities 2 and 3 full. 011: Priorities 1, 2, and 3 full. 100: Priorities 0, 1, 2, and 3 full. others Reserved. **Note**: This is an event (not a status) which occurs when the threshold is exceeded. It does not indicate when the shared memory becomes below the threshold.
23	CRC Error. Set when a data trailer CRC error is detected on an input port. The port is identified via the *5.6.9 CRC Error Port ID Register* on page 84. The number of CRC errors for all ports is reported in the field Trailer CRC Error Count of the .
24	Header Parity Error. Set when a parity error is detected in an incoming packet. The port is identified via the *5.6.10 Header Parity Error Port ID Register* on page 84. The number of parity errors for all ports is reported in the Header Parity Error Count field of the .
25	Address Corruption. Set every time an address corruption is detected. When an address is corrupted, one or more addresses are actually lost from the address space available to store packets.
26	Master/Slave Parity Error. Set when a parity error has been detected on the speed expansion bus which connects a master to a slave device.
27	Control Packet Drop. Set when a Control packet is dropped on an input as a result of a Control packet queue full condition.
28	Control Packet Received. Set when a new Control packet is received. The number of Control Packets currently in the Control Receive Queue is indicated by the “Control Packet Counter” register.
29	All Control Packets Transmitted. Set when a Control packet has been successfully transmitted to all the selected destinations.
30	Processor Error. This interrupt is generated when the local processor initiates a new command or operation while the chip internal logic is not ready. This interrupt can be generated in two situations: • Write *5.6.19 Shared Memory Pointer Register* on page 90 while either a Read or Write operation is pending. • Write/Read the *5.6.20 Shared Memory Data Register* on page 91 while either a Read or Write operation is pending.
31	Flow Control Violation. This interrupt is also generated when a packet is destined to outputs for which no output queue grant has been given in the past N packet cycles (see description of the Flow Control Latency field in the *5.6.1 Configuration 0 Register* on page 78. The violating ports are identified by the on page 85.

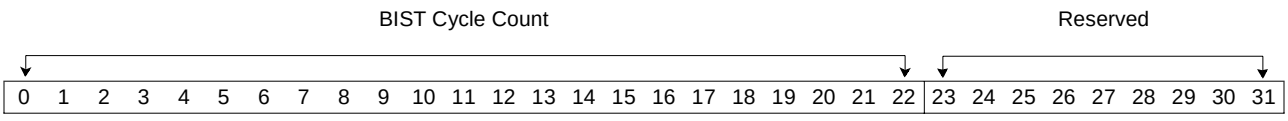
5.3.4 BIST Counter Register

This register specifies the number of BIST cycles to be performed.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’03’

Access Type Read / Write



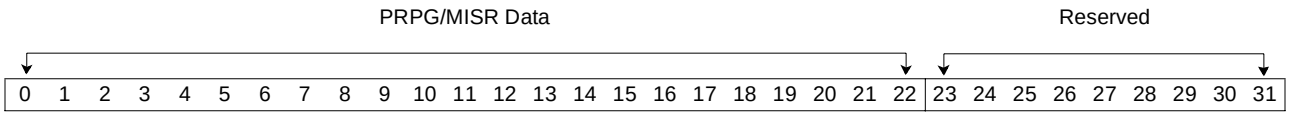
5.3.5 Bist Data Register

This register allows to Write/Read the BIST PRG/MISR registers.

Reset Value undefined

Address x'04'

Access Type Read / Write



5.3.6 BIST Select Register

This register, in addition to the *5.3.5 Bist Data Register* on page 63, allows indirect write/read access to the Internal
PRPG/MISR Registers. Write access is performed in two steps:

1. Write the *5.3.6 BIST Select Register* on page 63 with the BIST Register Select field specifying which BIST register has to be read.
2. Write the *5.3.5 Bist Data Register* on page 63 with the desired value (Selected Internal PRG/MISR Register is loaded).

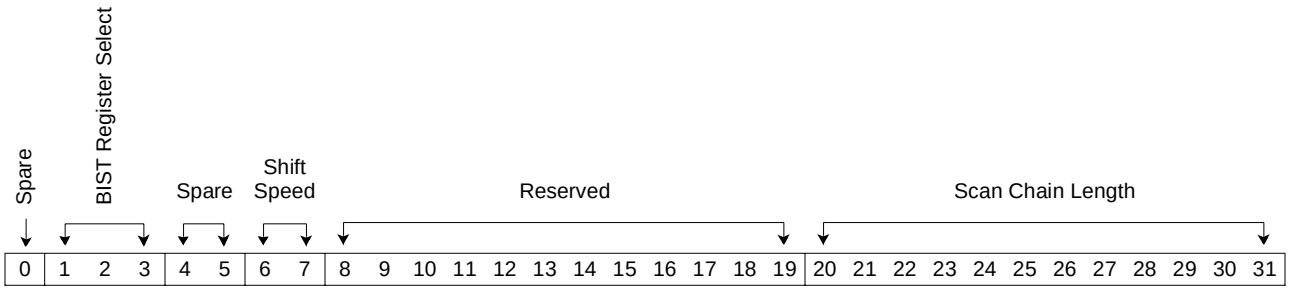
Read access is also performed in two steps:

1. Write the *5.3.6 BIST Select Register* on page 63 with the BIST Register Select field specifying which BIST register has to be written.
2. Read the *5.3.5 Bist Data Register* on page 63. The selected Internal PRPG/MISR Register value is returned.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'05'

Access Type Read / Write



Bit(s)	Description
0	Spare.

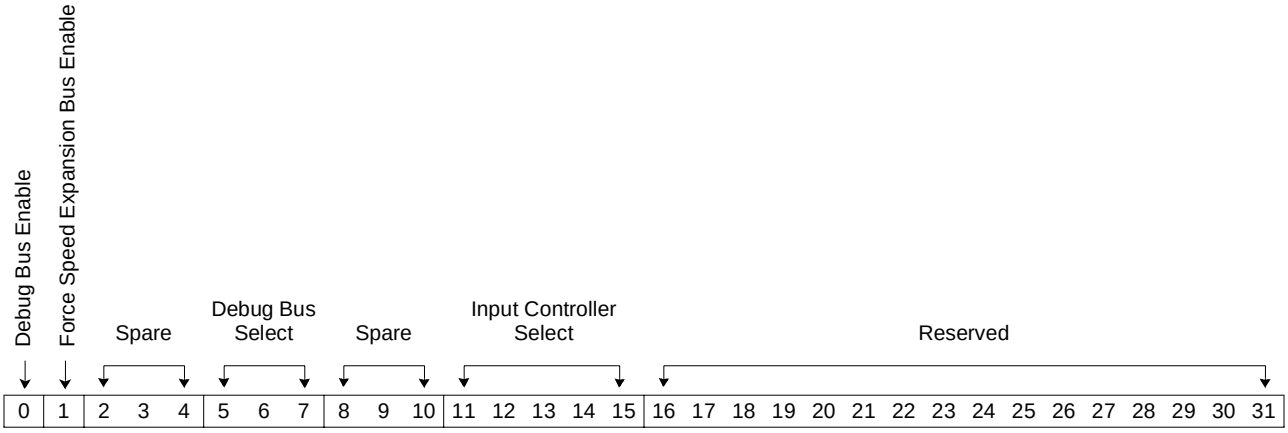
Bit(s)	Description
1:3	Bist Register Select. 000 PRPG0 001 PRPG1 010 PRPG2 011 PRPG3 100 MISR0 101 MISR1 110 MISR2 111 MISR3
4:5	Spare.
6:7	Shift Speed. Defines the delay between a B clock pulse and an A clock pulse while BIST shifting: 00 8 ns 01 16 ns 10 24 ns 11 32 ns
8:19	Reserved.
20:31	Scan Chain Length.

5.3.7 Debug Bus Select Register

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’06’

Access Type Read / Write



Bit(s)	Description
0	Debug Bus Enable. When on, debug bus drivers are enabled.
1	Force Speed Expansion Bus Enable. When on, the Master/slave bus drivers are enabled whatever the module configuration is. **Warning:** This bit must be used carefully in order to avoid the destruction of the module drivers when operating in external speed expansion mode.
2:4	Spare.

Bit(s)	Description
5:7	Debug Bus Selects: 000 Sequencer 001 Address manager 010 Packet Routing Switch general information 011 Input controller selected by Input Controller Select field 100 Picoprocessor 0 instruction address + companion clock 101 Picoprocessor 1 instruction address + companion clock 110 Picoprocessor 0 internal information 111 Picoprocessor 1 internal information
8:10	Spare.
11:15	Input Controller Select
16:31	Reserved.

Depending on the Debug Bus Select value, the following information is provided on the DebugDataOut(0:15) pins.

5.3.7.1 Sequencer Information

DebugDataOut(0:3) SEQ_T_TXARAToSlaves

DebugDataOut(4:7) SEQ_T_TXASAToSlaves

DebugDataOut(8) SEQ_MSSync

DebugDataBusOut(9) SPINP_SeqClk

DebugDataOut(10) SEQ_SeqClk

DebugDataOut(11) ARG_NotPGSelectToSEQ

DebugDataOut(12:15) SEQ_T_OQAStart

5.3.7.2 Address Manager Information

DebugDataOut(0) OQA_ASACountVal

DebugDataOut(1) ADM_MemFreeAck

DebugDataOut(2) ADM_Regenerating

DebugDataOut(3) ARG_NotPGSelectToADM

DebugDataOut(4) ADM_FreeARACtrlPktAck

DebugDataOut(5:15) OQA_ASAToADM

5.3.7.3 Packet Routing Switch General Information

DebugDataOut(0) B_CLK_8ns

DebugDataOut(1)	C_CLK_8ns
DebugDataOut(2)	CT_RAM_NC_CLK
DebugDataOut(3)	OCM_B_CLK
DebugDataOut(4)	DASL_TB1
DebugDataOut(5)	DASL_TC1
DebugDataOut(6)	DASL_TC2
DebugDataOut(7)	MABIST_BCLK
DebugDataOut(8)	MABIST_CCLK
DebugDataOut(9)	MABIST_STCLK
DebugDataOut(10)	MABIST_LBIST
DebugDataOut(11)	PLL_Lock
DebugDataOut(12)	PLL_Reset
DebugDataOut(13)	SynchronousFlush
DebugDataOut(14)	MabistRequest
DebugDataOut(15)	BistRequest

5.3.7.4 Input Controller Information (One among 32 Selected by Input Controller Select Field)

DebugDataOut(0:4)	ByteCounter
DebugDataOut(5)	RowCounter
DebugDataOut(6)	HdrPtyError
DebugDataOut(7)	CRCError
DebugDataOut(8)	Receiving
DebugDataOut(9)	IdleCell
DebugDataOut(10)	ControlCell
DebugDataOut(11)	DataCell
DebugDataOut(12)	ASAValid
DebugDataOut(13)	NSAValid
DebugDataOut(14)	AddrInTime
DebugDataOut(15)	MasterByteCounterVal

5.3.7.5 Picoprocessor 0 Instruction Address + Companion Clock; Picoprocessor 1 Instruction Address + Companion Clock

DebugDataOut(0:10)	Picoprocessor address
DebugDataOut(11:14)	Not used
DebugDataOut(15)	Picoprocessor clock

5.3.7.6 Picoprocessor 0 Internal Information; Picoprocessor 1 Internal Information

DebugDataOut(0:1)	from IDCD_CC unit
DebugDataOut(2)	MUXR_CNTL
DebugDataOut(3)	MUXQ_CNTL
DebugDataOut(4)	MUXA_CNTL
DebugDataOut(5)	ImmediateDataFromInstruction
DebugDataOut(6:7)	DataWidth
DebugDataOut(8:9)	ALU_B_Select
DebugDataOut(10:11)	ALU_A_Select
DebugDataOut(12)	AccessPY
DebugDataOut(13)	PY_AutoIncrement
DebugDataOut(14)	AccessPX
DebugDataOut(15)	PX_AutoIncrement

5.4 DASL Programming Registers

The DASL logic contains 32 DASL receivers, 32 DASL transmitters, and two Shared DASL controllers (SDC). Each SDC contains a picoprocessor with an instruction memory as well as local store, and a set of IO registers and hardware which assist in order to control the DASL receivers. SDC_0 handles ports 0 to 15, while SDC _1 handles ports 16 to 31. The following set of registers is used to access/control the whole DASL logic.

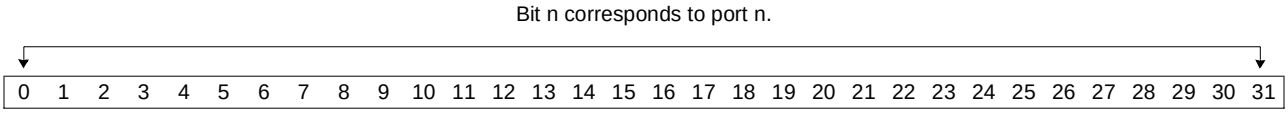
5.4.1 DASL Output Drivers Enable Register

A DASL output driver corresponding to a selected port is enabled when on, and if the OCD Enable bit is set in the *5.3.2 Reset Register* on page 59 and $\overline{\text{Fully Inserted}}$ input pin is active (low level). The DASL output driver is tri-stated when off.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’08’

Access Type Read / Write



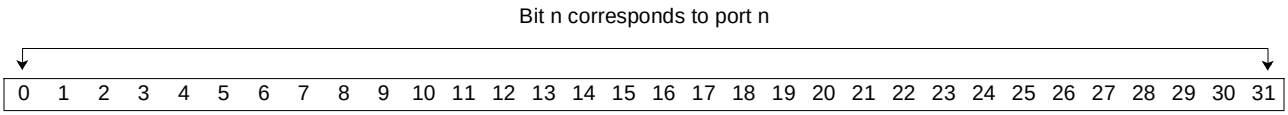
5.4.2 Output Port Enable Register

When on, normal data transmission is allowed on the corresponding port. When off, the data sent by the port is forced to ‘0’.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’09’

Access Type Read / Write



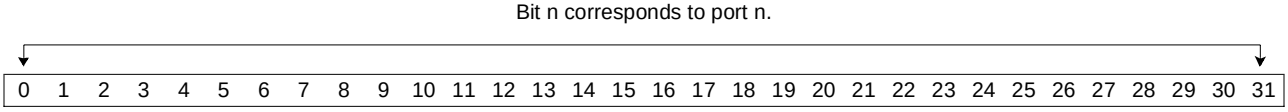
5.4.3 Synchronization Packet Transmit Register

Specifies which output ports have to transmit Sync Packets.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’0A’

Access Type Read / Write



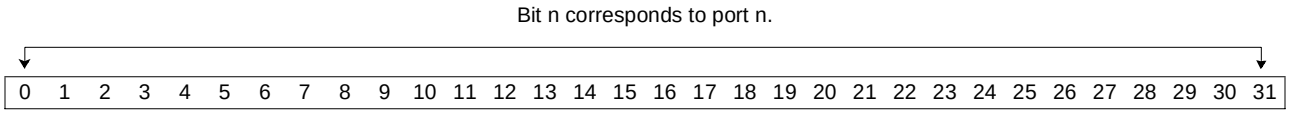
5.4.4 Input Port Enable Register

When on, the corresponding input port is enabled and DASL alignment is allowed on that port. When off, alignment is not possible and thus no packet can be received on the corresponding port.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’0B’

Access Type Read / Write

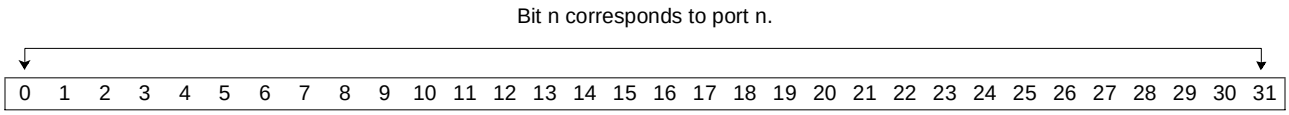


5.4.5 DASL Signal Lost Register

Reset Value Undefined

Address x’0C’

Access Type Read / Only



Bit(s)	Description
1:31	Bit n corresponds to port n. 1 No signal is detected on at least one DASL line of the corresponding port. 0 A correct signal is detected on all lines of the corresponding port. Any change in any of those bits generates a DASL Signal Interrupt if not masked

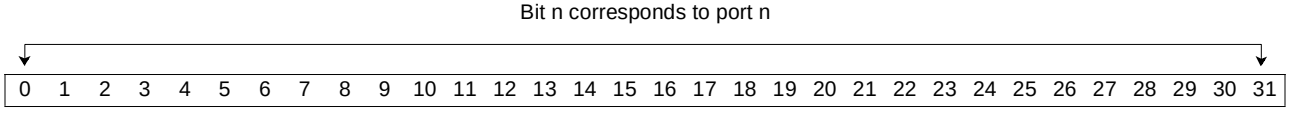
5.4.6 SDC Rlos Enable Register

When on, the Shared DASL Controller automatically reacts to a receiver lost signal assertion on a port receiver, by resetting the port phase alignment state machine. In this case, the port receivers have to be re-synchronized. When off, the SDC does not react to a RLOS condition on a port receiver. An RLOS condition is always reported by the DASL signal lost register.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’0D’

Access Type Read / Write



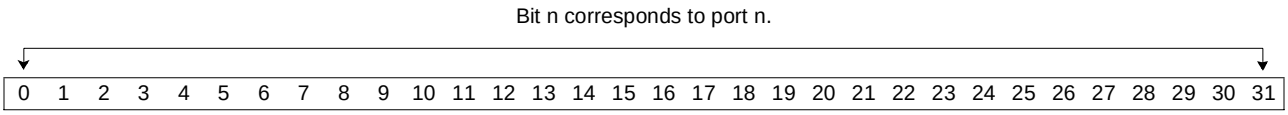
5.4.7 DASL Synchronization Hunt Register

A raising edge requests the picoprocessor to perform bit phase alignment and packet delineation. When the picoprocessor detects a raising edge, the corresponding bit in the *5.4.8 DASL Synchronization Status Register* on page 70 is cleared.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’0E’

Access Type Read / Write



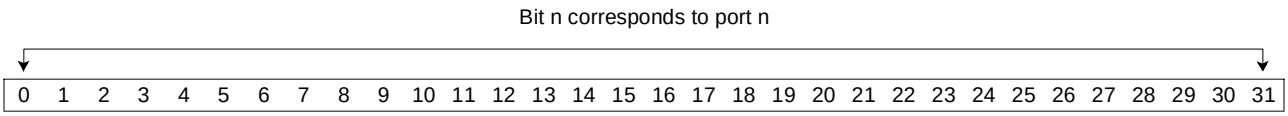
5.4.8 DASL Synchronization Status Register

When on (’1’): the bit phase alignment and packet delineation are completed and normal packet reception is possible on the corresponding port.
When off (’0’): the synchronization sequence initialed by a synchronization hunt command (through the *5.4.7 DASL Synchronization Hunt Register* on page 70) is not yet completed, or that the port is disabled.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’0F’

Access Type Read Only

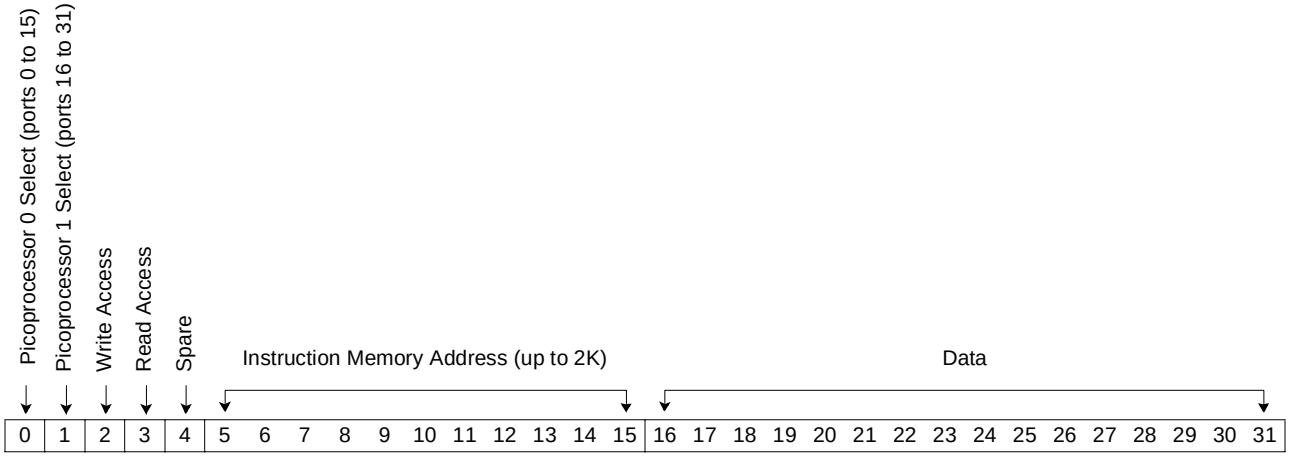


5.4.9 Picoprocessor Instruction Memory Access Register

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'10'

Access Type Read / Write



Bit(s)	Description
0	Picoprocessor 0 Select:. Access request to/from picoprocessor 0 instruction memory.
1	Picoprocessor 1 Select:. Access request to/from picoprocessor 1 instruction memory.
2	Write Access. Write access to both picoprocessor at the same time is allowed.
3	Read Access. Read access is performed in two steps: a. Write the *5.4.9 Picoprocessor Instruction Memory Access Register* on page 71 with either bit 0 or 1 set (both are not allowed at the same. Address field specifies the instruction memory address, Data field is don't care b. Read the *5.4.9 Picoprocessor Instruction Memory Access Register* on page 71. The read result is returned in the Data field.
4	Spare.
5:15	Instruction Memory Address (up to 2k).
16:31	Data.

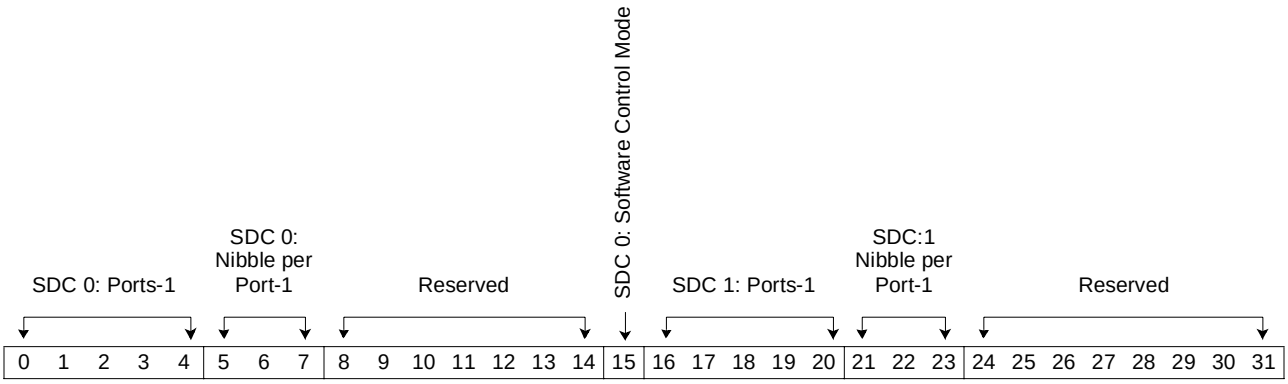
5.4.10 DASL Configuration Register

This register must be loaded before the Picoprocessor Reset is released in the *5.3.2 Reset Register* on page 59.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'11'

Access Type Read / Write



Bit(s)	Description
1:4	SDC 0/1 Number of Port-1. The number of physical ports handled by the picoprocessor. Must be set to 'x0F'
5:7	SDC 0/1 Number of Nibble per Port-1. The number of physical DASL links per port. Must be set to 3.
8:15	Reserved. Must be set to x'01'
16:20	SDC 1: Ports-1.
21:23	SDC 1: Nibble per Port-1.
24:31	Reserved. Must be set to x'01

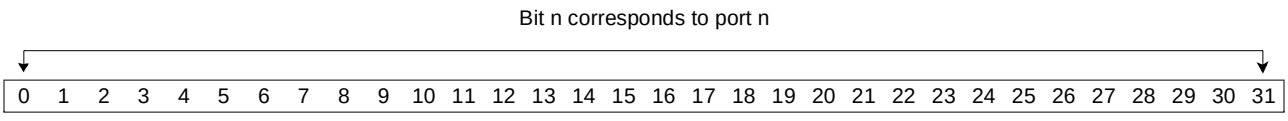
5.4.11 DASL Port Error Register

When on: the bit indicates that an error has been detected by the DASLRX state machine on the corresponding port.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'12'

Access Type Read Only



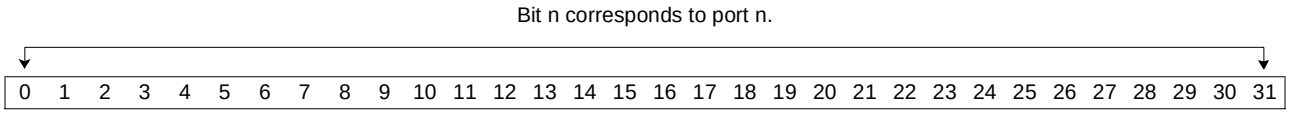
5.4.12 DASL Port Quality Mask Register

When on: the corresponding bit in the DASL Port Quality Register is reset.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'13'

Access Type Read / Write



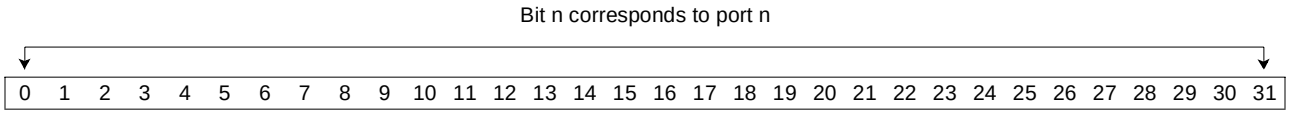
5.4.13 DASL Port Quality Register

When on: the picoprocessor has detected a minimum eye violation on a port. A minimum eye violation indicates that one or more of the DASL links which make up the port, is not operating within the specification.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'14'

Access Type Read Only



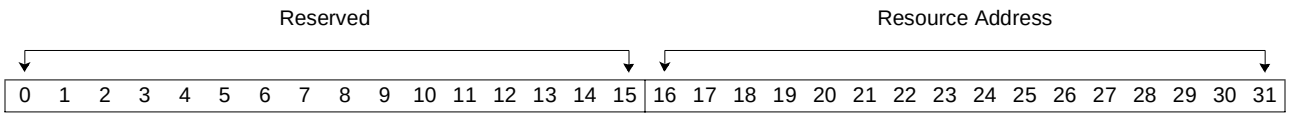
5.4.14 SDC Resources Address Register

Register x'15' is dedicated to SDC 0 (ports 0 to 15) while Register 'x19' is dedicated to SDC 1 (ports 16 to 31). These two registers are not used in normal operational mode.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'15', x'19'

Access Type Read / Write



Bit(s)	Description
0:15	Reserved.
16:31	Resource Address: Specifies the resource address for a read or a write operation. This allows indirect access to the local store and picoprocessor lO registers.

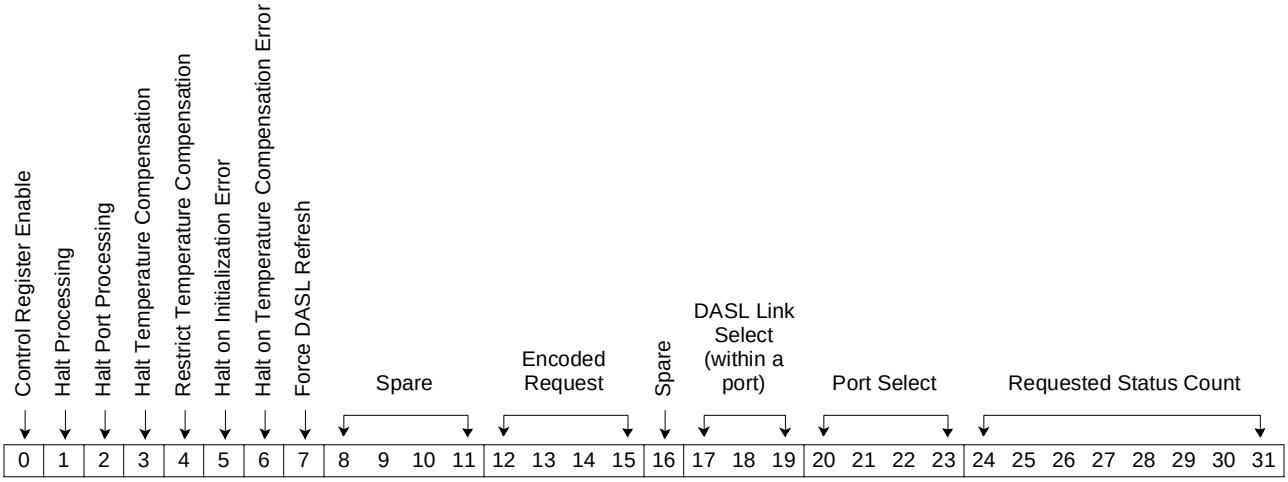
5.4.15 SDC Resources Control Register

Register x'16' is dedicated to SDC 0 (ports 0 to 15) while Register x'1A' is dedicated to SDC 1 (ports 16 to 31). Those registers are not used in operational mode.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'16', x'1A'

Access Type Read / Write



Bit(s)	Description
0	Control Register Enable. A high on this signal will enable the register content. When this bit is low, the rest of the bits will be ignored except the Requested Status Count field.
1	Halt Processing.
2	Halt Port Processing:.
3	Halt Temperature Compensation.
4	Restrict Temperature Compensation.
5	Halt On Initialization Error.
6	Halt On Temperature Compensation Error.
7	Force DASL Refresh.
8:11	Spare.
12:15	Encoded Request. Values indicate which services the picocode should perform. Valid values: 0000 No operation 0001 Request a read to the address given by the *5.4.14 SDC Resources Address Register* on page 73 0010 Request a write to the address given by the *5.4.14 SDC Resources Address Register* on page 73 0101 Load Sample Memory 1000 Request a delay line sample from the DASL specified by the Port Select and DASL Link Select fields 1001 Update the DASL Data Structure with the *5.4.16 SDC Resources Data Registers* on page 75 input for the DASL receiver specified by the Port Select and DASL Link Select fields. others Reserved

Bit(s)	Description
16	Spare.
17:19	DASL Link Select (within a port). An encoded value which selects the DASL link within the selected port for a given action.
20:23	Port Select. An encoded value which selects the port for a given action. A request which utilizes the Port Select field will get processed at the next service time for the selected port. If the supplied port value is out of range, the request will be processed for Port 0.
24:31	Requested Status Count. This field will be read by the picoprocessor, incremented, and written back out of the Updated Status Count field of the *5.4.17 SDC Status Register* on page 76. This behaves as a “keep alive” mechanism in order for the application to monitor the state of the picoprocessor.

5.4.16 SDC Resources Data Registers

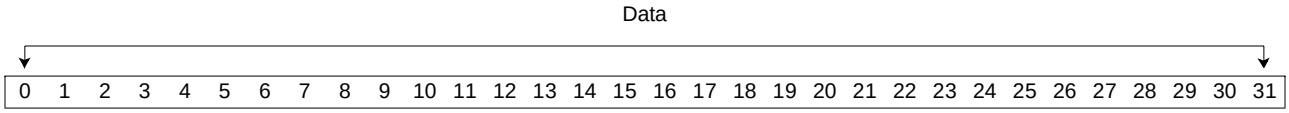
Register x’17’ is dedicated to SDC 0 (ports 0 to 15) while Register x’1B’ is dedicated to SDC 1 (ports 16 to 31). Those registers are not used in operational mode.

This register contains the SDC resource data for read and write operations.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’17’, x’1B’

Access Type Read / Write



5.4.17 SDC Status Register

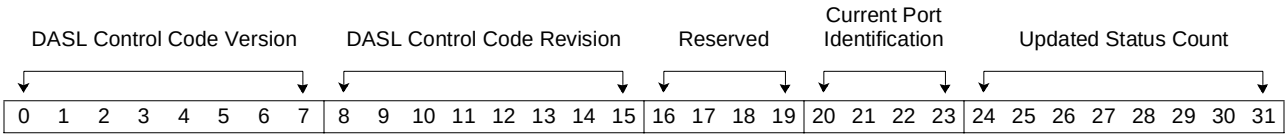
Register x'18' is dedicated to SDC 0 (ports 0 to 15) while Register x'1C' is dedicated to SDC 1 (ports 16 to 31).

This register is loaded by the functional picocode when the corresponding Picoprocessor 0/1 Reset bit is released in the *5.3.2 Reset Register* on page 59.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'18', x'1C'

Access Type Read Only



Bit(s)	Description
0:7	DASL Control Code Version. Which version of code is running.
8:15	DASL Control Code Revision. Which version of code is running.
16:19	Reserved
20:23	Current Port Identification. This field displays the port number which is currently processed by the Shared DASL Controller.
24:31	Updated Status Count. Written by the picoprocessor to be one greater than the value present in the Requested Status Count field of the *5.4.15 SDC Resources Control Register* on page 74. This serves as a "keep alive" mechanism to allow the application to monitor the state of the picoprocessor.

5.5 Flow Control Pins Status and Setting Registers

5.5.1 Send Grant Per Priority

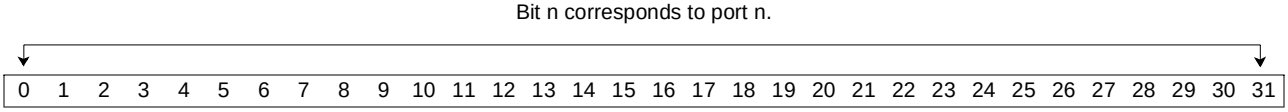
When SendGrantPerPriority bit is not active in the *5.6.2 Configuration 1 Register* on page 80, this indicates the current level of the switch send grant input signals coming from the attached devices.

When SendGrantPerPriority bit is active, a one in this register indicates that the incoming signal contains the Send Grant Per Priority framing '00001' and at least the Send Grant for one priority.

Reset Value Undefined

Address x'1E'

Access Type Read Only



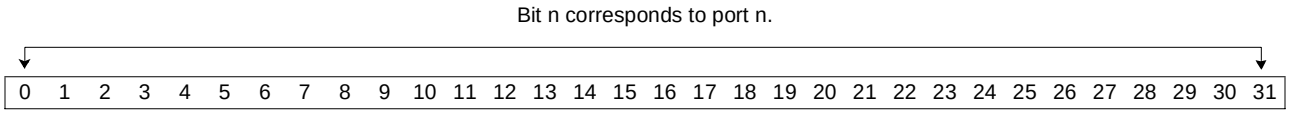
5.5.2 Send Grant Status Register

This indicates the current level of the switch's receive grant input signals.

Reset Value Undefined

Address x'1F'

Access Type Read Only



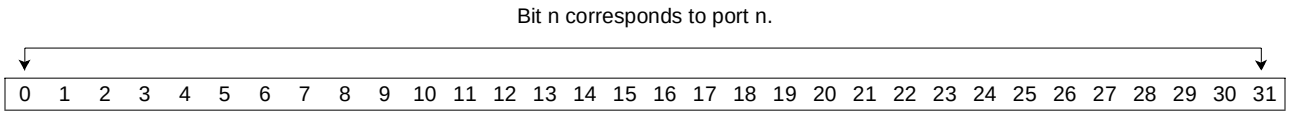
5.5.3 Receive Grant Status Register

When active, it indicates that the incoming send grant signals carry the send grant information for several priorities. When this bit is set to '0', the send grant signals are used as level and does not carry information per priority.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'1D'

Access Type Read / Write



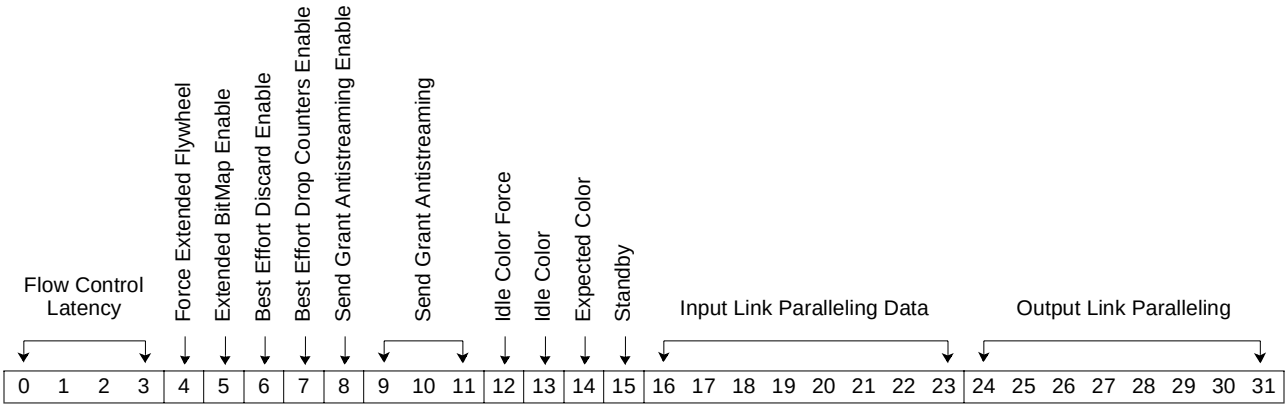
5.6 Functional Registers

5.6.1 Configuration 0 Register

Reset Value '0000 0000 0000 0001 0000 0000 0000 0000'

Address x'20'

Access Type Read / Write



Bit(s)	Description
0:3	Flow Control Latency. When active, flow control is checked by the input controllers. If a received packet is destined to an output for which no grant has been given in the past number of packet cycles defined by N, a Flow Control Violation interrupt is asserted. The packet is discarded. 0 Function is disabled. 1 N = 8 2 N =10 3 N=12 ... 14 N=34 15 N=36
4	Force Extended Flywheel. When on, the Packet qualifier bit 0 is forced to 1 on outgoing idle cells (used for test purpose)
5	Extended BitMap Enable. When on, indicates that the packet qualifier bit 0 of incoming data packet holds the "Extended Bitmap" information.This bit also enables the "Extended OQ grant flywheel".
6	Best Effort Discard Enable. When active , the best effort traffic can be discarded depending on the best effort discard thresholds and the level of best effort counters.
7	Best Effort Drop Counters Enable. When active, all the best effort packets discarded by the input controllers are count in best effort drop counters.
8	Send Grant Antistreaming Enable. When active, the Send Grant antistreaming function is enabled.

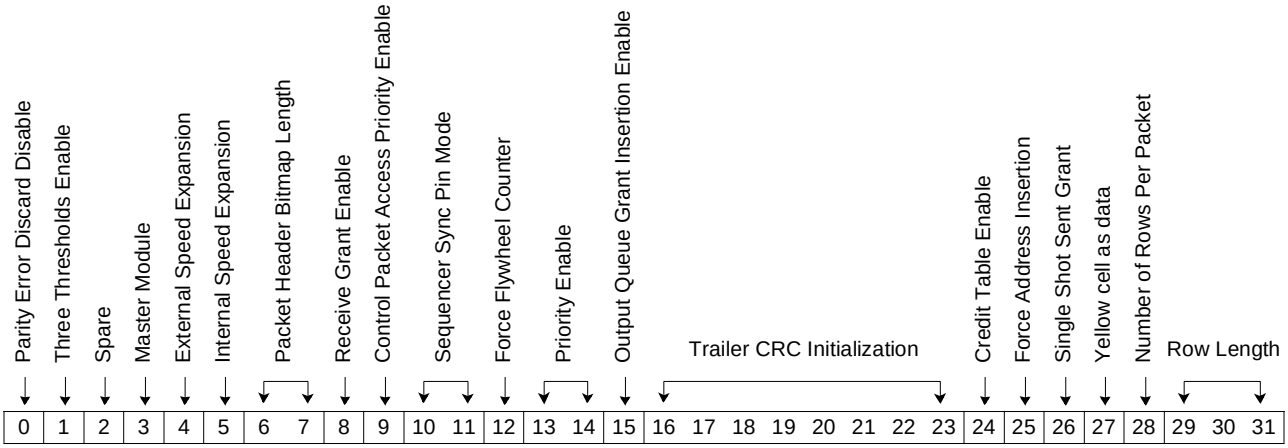
Bit(s)	Description
9:11	Send Grant Antistreaming. This function provides a protection when the adapter removes the “send grant” for a long time. If the “send grant” is not raised for N contiguous packet cycles, it is internally forced active until the adapter raises it again. The Send Grant Antistreaming field is encoded as follow: ‘000’ N=16 ‘001’ N=32 ‘010’ N=64 ‘011’ N=128 ‘100’ N=256 ‘101’ N=512 ‘110’ N=1024 ‘111’ N=2048
12	Idle Color Force. When active, all Idle Packets will be transmitted with the color specified by the Idle Color bit, regardless of the Expected Color bit setting. When the color mechanism is not used, this bit must be set to 1.
13	Idle Color. Specifies the color to give to all Idle Packets, when the Idle Color Force is set: 0 Blue Idle Packets 1 Red Idle Packets
14	Expected Color. Specifies the expected color of incoming packets after a Color Clear Command is initiated through the *5.6.21 Command Register* on page 91. 0 Blue packets 1 Red packets
15	Standby.
16:23	Input Physical Port Bundling. This field defines which input ports are merged to build an logical OC-192 links. Bit 0 Ports 00 to 03 are processed as an OC-192 logical link. Bit 1 Ports 04 to 07 are processed as an OC-192 logical link. Bit 2 Ports 08 to 11 are processed as an OC-192 logical link. Bit 3 Ports 12 to 15 are processed as an OC-192 logical link. Bit 4 Ports 16 to 19 are processed as an OC-192 logical link. Bit 5 Ports 20 to 23 are processed as an OC-192 logical link. Bit 6 Ports 24 to 27 are processed as an OC-192 logical link. Bit 7 Ports 28 to 31 are processed as an OC-192 logical link.
24:31	Output Physical Port Bundling. This field defines which output ports are merged to build an logical OC-192 links. The encoding is the same as Input Link Paralleling field.

5.6.2 Configuration 1 Register

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’21’

Access Type Read / Write



Bit(s)	Description
0	Parity Error Discard Disable. When active, incoming packets with invalid header parity are not discarded by the input controllers, and are received as normal packets but a Header Parity Error interrupt is raised, and the Packet Header Parity Error Count field is incremented in the on page 85.
1	Three Thresholds Enable. When active, the output queue grant information transmitted in the outgoing packets is only passed for priorities 0, 1 and 2. This means that after grants of priority 2 have been transmitted, the grants for priority 0 are transmitted in the next packet. This allows a reduction of the update time of the output queue grant information with four priorities. Also, when this bit is set, the output queue thresholds for priority 2 and 3 have to be set equal, as well for the shared memory thresholds for priority 2 and 3. When this bit is set to ‘0’, grants for all 4 priorities are transmitted if the Priority Enable field (of this register) is set to ‘11’.
2	Spare
3	Master Module. Defines which is the master module when using the device in external speed expansion.
4	External Speed Expansion. Enables 2-way external speed expansion. This configuration bit has to be specified for both master and slave modules.
5	Internal Speed Expansion. Enables 2-way internal speed expansion.
6:7	Packet Header Bitmap Length. This field defines the length of the packet header bit map as follow : ‘00’ 1 byte ‘01’ 2 bytes ‘10’ 3 bytes ‘11’ 4 bytes
8	Receive Grant Enable.
9	Control Packet Access Priority Enable. This bit guarantees access time to the shared memory by the local processor for read and write accesses. When set, a shared memory access by the local processor is performed within a maximum of three LU time units after the command has been issued. It only has to be set for row length of 16 or 32 bytes. For other row lengths, the sequencer operation guarantees local processor access time to the shared memory independently of the setting of this bit.

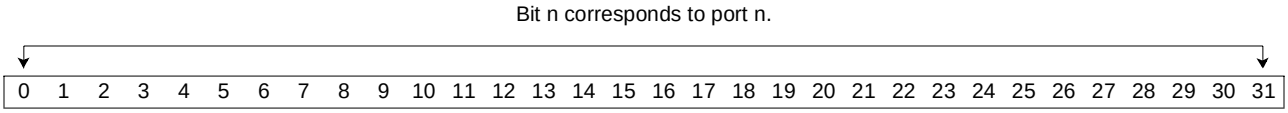
Bit(s)	Description
10:11	Sequencer Sync Pin Mode. Specifies the operation of the SEQ_SYNC pin. 00 SEQ_SYNC is tri-stated, and the chip sequencer runs on its own. 01 SEQ_SYNC is an output signal generated by the internal sequencer, which is running on its own. 10 SEQ_SYNC is an input signal on which the internal sequencer synchronizes. 11 Reserved.
12	Force Fly Wheel. When on the flywheel counter is forced to the value defined in priority enable filed (Used for test purpose)
13:14	Priority Enable. This controls the cycling process of Output Queue Grant information to the attached adapter. 00 priority 0 only enabled. 01 priority 0 and 1 enabled. 10 priority 0, 1 and 2 enabled. 11 priority 0, 1, 2 and 3 enabled.
15	Output Queue Grant Insertion Enable. When asserted, the insertion of the output queue grant in the outgoing packet header byte is performed. When deasserted, output queue grant is not inserted, and the received header is transmitted unchanged.
16:23	Trailer CRC Initialization. Specifies the initialization value to be used for the 8-bit Idle Packet trailer CRC check. LU LengthCRC Init Register in Hex 16 D0 17 DD 18 04 19 FA 20 07 32 9B 34 67 36 2E 38 61 40 5A
24	Credit Table Enable. This bit enables the scanning of the credit table by the switch output control logic. It allows the application to not initialize the credit table if the function is not used.
25	Force Address Insertion. When active the input controller inserts its address into the H2 field (Used for test purpose).
26	Single Shot Sent Grant. When this bit is set to ‘0’, grants for all 4 priorities are transmitted if the Priority Enable field of Configuration 1 Register is set to 3 (Used for test purpose).
27	Yellow cell as data. When active, an incoming Yellow cell is processed as data cell. This means the LU doesn't contains the CRC. When inactive, the incoming Yellow cell is processed as Idle cell. This means the LU contains the CRC. Therefore, the CRC is checked.
28	Number Of Rows Per Packet. Defines the number of rows contained in a logical unit. 0 1 row 1 2 rows
29:31	Row Length. Provides the length of a row in the shared memory. ‘000’ 16 bytes ‘001’ 17 bytes ‘010’ 18 bytes ‘011’ 19 bytes ‘100’ 20 bytes others reserved

5.6.3 Output Queue Enable Register

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’22’

Access Type Read / Write



Bit(s)	Description
0:31	Bit n corresponds to port n. When active, the corresponding output queue is enabled and packets destined for that output are enqueued. When set to 0, the output queue is disabled and the following actions are taken: 1. Unicast packets destined to a disabled output queue are discarded and multicast packets are enqueued only in enabled output queues. 2. A disabled output queue is “slow flushed”. Addresses are dequeued and recycled as during normal operation. The slow flush takes place regardless of SND_GRANT signal, al long as the queue is disabled. 3. The output queue grant corresponding to a disabled queue are forced to ‘1’. 4. Idle packets are transmitted if the corresponding bit is set in the *Output Drivers Enable Register* and in the *5.4.2 Output Port Enable Register* on page 68.

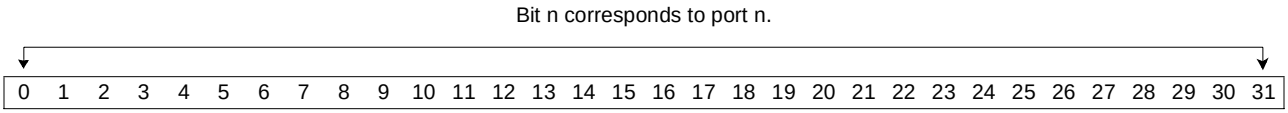
5.6.4 Input Controller Enable Register

When active, normal data and Control Packet reception is allowed on that port. When set to 0, neither data nor Control Packets are received.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’23’

Access Type Read / Write



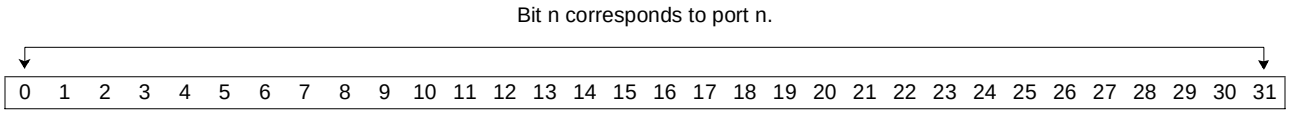
5.6.5 Color Detection Disable Register

When active, the color detection mechanism is disabled for that input and the corresponding bit in the *5.6.8 Expected Color Received Register* on page 84 is set. See section on Color Mechanism.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’24’

Access Type Read / Write



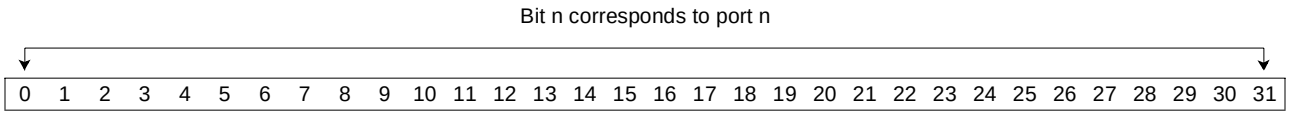
5.6.6 Send Grant Enable Register

When active, normal send grant signal decoding is performed, and packet transmission is possible only if the retrieved send grant information is active. When set to 0, packets are not transmitted, regardless of the value of the send grant signal except if the corresponding bit is set in the *5.6.7 Force Send Grant Register* on page 83.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’25’

Access Type Read / Write



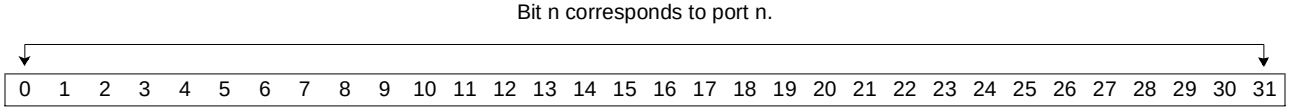
5.6.7 Force Send Grant Register

When active, the device control logic considers that the send grant signals are always active whatever the *5.6.6 Send Grant Enable Register* on page 83 specifies and whatever the send grant pin values indicate.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’26’

Access Type Read / Write



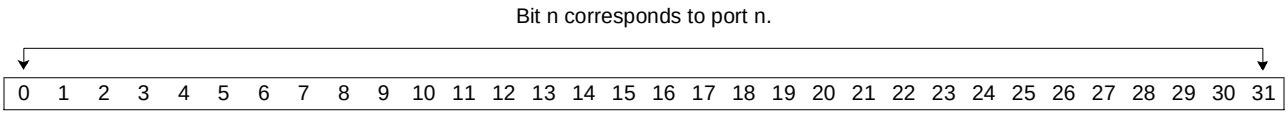
5.6.8 Expected Color Received Register

When active, it either indicates that the expected color has been received on that input, since the last Color Clear command, or that the corresponding bit in the *5.6.5 Color Detection Disable Register* on page 83 is set. When a bit is set to 0, it indicates that the color opposite to the expected color is still being received on that input.

Reset Value Undefined

Address x'27'

Access Type Read Only



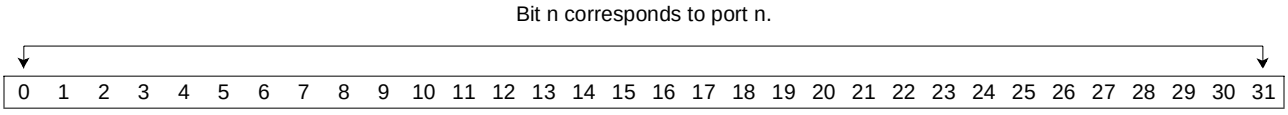
5.6.9 CRC Error Port ID Register

Indicates which input port has detected a trailer CRC error within an idle cell. This register is cleared on read.

Reset Value '0000 0000 0000 0000 0000 0000 0000 000'

Address x'28'

Access Type Read Only



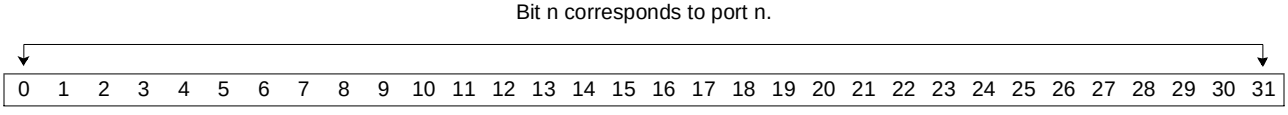
5.6.10 Header Parity Error Port ID Register

Indicates which input port has detected a packet header parity error. This register is cleared on read.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'29'

Access Type Read / Only

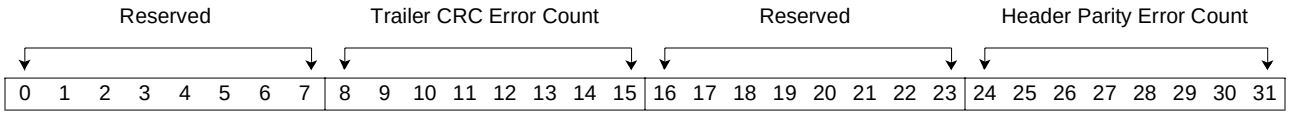


5.6.11 Error Counters Register

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'2A'

Access Type Read Only



Bit(s)	Description
1:7	Reserved.
8:15	Trailer CRC Error Count. provides the total number of CRC errors for all input ports since the last read. This counter freezes once it reaches x'FF', and is cleared when read.
16:23	Reserved.
24:31	Header Parity Error Count. provides the total number of packet header parity errors for all input ports since the last read. This counter freezes once it reaches x'FF', and is cleared when read.

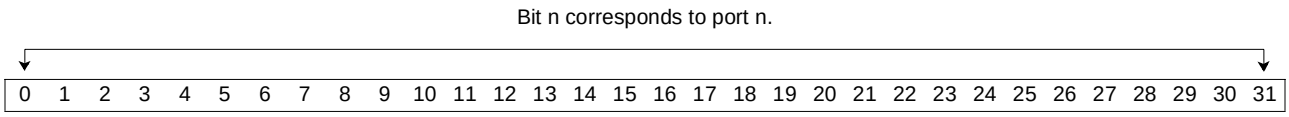
5.6.12 Flow Control Violation Port ID Register

Indicates which port has generated Flow Control Violation interrupt. This register is cleared on read.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'2B'

Access Type Read Only

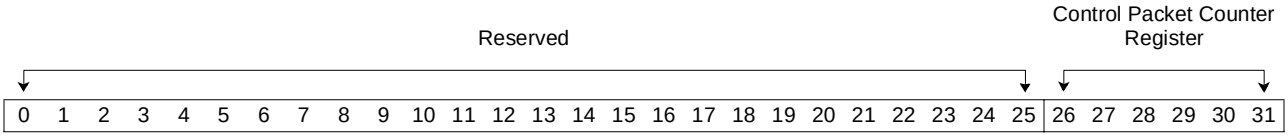


5.6.13 Control Packet Counter Register

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’2C’

Access Type Read Only



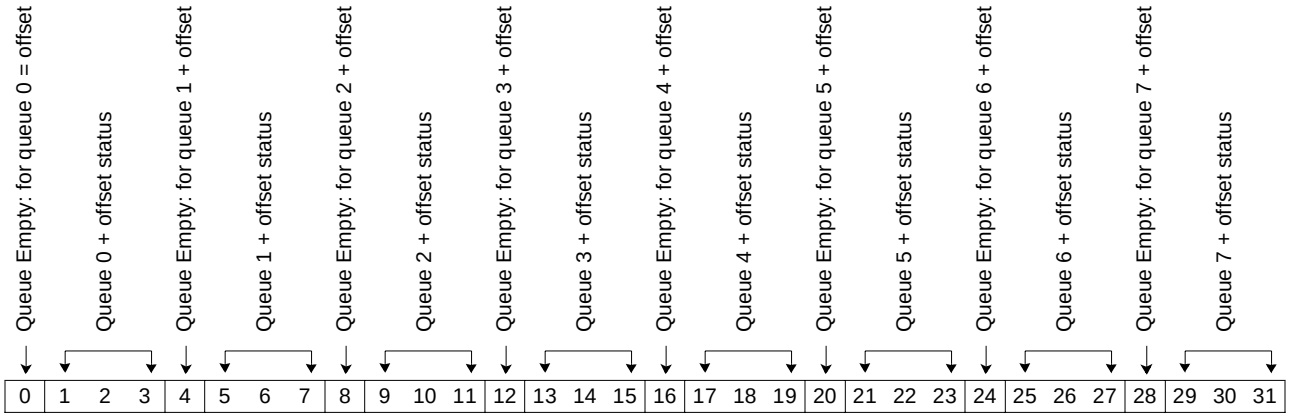
Bit(s)	Description
0:25	Reserved.
26:31	Control Packet Counter. Indicates the number of Control Packets currently in the Control packet receive queue. Value range is 0 to 32.

5.6.14 Output Queues Status Registers 1-4

These four registers provide the status of the 32 output queues. Register 1 shows the status of the 32 output queues. Each register is composed of eight 4-bit ranges that describe a particular output queue status: Register 1 shows the status of output queues 0-7; register 2 of queues 8-15; register 3 of queues 16-23; and register 4 of queues 24-31. Within each queue's 4-bit range, the first bit is the Output Queue Empty field, which indicates whether the queue has been emptied since the last read operation, and the other three bits are the status field, which indicates the maximum level reach by the output queue since the last read operation. Because the maximum level occurs when the threshold is exceeded, the status field does not indicate when the queue occupancy is below the threshold.

The offsets referred to in the bit-by-bit descriptions of these four registers are shown with the addresses below:

Reset Value	'1000 1000 1000 1000 1000 1000 1000 1000'		
Address	Output Queues 0-7	x'2D'	Offset = 0
	Output Queues 8-15	x'2E'	Offset = 8
	Output Queues 16-23	x'2F'	Offset = 16
	Output Queues 24-31	x'30'	Offset = 24
Access Type	Read Clear		



Bit(s)	Description
0	Queue Empty for Queue 0 + offset. When set to '1', the corresponding output queue is empty.
1:3	Status for Queue 0 + offset. This field indicates for which priority the total number of packets in the output queue has crossed the corresponding threshold value. Each time a priority threshold is crossed for an output queue, the Status field for that particular output queue is updated. This only occurs when the threshold is exceeded. Coding is as follows: 000 queue not full for any priorities 001 Priority 3 full 010 Priorities 2 and 3 full 011 Priorities 1 to 3 full 100 Priorities 0 to 3 full others Reserved
4	Queue Empty for Queue 1 + offset. When set to '1', the corresponding output queue is empty.
5:7	Status for Queue 1 + offset. See description of bits 1:3.

Bit(s)	Description
8	Queue Empty for Queue 2 + offset. When set to '1', the corresponding output queue is empty.
9:11	Status for Queue 2 + offset. See description of bits 1:3.
12	Queue Empty for Queue 3 + offset. When set to '1', the corresponding output queue is empty.
13:15	Status for Queue 3 + offset. See description of bits 1:3.
16	Queue Empty for Queue 4 + offset. When set to '1', the corresponding output queue is empty.
17:19	Status for Queue 4 + offset. See description of bits 1:3.
20	Queue Empty for Queue 5 + offset. When set to '1', the corresponding output queue is empty.
21:23	Status for Queue 5 + offset. See description of bits 1:3.
24	Queue Empty for Queue 6 + offset. When set to '1', the corresponding output queue is empty.
25:27	Status for Queue 6 + offset. See description of bits 1:3.
28	Queue Empty for Queue 7 + offset. When set to '1', the corresponding output queue is empty
29:31	Status for Queue 7 + offset. See description of bits 1:3.

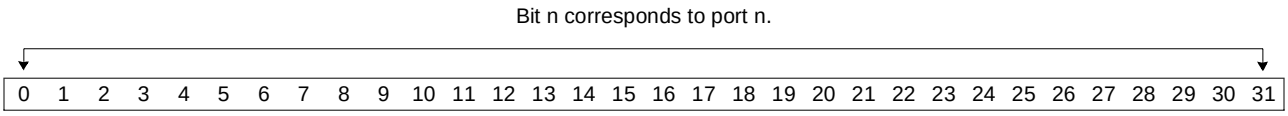
5.6.15 Color Packet Received Register

Specifies if at least one cell has been received with the color selected by the 'ColorSelect' field of the Command register. The register is cleared on read, and bits are set dominant.

Reset Value undefined

Address x'31'

Access Type Read Only



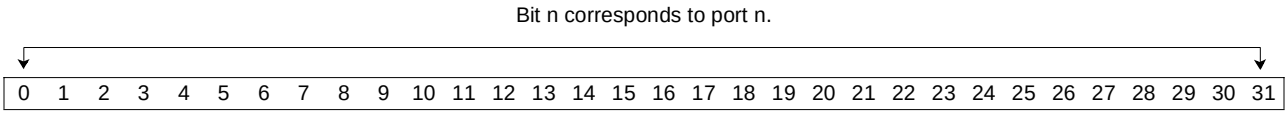
5.6.16 Send Grant Violation Register

Indicates which ports have removed the send grant for too long (see send grant antistreaming function in *5.6.1 Configuration 0 Register* on page 78). The register is cleared on read, and bits are set dominant. This function is only activated if the Send Grant is enabled.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'32'

Access Type Read / Only

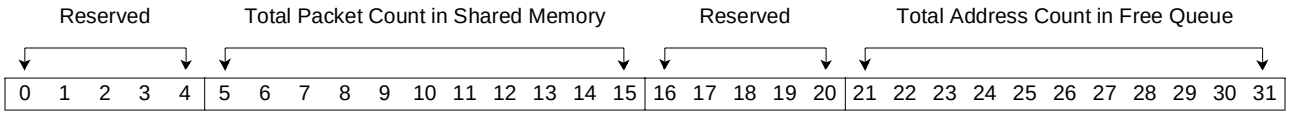


5.6.17 Occupancy Counters Register

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'33'

Access Type Read Only



Bit(s)	Description
0:4	Reserved.
5:15	Total Packet Count in Shared Memory. Indicates the number of packets that are currently in the shared memory. This field is continuously refreshed (each Network Processor internal clock cycle).
16:20	Reserved.
21:31	Total Address Count in Free Queue. Indicates the current number of available addresses. This field in continuously refreshed each internal clock cycle.

5.6.18 Shared Memory Access Registers

The internal shared memory can be accessed via an internal five words (a word is 32 bits) register called the Internal Memory Row Register (This is an internal register accessible via shared memory data register). This register, together with the *5.6.19 Shared Memory Pointer Register* on page 90 and the *5.6.20 Shared Memory Data Register* on page 91, allows to Write / Read any location in the shared memory.

Writing a row to the shared memory:

1. Build the row to be loaded into the shared memory by written five times into the *5.6.20 Shared Memory Data Register* on page 91
   a. First write: loads bytes 0 to 3 of the Internal Memory Row Register (can't find this register).
   b. Second write: loads bytes 4 to 7 of the Internal Memory Row Register.
   c. Third write: loads bytes 8 to11of the Internal Memory Row Register.
   d. Fourth write: loads12 to15 of the Internal Memory Row Register.
   e. Fifth write: loads bytes16 to19 of the Internal Memory Row Register.
2. Write the *5.6.19 Shared Memory Pointer Register* on page 90 with Write bit = 1, and bits 20 to 31 defining the row address. The Internal Memory Row Register content is loaded into the shared memory.

Reading a row from the shared memory:

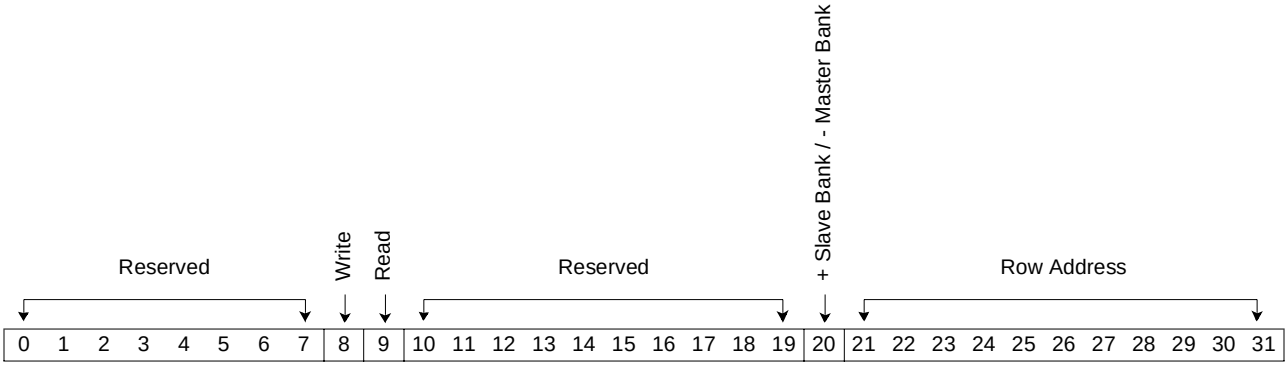
a. Write the *5.6.19 Shared Memory Pointer Register* on page 90 with Read bit = 1, and bits 20 to 31 defining the row address.
b. Read five times the *5.6.20 Shared Memory Data Register* on page 91.
c. First read: returns bytes 0 to 3 of the Internal Memory Row Register.
d. Second read: returns bytes 4 to 7 of the Internal Memory Row Register.
e. Third read: returns bytes 8 to 11 of the Internal Memory Row Register.
f. Fourth read: returns bytes 12 to 15 of the Internal Memory Row Register.
g. Fifth read: returns bytes 16 to 19 of the Internal Memory Row Register

5.6.19 Shared Memory Pointer Register

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’34’

Access Type Read / Write



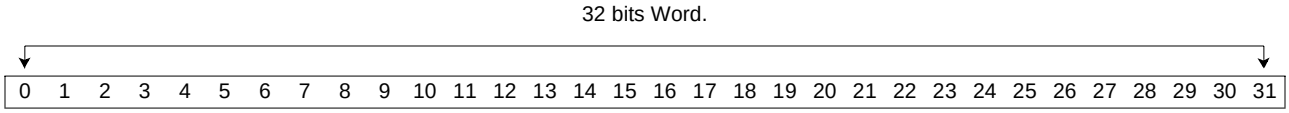
Bit(s)	Description
0:7	Reserved
8	Write. Allow to write data into the shared memory row register
9	Read. Allow to read data from the shared memory row register
10:19	Reserved.
20	+ Slave Bank /- Master Bank. Select the shared memory bank
21:31	Row Address. Select one of the 2048 row address

5.6.20 Shared Memory Data Register

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'35'

Access Type Read / Write



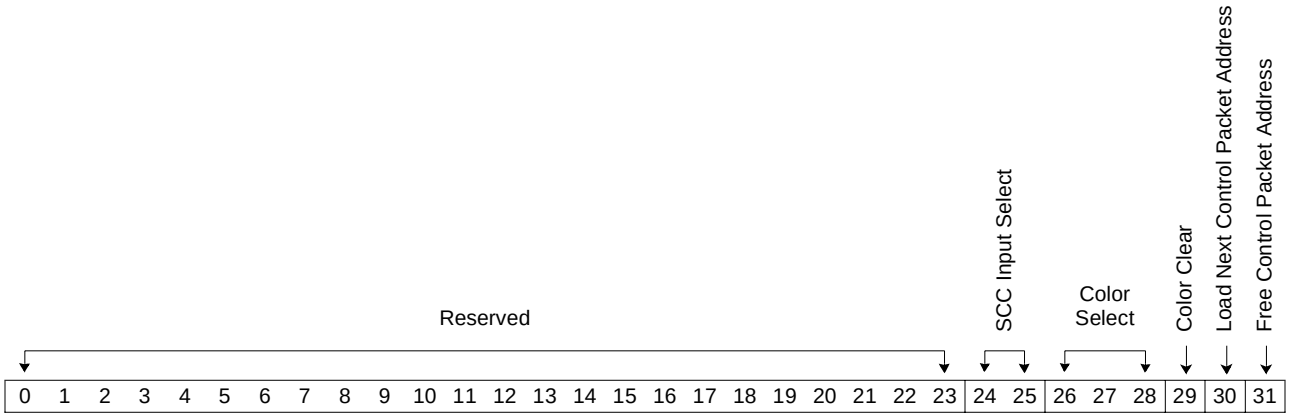
5.6.21 Command Register

This register allows the software to initiate specific actions. Bits 29 to 31 are interpreted as pulses, that is a command is only executed when the register is written to. These registers do not have to be cleared after the command has been executed.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'36'

Access Type Read / Write



Bit(s)	Description
0:23	Reserved
24:25	SCC Input Select. These 2 bits are used to select the incoming Side Communication Channel bit, 1 among 4, that must be reported in the 'Side Communication Channel Input Reporting' register. 00 SCC input bit 0 is reported 01 SCC input bit 1 is reported 10 SCC input bit 2 is reported 11 SCC input bit 3 is reported

Bit(s)	Description
26:28	Color Select. These 3 bits select the color that must be reported in the “Color Packet Received” Register. 000 Blue 001 Red 010 Yellow 1 011 Yellow 2 100 Yellow 3 101 Synchro cell 110 Reserved 111 Reserved
29	Color Clear. This command clears the Idle Packet Color State machine. After this command is initiated, Idle Packets are transmitted with the color specified in the Expected Color field of the *5.6.1 Configuration 0 Register* on page 78. Transmission on a given output port, only takes place when both following conditions are satisfied: at least one packet of the expected color is received on all inputs, and the corresponding output queue is empty. As long as these two conditions are not met, Idle Packets are transmitted with the opposite color to the expected color.
30	Load Next Control Packet Address. Loads the address at the top of the Control packet queue into the Internal Memory Row Register. Note that the read address stays at the top of the Control packet queue.
31	Free Control Packet Address. Frees the address at the top of the Control packet queue. The next address (if available) moves to the top.

5.6.22 Control Packet Destination Register

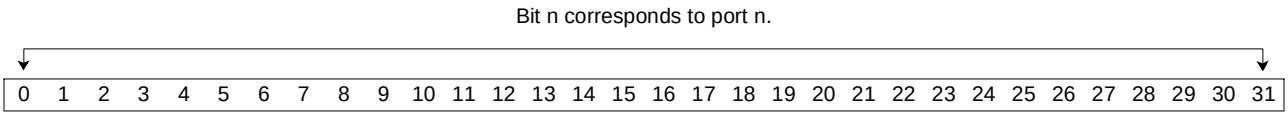
Provides the bit map of the output ports to which the current Control packet is destined. A ‘1’ at position n indicates that the Control packet has to be sent to output n.

When the Control packet is sent over the line, the corresponding bit is cleared in the Control Packet Destination Register. When the Control packet is sent to the last port (register value returns to x’0000 0000’), an All Control Packets Transmitted interrupt is generated. If, for any reason, the Control Packet is not sent to a particular port, the application can reset the corresponding bit into the register.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’37’

Access Type Read / Write



Bit(s)	Description
0:28	Reserved.
29	Color Clear. This command clears the Idle Packet Color State machine. After this command is initiated, Idle Packets are transmitted with the color specified in the Expected Color field of the *5.6.1 Configuration 0 Register* on page 78. Transmission on a given output port, only takes place when both following conditions are satisfied: at least one packet of the expected color is received on all inputs, and the corresponding output queue is empty. As long as these two conditions are not met, Idle Packets are transmitted with the opposite color to the expected color.
30	Load Next Control Packet Address. Loads the address at the top of the Control packet queue into the Internal Memory Row Register. Note that the read address stays at the top of the Control packet queue.

Bit(s)	Description
31	Free Control Packet Address. Frees the address at the top of the Control Packet queue. The next address (if available) moves to the top.

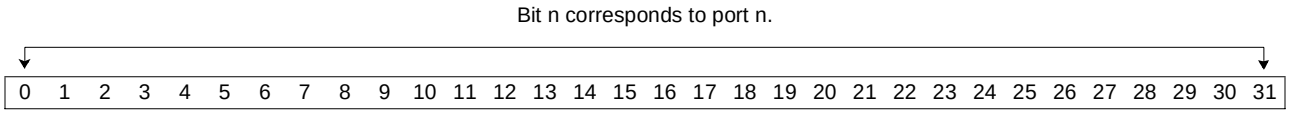
5.6.23 Bitmap Filter Register

Specifies the mask to apply to the received packet bit map for switchover support. See the description of the received packet header for details on the operation of this mask. Note that this register can't be written while the module is in standby mode.

Reset Value '1111 1111 1111 1111 1111 1111 1111 1111'

Address x'38'

Access Type Read / Write



5.6.24 Thresholds Access Register

This register gives indirect access to all the Internal Threshold Registers.

Write access is performed in one SHI command: Set the Write bit to '1', select the Internal Threshold Register with the Threshold Select field, and indicate the value to be loaded with the Threshold Value field.

Read access requires two SHI commands:

- Write the Threshold Access Register with the Write bit cleared to '0' and select the Internal Threshold Register to be read with the Threshold Select field.
- Read the Threshold Value from the Threshold Access Register.

Shared Memory Thresholds: The shared memory thresholds are composed of four 11-bit fields. These values have a range of 0 to 2047. When the number of packets in shared memory becomes equal or greater than the threshold value of a given priority, the corresponding memory grant is low ('0'b). Otherwise, the memory grant bit is high ('1'b).

Threshold Select = 0: Internal Shared Memory Threshold Register for priority 0.
Threshold Select = 1: Internal Shared Memory Threshold Register for priority 1.
Threshold Select = 2: Internal Shared Memory Threshold Register for priority 2.
Threshold Select = 3: Internal Shared Memory Threshold Register for priority 3.

Output Queues Thresholds: The output queues thresholds are composed of four 11-bit fields. These values have a range of 0 to 2047. When the number of packets in a queue becomes equal or greater than the threshold value of a given priority, the corresponding output queue grant is low ('0'b).

Threshold Select = 4: Output Queue Threshold Register for priority 0.
Threshold Select = 5: Output Queue Threshold Register for priority 1.
Threshold Select = 6: Output Queue Threshold Register for priority 2.
Threshold Select = 7: Output Queue Threshold Register for priority 3.

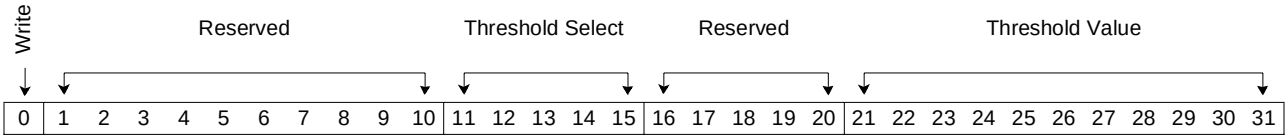
Output Queues Thresholds for Link Paralleling: Same as above but used when a port is defined as a link paralleling one.

Threshold Select = 8: Output Queue Threshold Register for priority 0.
Threshold Select = 9: Output Queue Threshold Register for priority 1.
Threshold Select = 10: Output Queue Threshold Register for priority 2.
Threshold Select = 11: Output Queue Threshold Register for priority 3.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’39’

Access Type Read / Write



Bit(s)	Description
0	Write
1:10	Reserved
11:15	Threshold Select
16:20	Reserved
21:31	Threshold Value

5.6.25 Credit Table Access Register

This register allows indirect access to the credit table. There are 256 credits per port. Therefore, each port has 32 addresses of 16 bits. The One Credit field contains the priority for which a credit must be generated.

Write access is performed in one SHI command: set the Write/Read bit to ‘1’, specify the output port number with the Port Number field, specify the address to be written with the Credit Table Address field, and set the eight credits with the eight One Credit fields.

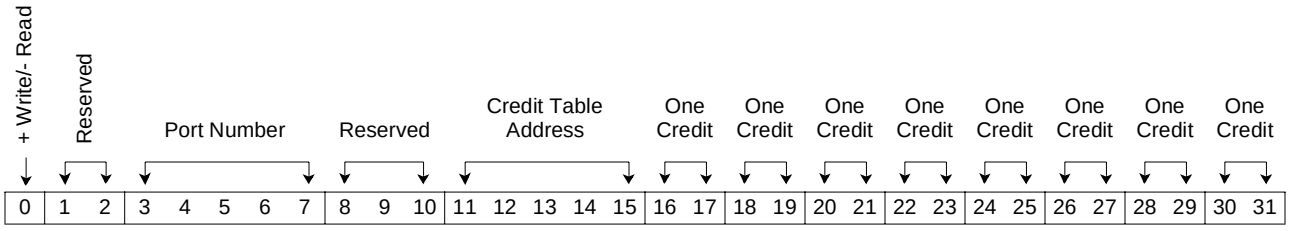
Read access requires two SHI commands:

1. Write this register with the Write / Read bit cleared to ’0’, specify the port number with the Port Number field, and specify the address to be read with Credit Table Address field.
2. Read the eight credit fields from the *5.6.25 Credit Table Access Register* on page 94.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’3A’

Access Type Read / Write



5.6.26 Lookup Tables Access Register

A Lookup Table is a 16-entry table that allows to rearrange the first 16 bytes of each data row of a byte stream before transmission. Thus, there are two such tables, one for the master stream, and one for the slave stream.

For each table, entry at location N of the lookup table points to the data byte to be sent as the N^{th} byte in the same data stream.

The Lookup Table reset values are the normal byte order, from 0 to 15 (no rearranging). Note that the Lookup Table can’t be written while the module is in standby mode. This register allows indirect access to the two Lookup Tables. There are 16 entries per table.

Write access is performed in one SHI command: SHI write to the *5.6.26 Lookup Tables Access Register* on page 95, bit 0 =’1’.

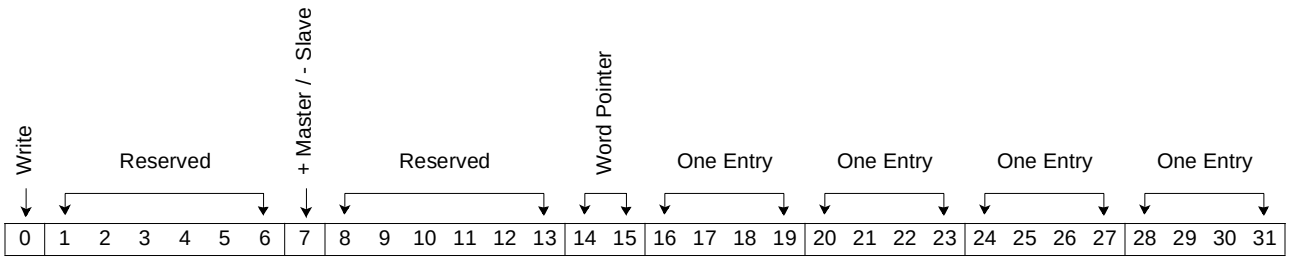
Read Access requires two SHI commands: (should this be here in a footer?)

1. SHI write to the *5.6.26 Lookup Tables Access Register* on page 95, bit 0 =’0’, Master/Slave and Word Pointer fields specify the lookup table address.
2. SHI read to the *5.6.26 Lookup Tables Access Register* on page 95. Four entries are returned.

Reset Value ’0000 0000 0000 0000 0000 0001 0010 0011’

Address x’3B’

Access Type Read / Write



Bit(s)	Description
0	Write. Write access is performed in one SHI command: SHI write to the *5.6.26 Lookup Tables Access Register* on page 95, bit 0 0 0 = ‘1’.
1:6	Reserved.
7	+Master / - Slave. Selects either the master or the slave lookup table
8:13	Reserved.
14:15	Entries Pointer. Selects four entries among 16: 00 0 to 3 01 4 to 7 10 8 to 11 11 12 to 15 The values of the four selected entries are defined from bits 16 to 31.
16:19	One Entry.
20:23	One Entry.
24:27	One Entry.
28:31	One Entry.

5.6.27 Bitmap Mapping Register

This register defines the mapping between a bit position in the Packet header bytes H1 to H4 and a physical queue. All the packets received with a bitmap bit N set to ‘1’ are routed to the physical queue defined by the Physical queue Field (Logical Bitmap Bit Position field).

Example: Assume the *5.6.27 Bitmap Mapping Register* on page 96 is written with the Logical Bitmap Bit Position field = 3, Physical Queue Field = 7. This means that all the packets received with the bitmap bit 3 set to ‘1’ will be routed to Physical output queue 7.

Write access is performed in one SHI command:

1. Set the Write bit to ‘1’, specify both Logical Bitmap Bit Position and Physical Queue fields.

Read access requires two SHI commands:

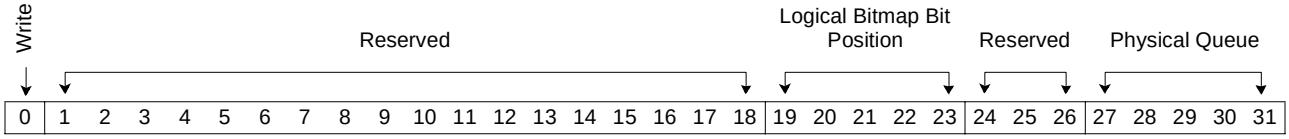
1. Write the *5.6.27 Bitmap Mapping Register* on page 96 with the Write bit cleared to ‘0’ and specify the Logical Bitmap Bit Position field.
2. Read operation will return the corresponding Physical Queue field.

The Bitmap Mapping default value is such as there is no rearranging (Bitmap bit N points to Physical queue N). Note that the *5.6.27 Bitmap Mapping Register* on page 96 can’t be written while the module is in Standby mode.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'3C'

Access Type Read / Write



Bit(s)	Description
0	Write.
1:18	Reserved.
19:23	Logical Bitmap Bit Position.
24:26	Reserved.
27:31	Physical Queue.

5.6.28 Best Effort Resources Access Register

This register gives indirect access to all the Best Effort logic resources.

Write access is performed in one SHI command:

1. Set the Write bit to '1', set both Port Number and Resource Select fields. The Resource Value will be loaded into the selected resource.

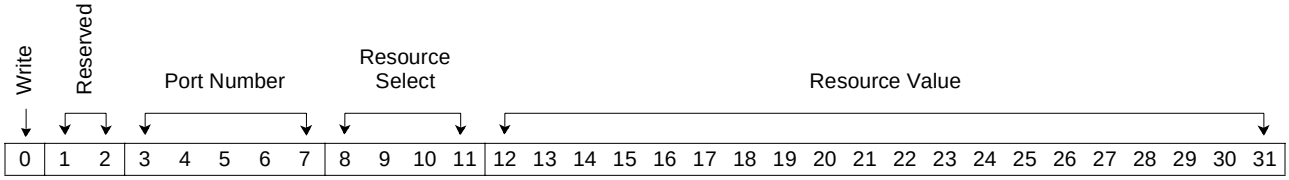
Read access requires two SHI commands:

1. Write the *5.6.28 Best Effort Resources Access Register* on page 97 with the Write bit cleared to '0' and select set Port Number and Resource Select fields.
2. Read the Resource Value from the *5.6.28 Best Effort Resources Access Register* on page 97.

Reset Value '0000 0000 0000 0000 0000 0000 0000 0000'

Address x'3D'

Access Type Read / Write



Bit(s)	Description
0	Write.

Bit(s)	Description
1:2	Reserved.
3:7	Port Number.
8:11	Resource Select. Resource Select encoding: 0000 Best Effort counter 1 0001 Best Effort counter 2 0010 Best Effort counter 3 0011 Best Effort counter 4 0100 Best Effort counter 5 0101 Best Effort Threshold 1 0110 Best Effort Threshold 2 0111 Best Effort Threshold 3 1000 Priority 0 Best Effort Drop counter 1001 Priority 1 Best Effort Drop counter 1010 Priority 2 Best Effort Drop counter 1011 Priority 3 Best Effort Drop counter 1100 Protocol engine virtual packet clock (only bits 22 to 31 are valid in that case). 1101 Reserved 1110 Reserved 1111 Reserved
12:31	Resource value. When the Resource Select field is a counter, it indicates the counter value. When the Resource Select field is a protocol engine virtual clock, the encoded value is: x'00000' No protocol engine packet clock. (protocol engine packet clock frequency = 0) x'00001' 'Single Shot Protocol Engine Packet Clock'. This generates a pulse on the internal protocol engine packet clock (used for test purpose) x'nnnnn' The protocol engine packet clock has a “nnnnn” x 4 ns period.

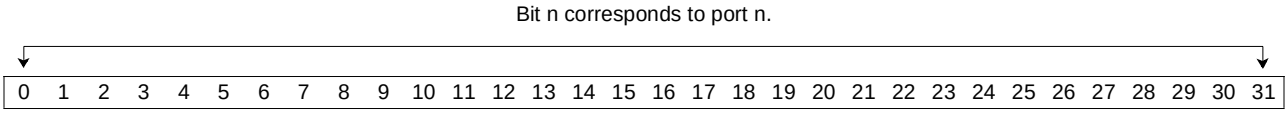
5.6.29 Best Effort Discard Alarm Register

Indicates that the best effort discard logic is currently in the phase where best effort packet could be discarded.

Reset Value ’0000 0000 0000 0000 0000 0000 0000 0000’

Address x’3E’

Access Type Read Only



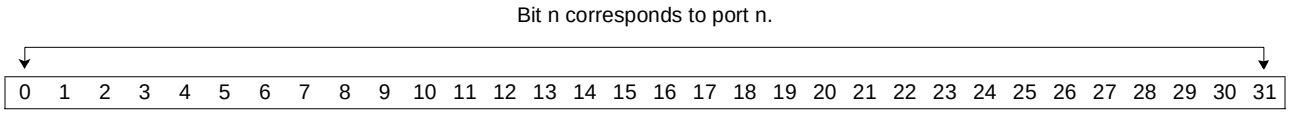
5.6.30 Side Communication Channel Input Reporting Register

Indicates the level of the incoming SCC information extracted from the Idle cells bit n (n can be 0 to 3 depending on the SCC Input Select field of the Command register)

Reset Value Undefined

Address x'3F'

Access Type Read Only



6. Reset, Initialization, and Operation

6.1 Clock and PLL

A reference clock must be provided on the SYS_CLK input. The frequency of the reference clock is 50% of the byte clock, based on the setting of the *5.3.1 PLL Programming Register* on page 58.

6.2 Reset

The NotPowerOnResetIn input must be asserted after a system power-up sequence occurs. For the full reset sequence see *5.3.2.3 Reset Sequence* on page 60 and for the reset register see *5.3.2 Reset Register* on page 59

6.3 DASL Initialization and Operation

Once the switch has been fully configured and before actual data traffic can take place between the switch and the adapter(s), the DASL interfaces between the switch and the remote device connected to specific switch ports must be initialized to provide bit phase alignment and packet alignment at the data receivers in both directions.

The port synchronization is under the overall control of the system Control Processor which coordinates the operation between the switch core and the adapters. For switch elements, this synchronization is handled by the local processor (running the switch control microcode), connected to the SHI interface, after initialization or after error detection. It can also be performed directly through interface lines between the switch control and the port adapter.

The registers of interest are:

Table 19: DASL Initialization and Operation Registers

Address	Register	Description
x'0B'	*5.4.4 Input Port Enable Register* on page 69	
x'09'	*5.4.2 Output Port Enable Register* on page 68	
x'08'	*5.4.1 DASL Output Drivers Enable Register* on page 68	
x'0C'	*5.4.5 DASL Signal Lost Register* on page 69	
x'0F'	*5.4.8 DASL Synchronization Status Register* on page 70	Reports the status of the input receiver. When the packet routing switch is in external speed expansion mode, the local processor must set the Figure DASL Synchronization Hunt Register and check the Figure DASL Synchronization Status Register on both devices separately. Similarly, it has to inform both the master and slave device to transmit Sync Packets via the Figure Synchronization Packet Transmit Register.

Table 19: DASL Initialization and Operation Registers (Continued)

Address	Register	Description
x‘0E’	*5.4.7 DASL Synchronization Hunt Register* on page 70	Forces the input ports to start the synchronization sequence. When the packet routing switch is in external speed expansion mode, the local processor must set the Figure DASL Synchronization Hunt Register and check the Figure DASL Synchronization Status Register on both devices separately. Similarly, it has to inform both the master and slave device to transmit Sync Packets via the Figure Synchronization Packet Transmit Register.
x’0A’	*5.4.3 Synchronization Packet Transmit Register* on page 68	Specifies on which port the sync packets are to be transmitted in order for the remote device input port to synchronize. While Sync Packets are transmitted on an output port, Data Packets destined to that output port will be discarded at the same rate as if they were actually sent on the line. When not transmitting Sync Packets, the output ports transmit normal traffic packets, Data Packets, or Idle Packets.

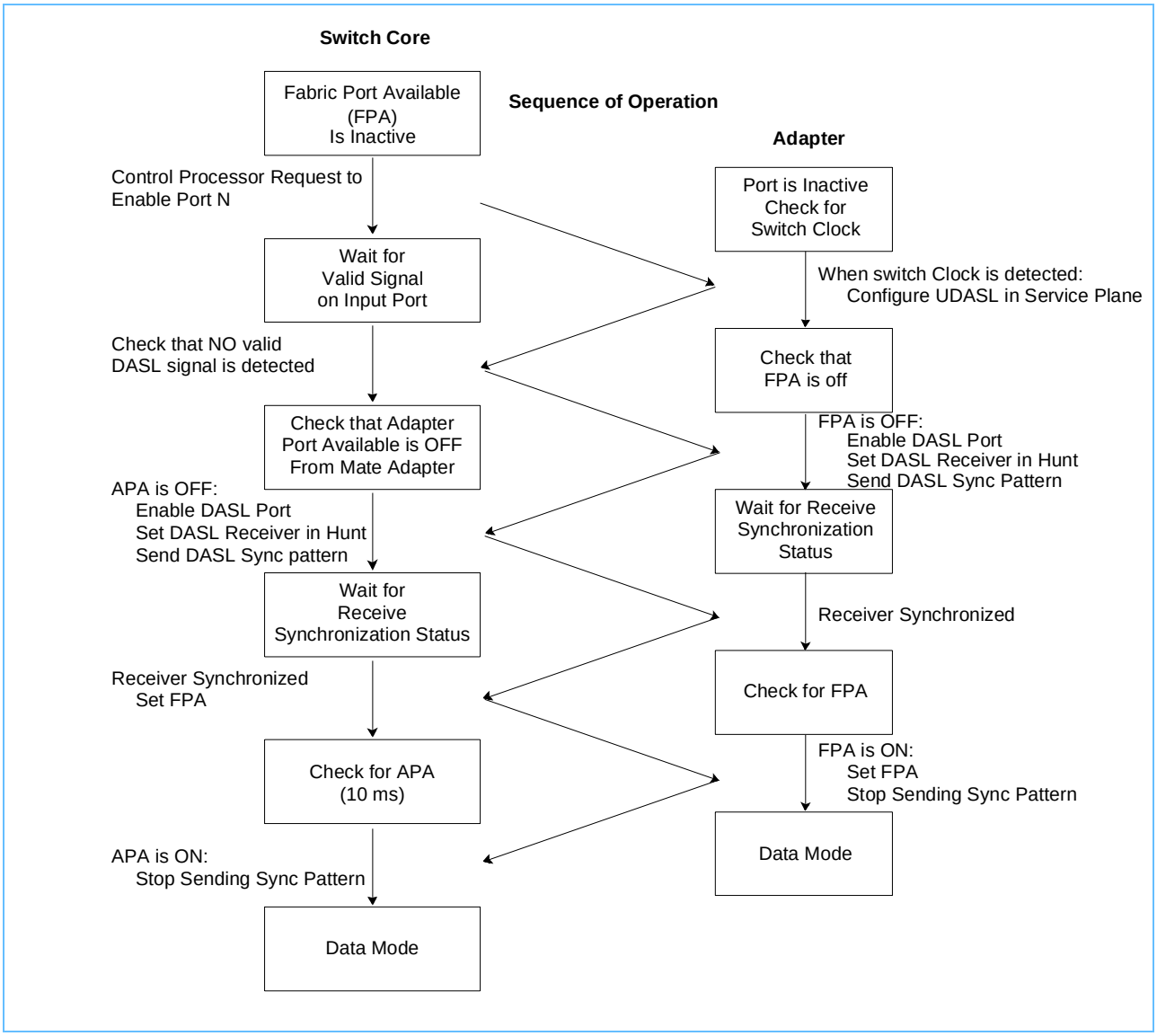
In summary, the following steps must be taken in order to synchronize input ports, either after reset and initialization of the device, or when the control processor decides to resynchronize a link due to data errors on the incoming packets. The same steps must be taken on both the switch port and the remote device. They do not have to be synchronous, but the global sequence of operation must be followed:

1. Disable the switch port, by writing ’0’ in register x‘09’ and x‘0B’ (between 10 to 60 ms to be effective), and remote chip ports
2. Enable the switch port, by writing ’1’ in register x‘09’ and x’0B’ (between 10 to 60 ms to be effective), and remote chip ports
3. Disable DASL transmission, by writing a ’0’ in register x’08’.
4. Check for valid connectivity of the receiver to a differential transmitter through the *5.4.5 DASL Signal Lost Register* on page 69 to ensure the integrity of the serial links.
5. Enable Transmit Sync Packets in the *5.4.3 Synchronization Packet Transmit Register* on page 68 x’0A’.
6. Enable DASL transmission, by writing a ’1’ (it is a bit map field) into registers x’08’ and x‘09’.
7. Enable *5.4.4 Input Port Enable Register* on page 69 x’0B’, by writing a ’1’ in the bit position corresponding to the port.
8. Write to the *5.4.7 DASL Synchronization Hunt Register* on page 70 x’0E’ for the enabled ports to start synchronization.
9. Poll the *5.4.8 DASL Synchronization Status Register* on page 70 x’0F’ to verify completion of synchronization after a Sync Time-Out period. If some ports failed to synchronize, their Sync Status bits will be ’0’.
10. When synchronization has been achieved the local processor and the remote device report to the control processor that it is ready for data transfer.
11. Remove synch hunt from *5.4.7 DASL Synchronization Hunt Register* on page 70 x’0E’.
12. Read *5.6.9 CRC Error Port ID Register* on page 84 x’28’ and x’2A’ to clear any CRC error indication that might have been set up during synchronization period.
13. Upon reception of both reports, the control processor commands both ends of the full duplex link (switch port and remote device) to stop transmitting Sync Packets. Normal packet transfer (idle or data) on the input ports can then be initiated.
14. Activate *5.6.3 Output Queue Enable Register* on page 82 x’22’.

15. Activate *5.6.3 Output Queue Enable Register* on page 82 x'23'.
16. As the link is now in data mode the switch control must poll the CRC Error Registers and the *5.4.5 DASL Signal Lost Register* on page 69 to check for error free operation.

This sequence of operation is shown in the following flow chart. Note that this sequence can be performed on multiple ports at the same time, or on one port at a time.

Figure 14: Switch Fabric DASL Port Synchronization Sequence



7. I/O Definitions and Timing

Note: All signals are active high unless shown with an overbar, which denotes active low.

Table 20: Signal Definitions (Page 1 of 4)

Signal Name	Type I/O Level	I/O Cell Name	Description
Clock and Reset Signals			
OscillatorIn	Input	AIT330S_A	System clock used for the internal clock generation network. OscillatorIn is equal to half the internal byte clock. Its frequency is between 50 to 62.5 MHz with 50% duty cycle.
Osc125MhzOut $\overline{\text{Osc125MhzOut}}$	Output Differential	ODASL	Free running B clock phase of the byte clock at the output of the internal clock tree. This clock output will be at the correct frequency of twice OscillatorIn
Osc250MhzOut $\overline{\text{Osc250MhzOut}}$	Output Differential	ODASL	Free running 250 MHz clock.
$\underline{\text{Osc500MhzOut}}$ $\overline{\text{Osc500MhzOut}}$	Output Differential	ODASL	DASL PLL bit clock.
NotPowerONresetIn	Input	BP2550_A	This signal must be active for at least four SHI clock cycles. Asserting this pin causes a reset of all internal logic, except the IEEE 1149.1 (JTAG logic block). See *6. Reset, Initialization, and Operation* on page 100 for details on the reset sequence.
MSBusPacketClockBidi	Bidirectional	BC2550_C	Low frequency clock used to synchronize the internal sequencers of different modules. For external speed expansion, it is a synchronous signal generated by the master device and fed to the slave device. The period of this clock is equal to LU time with 50% duty cycle. The mode of operation of this pin is programmable via the Sequencer Sync Mode bit in Configuration 1 Register. It can be either tri-stated for single device operation, generated by the device, or received by the device.
NotFullyInsertedIn	Input	BP2550_A	Used to set the DASL drivers in High Z until the board housing the device is fully inserted. This pin ensures that both ends of the switch board on which the device is soldered are fully inserted. An external pull-up resistor forces the inactive state.
Data Signals			
DaslData[00:31]In(0:3) $\overline{\text{DaslData[00:31]In(0:3)}}$	Input	IDASL	DaslData[p]In(n) $\overline{\text{DaslData[p]In(n)}}$ form one of the four 500 Mb/s differential signals for input port p. For each port, bits 0 and 1 carry the slave byte stream and bits 2 and 3 carry the master byte stream.
DaslData[00:31]Out(0:3) $\overline{\text{DaslData[00:31]Out(0:3)}}$	Output	ODASL	DaslData[p]Out(n) $\overline{\text{DaslData[p]Out(n)}}$ Output form one of the four 500 Mb/s differential signals for output port p. For each port, bits 0 and 1 carry the slave byte stream and bits 2 and 3 carry the master byte stream.

Table 20: Signal Definitions (Page 2 of 4)

Signal Name	Type I/O Level	I/O Cell Name	Description
MemoryGrantOut[0:3]	Output	BP2550_A	MemoryGrantOut[n] provides the grant status of the shared memory for priority n. The MemoryGrantOut[n] pins are updated every four clock cycles (of 8 to 10 ns). The device pin encoding and corresponding grant definition is as follows: 0 1 2 3: priority level 0 0 0 0: No Grant 1 0 0 0: Priority 0, or Control Packets 1 1 0 0: Priority 0 or 1, or Control Packets 1 1 1 0: Priority 0 .. 2, or Control Packets 1 1 1 1: Priority 0 .. 3, or Control Packets others: Reserved. See *3. Functional Description* on page 25 for further details. When two devices are in external speed expansion, only the MemoryGrantOut[0:3] bus from the master device is used.
SendGrantIn[0:31]	Input	BP2550_B	Grants the output ports the opportunity to transmit packets. When bit n is active, a packet can be transmitted on portn. When inactive, Data Packets are not allowed to be transmitted; only Idle Packets. When a Data packet is to be transmitted, the packet of highest available priority will be transmitted unless otherwise specified by the credit table. When two devices are in external speed expansion, the SendGrantIn[0:31] bus of the slave device is not used and thus not connected.
ReceiveGrantIn[0:31]	Input	BP2550_B	Provides a grant to receive packets for each output. Incoming packets will only be received for output n if the ReceiveGrantIn[0:31] is active high. Packets destined for outputs for which the ReceiveGrantIn[0:31] is deasserted will not be enqueued in those outputs. When two devices are in external speed expansion, the ReceiveGrantIn[0:31] bus of the slave device is not used and thus not connected.
Master-Slave Speed Expansion Signals The following I/Os are only connected when two devices are in external speed expansion. Note that these signals are synchronous to the internal master/slave clocks and point-to-point between the two devices.			
MSBusSynchOut	Output	BC2550_C	Connects directly to the MSBusSynchIn pin of the other device. This signal is a synchronous one clock cycle (of 8 to 10 ns) pulse to frame the information that is multiplexed on the MSBusInputAddrBidi[0:25] and MSBusOutputAddrBidi[0:25] busses. When the device is used alone (without external speed expansion) this signal is tri stated.
MSBusSynchIn	Input	BC2550_C	Connects directly to the MSBusSynchOut of the other device.

Table 20: Signal Definitions (Page 3 of 4)

Signal Name	Type I/O Level	I/O Cell Name	Description
MSBusInputAddrBidi[0:25]	Bidirectional	BC2550_C	Connects the master to the slave. In single device mode, this bus is tri-stated. It provides the address used by the input controllers to store the next packet. MSBusInputAddrBidi(0:10) are the 11-bit wide address value fields, MSBusInputAddrBidi(11) is the valid bit of the address, and MSBusInputAddrBidi(12) is the odd parity bit over MSBusInputAddrBidi(0:11). MSBusInputAddrBidi(0:12) is a multiplexed bus on which the store addresses of the 16 input controllers with the lowest port number are transmitted. Similarly MSBusInputAddrBidi(13:25) is a multiplexed bus on which the store addresses of the 16 input controllers with the highest port number are transmitted. MSBusInputAddrBidi(24) is the valid bit of the address, and MSBusInputAddrBidi(25) is the odd parity bit over MSBusInputAddrBidi(13:24). They are synchronized with MSBusSynchOut. When MSBusSynchOut is high, the address for port 0 and 16 are carried. The address for port N is carried for N clock cycles after MSBusSynch high, The values of the address and the valid bit are used by the master to indicate to the slave ports, according to the table below, whether to ignore the incoming Data Packet, to receive the incoming Data Packet, or to treat the incoming packet as idle. The slave action is as follows: Bits 0:10 (Address) Bit 1 (Valid Bit) Slave Action Zero value1 Ignore the Data Packet Non Zero value1 Receive the Data Packet - 0 Treat incoming packet as an Idle Packet
MSBusOutputAddrBidi[0:25]	Bidirectional	BC2550_C	Connects the master to the slave. In single device mode, this bus is tri-stated. It provides the address used by the output controllers to read the next packet. MSBusOutputAddrBidi(0:10 and 13:23) are the two times 11-bit wide address value fields, MSBusOutputAddrBidi(11 and 24) is the valid bit of the address, and MSBusOutputAddrBidi(12 and 25) is the odd parity bit over MSBusOutputAddrBidi(0:11 and 13:24). MSBusOutputAddrBidi(0:12 and 13:25) is a multiplexed bus on which the read addresses of all 32 output controllers are transmitted. It is synchronized with MSBusSynchOut. When MSBusSynchOut is high, the address for port 0 and 16 is carried. The address for port N is carried N clock cycles after MSBusSynchOut high.
SHI Interface Signals			
SHIClockIn	Input	BP2550_A	Free running clock line that generates the SHI clock.
NotInterruptOut	Output	BP2550_A	Used to generate an interrupt to the microprocessor. The signal remains asserted until an SHI_READ_STATUS command occurs. To support a wired-OR configuration, NotInterruptOut uses an open-drain driver and is in the high-impedance state when inactive.
SHISerialDataIn	Input	BP2550_A	Serial data line that shifts into the SHI Instruction Register.
SHISerialDataOut	Output	BP2550_A	Serial data line that shifts out of the SHI Instruction Register. The SHISerialDataOut is placed in high-impedance state when the SHI is not in shift state. The SHI is in shift state one SHI clock cycle after SHINotSelectIn = '0'.
SHINotSelectIn	Input	BP2550_A	Enables the SHI operation. One SHI clock cycle after the SHINotSelectIn signal becomes active, the instruction is serially shifted into the SHI Instruction Register.

Table 20: Signal Definitions (Page 4 of 4)

Signal Name	Type I/O Level	I/O Cell Name	Description
Debug Bus Signals			
DebugDataOut[0:15]	Output	BC2550_C	This 16-bit bus provides direct I/O access (logic analyzer) to the Debug Bus specified by the *5.3.7 Debug Bus Select Register* on page 64
Digital Inputs			
SCCIn[0:3]	Input	BP2550_B	Allow access to some information from the switch board

Table 21: Test Signals

Signal Name	I/O Cell Name	Comment
LSSD_SCAN_IN(0:20)	IC25PDT_A	Scan chains inputs
LSSD_SCAN_IN(21:23)	IC25PUT_A	Scan chains inputs JTAG Test Data Input (TDI) = LSSD_SCAN_IN(21) JTAG Test Mode Select (TMS) = LSSD_SCAN_IN(22) JTAG Test Clock (TCK) = LSSD_SCAN_IN(23)
LSSD_SCAN_OUT(0:23)	BP2550T_A	Scan chains outputs Memory BIST DIAG_OUT = LSSD_SCAN_OUT(20) PLL_LOCK = LSSD_SCAN_OUT(21) PLL_TESTOUT = LSSD_SCAN_OUT(22) JTAG Test Data Output (TDO) = LSSD_SCAN_OUT(23)
LSSD_SCAN_MODE	IC25PDT_A	Allows all clocks to be controlled from the primary inputs and connects all scan chains.
LSSD_A_CLK	IC25PUT_A	Used as an external source for the internal SRL A clock.
LSSD_B1_CLK	IC25PUT_A	Used as an external source for the internal SRL B clock.
LSSD_C1_CLK	IC25PUT_A	Used as an external source for the internal SRL C clock.
LSSD_C2_CLK	IC25PDT_A	Used as an external source for the internal SRAM C clock.
LSSD_C3_CLK	IC25PUT_A	Used as an external source for the internal GRA C clock.
LSSD_C4_CLK	IC25PUT_A	Used as an external source for the internal GRA C clock. (4 ports GRA only)
OscillatorIn	AIT330S_A	System clock used for the internal clock generation network.
DI1	IC25D1PUT_A	Serves as the driver inhibit for all non-test outputs. Active low
DI2	IC25D2PUT_A	Serves as the driver inhibit for all test outputs. Active low
RI	IC25RIT_A	Serves as the receiver inhibit for all inputs. Active low External PullUp required
TRST	IC25PDT_A	JTAG Test Reset
IOTEST	IC25PDT_A	Used for Reduced Pin Count Testing. It allows all LSSD boundary inputs to drive signals out LSSD_SCAN_GATE = IOTEST BISTTESTM1 = IOTEST BISTGATE = IOTEST
LSSD_TAP_C1	IC25PUT_A	JTAG Controller C1 clock
LSSD_TAP_C2	IC25PUT_A	JTAG Controller C2 clock

Table 21: Test Signals (Continued)

Signal Name	I/O Cell Name	Comment
BISTTESTM3	IC25PDT_A	Used to handle the internal memory BIST controllers PLL_TESTIN = IOTEST and (not BISTTESTM3)
LeakageTest	IC25LTPUT_A	Used during the leakage test
PLL_VDDA	AID330S_A	PLL Analog voltage
DELAYIn	BP2550PD_A	Input of the internal delay element used for process measurement. The internal delay element is built with a chain of 300 INVERT_O gate
DELAYOut	BP2550_A	Output of the internal delay element used for process measurement. Min=11ns (Process=-3 Sigma, Temperature=0 C, Voltage=2.7v Max=28ns (Process=+3 Sigma, Temperature=125 C, Voltage=2.3v

7.1 I/O Timing

7.1.1 DASL Signals

The following table provides the skew requirements related to the DASL links.

Table 22: DASL Interface Skew

Parameter	Rating	Units	Notes
Maximum skew between the two lines of a differential pair	± 130	ps	
Maximum skew between two 500 Mb/s links of the same port (This applies also for any two ports in speed expansion)	± 2	clock cycles (8 to 10 ns)	1

1. Clock cycle = 8 ns (125 MHz operation) to 10ns (100MHz operation).

7.1.2 SHI Interface Signals

Figure 15: SHI Interface Signal Timing Diagram

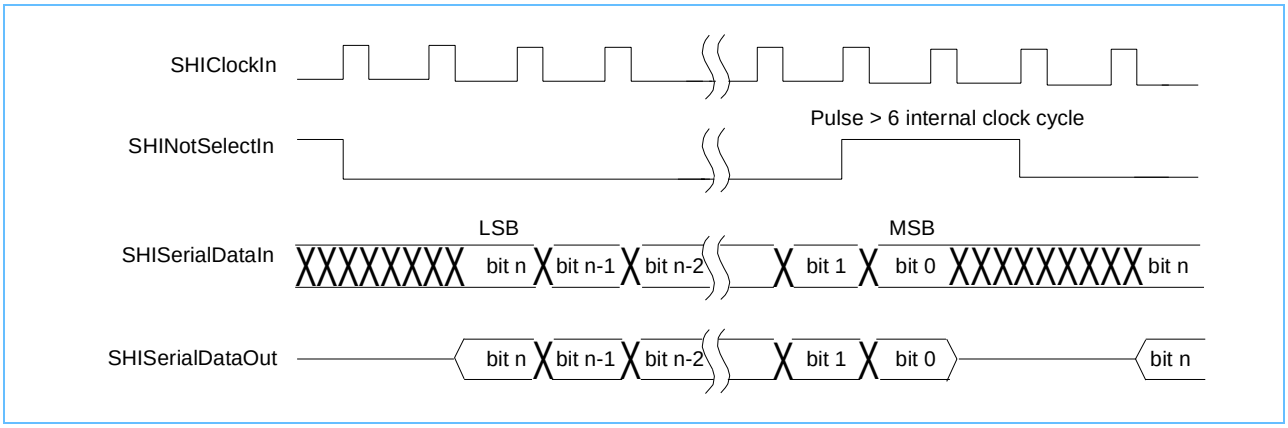


Figure 16: SHI Signal Timing Diagram

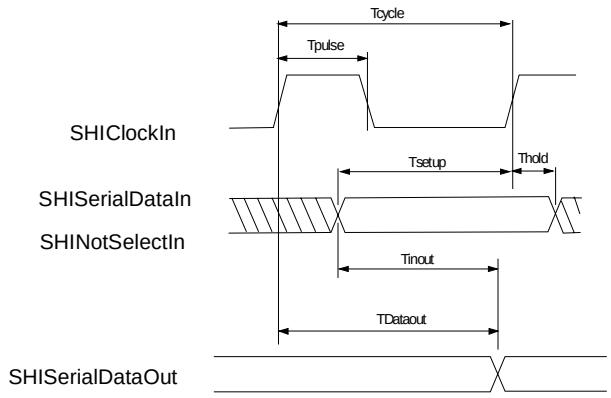


Table 23: SHI Signal Timing Values

Symbol	Parameter	Rating		Units
		Min.	Max.	
T_{CYCLE}	Cycle time	16		ns
T_{PULSE}	Pulse width	4		
T_{SETUP}	Setup time	4		
T_{HOLD}	Hold time	5		
T_{inout}	SHINotSelectIn to SHISerialDataOut	2	6	
$T_{dataout}$	SHIClockIn to SHISerialDataOut	3	8	

7.1.2.1 Timing Window

The sampling window of SND_GRANT(n) is the time window during which the SND_GRANT(n) signal on the device pin is sampled in order to decide on the transmission of a data or Control Packet. This window is defined in relationship with the beginning of transmission of that packet on the device output pins.

Figure 17: SND_GRANT(*n*) Sampling Window in relationship with the packet flow on output n pins.

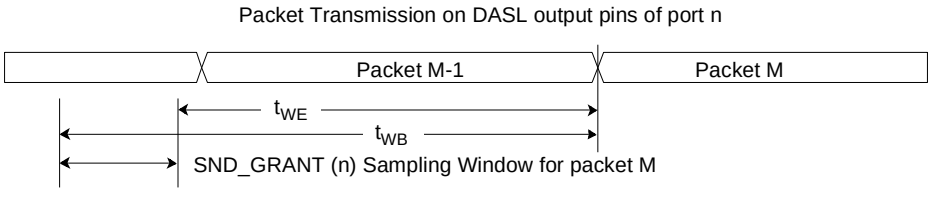


Table 24: SND_GRANT Sampling Window Timing

Symbol	Parameter	Rating	Note
t_{WB}	Sampling Window Beginning (including setup time)	21 ns +18 byte cycles	1
t_{WE}	Sampling Window End (including hold time)	17 ns +18 byte cycles	1

1. Clock cycle = 8 to 10 ns.

7.1.2.2 RCV_GRANT Sampling Window

The sampling window of RCV_GRANT(n) is the time window during which the RCV_GRANT signals on the device pins are sampled in order to decide on the reception of a packet for output n. This window is defined in relationship with the beginning of the packet reception on the device input pins.

Figure 18: RCV_GRANT(*n*) Sampling Window in relationship with the packet flow on any input.

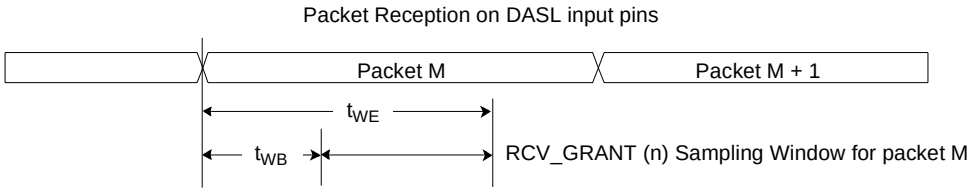


Table 25: RCV_GRANT (i Sampling Window Timing)

Symbol	Parameter	Rating	Note
t_{WB}	Sampling Window Beginning - (including setup time)	-3 ns + 2 clock cycles	1
t_{WE}	Sampling Window End - (including hold time)	14 ns + 5.25 clock cycles	1

1. Clock cycle = 8 to 10 ns.

7.1.3 Master-Slave Speed Expansion Signals

The Master-Slave speed expansion bus (SYS_CLK, MS_SYNC_IN, MS_SYNC_OUT, MS_IN_ADDR, and MS_OUT_ADDR) is synchronous to the internal byte clock of the devices. They are connected point-to-point between the master and the slave devices. In order to guarantee the timing on those signals, the wiring is limited in length.

The Master-Slave speed expansion bus (SEQ_CLK, MS_SYNC_IN, MS_SYNC_OUT, MS_IN_ADDR, and MS_OUT_ADDR) is synchronous to the internal byte clock of the devices.

8. Pin Information

Figure 19: Pinout (42.5 mm CCGA Package Top View Through Ceramic)
For a listing of pin assignments, see *Table 29: I/O Signal List, Sorted by Grid Position* on page 120.

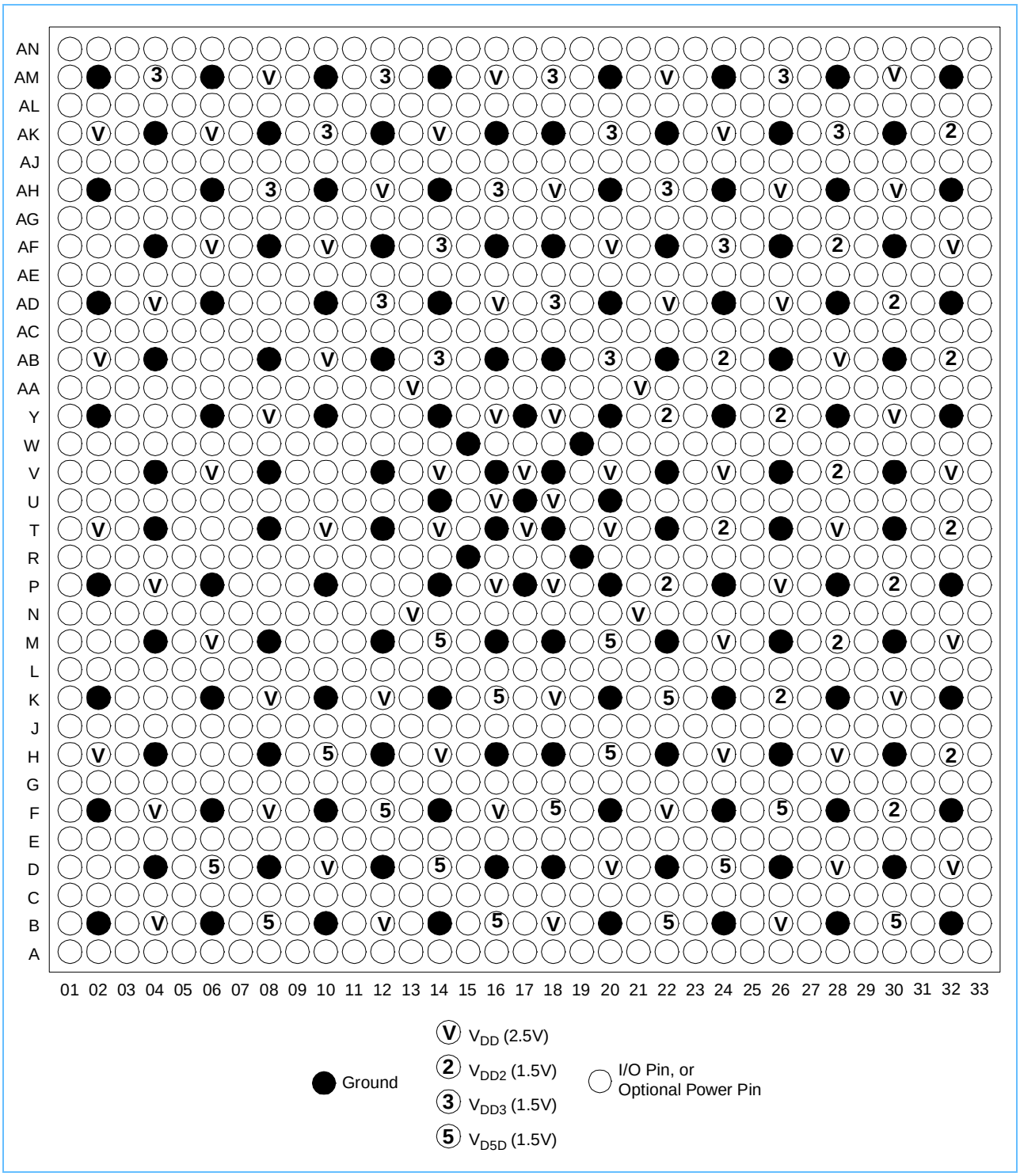


Table 26: Ground and V_{DD} Pin Locations

Pin Function	Pin Grid Locations
GND	AB04, AB08, AB12, AB16, AB18, AB22, AB26, AB30, AD02, AD06, AD10, AD14, AD20, AD24, AD28, AD32, AF04, AF08, AF12, AF16, AF18, AF22, AF26, AF30, AH02, AH06, AH10, AH14, AH20, AH24, AH28, AH32, AK04, AK08, AK12, AK16, AK18, AK22, AK26, AK30, AM02, AM06, AM10, AM14, AM20, AM24, AM28, AM32, B02, B06, B10, B14, B20, B24, B28, B32, D04, D08, D12, D16, D18, D22, D26, D30, F02, F06, F10, F14, F20, F24, F28, F32, H04, H08, H12, H16, H18, H22, H26, H30, K02, K06, K10, K14, K20, K24, K28, K32, M04, M08, M12, M16, M18, M22, M26, M30, P02, P06, P10, P14, P17, P20, P24, P28, P32, R15, R19, T04, T08, T12, T16, T18, T22, T26, T30, U14, U17, U20, V04, V08, V12, V16, V18, V22, V26, V30, W15, W19, Y02, Y06, Y10, Y14, Y17, Y20, Y24, Y28, Y32
V_{DD}	AA13, AA21, AB02, AB10, AB28, AD04, AD16, AD22, AD26, AF06, AF10, AF20, AF32, AH12, AH18, AH26, AH30, AK02, AK06, AK14, AK24, AM08, AM16, AM22, AM30, B04, B12, B18, B26, D10, D20, D28, D32, F04, F08, F16, F22, H02, H14, H24, H28, K08, K12, K18, K30, M06, M24, M32, N13, N21, P04, P16, P18, P26, T02, T10, T14, T17, T20, T28, U16, U18, V06, V14, V17, V20, V24, V32, Y08, Y16, Y18, Y30,
V_{DD2}	AB24, AB32, AD30, AF28, AK32, F30, H32, K26, M28, P22, P30, T24, T32, V28, Y22, Y26
V_{DD3}	AB14, AB20, AD12, AD18, AF14, AF24, AH08, AH16, AH22, AK10, AK20, AK28, AM04, AM12, AM18, AM26
V_{DD5}	B08, B16, B22, B30, D06, D14, D24, F12, F18, F26, H10, H20, K16, K22, M14, M20

Table 27: Optional Power Pins (Recommended)

Pin Function	Pin Grid Location
GND_A	AA15, AA16, AA19, AA29, AB21, AB31, AC19, AE13, AE21, AE25, AE27, AN33, G07, K13, K17, L20, M11, M13, M21, M23, N15, N17, N19, N23, P21, P29, R11, R14, R16, R23, T21, U11, U21, V13, W17, W31, Y11, Y13, Y23
VDD150_A	AC13, AC20, AE07, AG27, AL13, C13, G25, L14, L19, P11, P23, R13, R18, U12, W12, W18, Y21
VDD_A	AA22, AB29, AC22, AD17, AN01, G09, J25, J27, L11, L21, N09, N16, R17, R21, R22, R27, U15, U24, U28, W25, W29, Y19, Y27

Table 28: I/O Signal List, Sorted by Signal Name (Page 1 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
BISTTESTM3	AJ01	DaslData01OutNot(2)	AE26	DaslData03OutNot(1)	AG26
DaslData00In(0)	AE33	DaslData01OutNot(3)	AE28	DaslData03OutNot(2)	AG24
DaslData00In(1)	AE31	DaslData02In(0)	AL30	DaslData03OutNot(3)	AJ25
DaslData00In(2)	AF33	DaslData02In(1)	AL32	DaslData04In(0)	AN24
DaslData00In(3)	AE30	DaslData02In(2)	AN30	DaslData04In(1)	AL24
DaslData00InNot(0)	AD33	DaslData02In(3)	AN32	DaslData04In(2)	AN23
DaslData00InNot(1)	AD31	DaslData02InNot(0)	AK29	DaslData04In(3)	AK23
DaslData00InNot(2)	AE32	DaslData02InNot(1)	AM33	DaslData04InNot(0)	AN25
DaslData00InNot(3)	AF31	DaslData02InNot(2)	AM29	DaslData04InNot(1)	AL25
DaslData00Out(0)	AB27	DaslData02InNot(3)	AM31	DaslData04InNot(2)	AM23
DaslData00Out(1)	AC27	DaslData02Out(0)	AG28	DaslData04InNot(3)	AL23
DaslData00Out(2)	AC25	DaslData02Out(1)	AF29	DaslData04Out(0)	AE22
DaslData00Out(3)	AC28	DaslData02Out(2)	AJ27	DaslData04Out(1)	AF23
DaslData00OutNot(0)	AA26	DaslData02Out(3)	AJ28	DaslData04Out(2)	AF21
DaslData00OutNot(1)	AB25	DaslData02OutNot(0)	AH29	DaslData04Out(3)	AJ23
DaslData00OutNot(2)	AC26	DaslData02OutNot(1)	AG29	DaslData04OutNot(0)	AG23
DaslData00OutNot(3)	AC29	DaslData02OutNot(2)	AJ26	DaslData04OutNot(1)	AE23
DaslData01In(0)	AH31	DaslData02OutNot(3)	AH27	DaslData04OutNot(2)	AG22
DaslData01In(1)	AG32	DaslData03In(0)	AN28	DaslData04OutNot(3)	AH23
DaslData01In(2)	AJ30	DaslData03In(1)	AK27	DaslData05In(0)	AM21
DaslData01In(3)	AJ32	DaslData03In(2)	AM25	DaslData05In(1)	AL22
DaslData01InNot(0)	AG30	DaslData03In(3)	AL26	DaslData05In(2)	AN20
DaslData01InNot(1)	AH33	DaslData03InNot(0)	AM27	DaslData05In(3)	AL20
DaslData01InNot(2)	AK31	DaslData03InNot(1)	AL28	DaslData05InNot(0)	AN22
DaslData01InNot(3)	AK33	DaslData03InNot(2)	AN26	DaslData05InNot(1)	AK21
DaslData01Out(0)	AC24	DaslData03InNot(3)	AK25	DaslData05InNot(2)	AN21
DaslData01Out(1)	AD27	DaslData03Out(0)	AE24	DaslData05InNot(3)	AL21
DaslData01Out(2)	AF27	DaslData03Out(1)	AF25	DaslData05Out(0)	AJ22
DaslData01Out(3)	AE29	DaslData03Out(2)	AJ24	DaslData05Out(1)	AJ21
DaslData01OutNot(0)	AD25	DaslData03Out(3)	AH25	DaslData05Out(2)	AE20
DaslData01OutNot(1)	AD29	DaslData03OutNot(0)	AD23	DaslData05Out(3)	AH19

Table 28: I/O Signal List, Sorted by Signal Name (Page 2 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
DaslData05OutNot(0)	AG21	DaslData07Out(3)	AJ15	DaslData09Out(2)	AG11
DaslData05OutNot(1)	AH21	DaslData07OutNot(0)	AE15	DaslData09Out(3)	AE11
DaslData05OutNot(2)	AG20	DaslData07OutNot(1)	AH17	DaslData09OutNot(0)	AF13
DaslData05OutNot(3)	AJ20	DaslData07OutNot(2)	AF15	DaslData09OutNot(1)	AJ11
DaslData06In(0)	AN19	DaslData07OutNot(3)	AJ16	DaslData09OutNot(2)	AE12
DaslData06In(1)	AL19	DaslData08In(0)	AN12	DaslData09OutNot(3)	AF11
DaslData06In(2)	AN18	DaslData08In(1)	AK13	DaslData10In(0)	AM07
DaslData06In(3)	AN17	DaslData08In(2)	AM11	DaslData10In(1)	AL06
DaslData06InNot(0)	AM19	DaslData08In(3)	AL11	DaslData10In(2)	AM05
DaslData06InNot(1)	AK19	DaslData08InNot(0)	AM13	DaslData10In(3)	AK05
DaslData06InNot(2)	AL18	DaslData08InNot(1)	AL12	DaslData10InNot(0)	AN06
DaslData06InNot(3)	AL17	DaslData08InNot(2)	AN11	DaslData10InNot(1)	AK07
DaslData06Out(0)	AF19	DaslData08InNot(3)	AK11	DaslData10InNot(2)	AN04
DaslData06Out(1)	AJ18	DaslData08Out(0)	AG14	DaslData10InNot(3)	AL04
DaslData06Out(2)	AE19	DaslData08Out(1)	AJ14	DaslData10Out(0)	AG10
DaslData06Out(3)	AJ17	DaslData08Out(2)	AG13	DaslData10Out(1)	AJ09
DaslData06OutNot(0)	AG19	DaslData08Out(3)	AH13	DaslData10Out(2)	AD11
DaslData06OutNot(1)	AJ19	DaslData08OutNot(0)	AE14	DaslData10Out(3)	AG08
DaslData06OutNot(2)	AD19	DaslData08OutNot(1)	AH15	DaslData10OutNot(0)	AJ10
DaslData06OutNot(3)	AF17	DaslData08OutNot(2)	AJ12	DaslData10OutNot(1)	AH09
DaslData07In(0)	AK17	DaslData08OutNot(3)	AJ13	DaslData10OutNot(2)	AE10
DaslData07In(1)	AL16	DaslData09In(0)	AN09	DaslData10OutNot(3)	AF09
DaslData07In(2)	AM15	DaslData09In(1)	AL09	DaslData11In(0)	AK03
DaslData07In(3)	AK15	DaslData09In(2)	AN08	DaslData11In(1)	AM03
DaslData07InNot(0)	AM17	DaslData09In(3)	AK09	DaslData11In(2)	AK01
DaslData07InNot(1)	AN16	DaslData09InNot(0)	AN10	DaslData11In(3)	AM01
DaslData07InNot(2)	AN15	DaslData09InNot(1)	AL10	DaslData11InNot(0)	AJ04
DaslData07InNot(3)	AL15	DaslData09InNot(2)	AM09	DaslData11InNot(1)	AN02
DaslData07Out(0)	AD15	DaslData09InNot(3)	AL08	DaslData11InNot(2)	AJ02
DaslData07Out(1)	AG17	DaslData09Out(0)	AG12	DaslData11InNot(3)	AL02
DaslData07Out(2)	AG15	DaslData09Out(1)	AH11	DaslData11Out(0)	AJ08

Table 28: I/O Signal List, Sorted by Signal Name (Page 3 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
DaslData11Out(1)	AH07	DaslData13Out(1)	AC08	DaslData15Out(1)	V05
DaslData11Out(2)	AH05	DaslData13Out(2)	AA08	DaslData15Out(2)	W09
DaslData11Out(3)	AG05	DaslData13Out(3)	AC05	DaslData15Out(3)	U08
DaslData11OutNot(0)	AJ07	DaslData13OutNot(0)	AC07	DaslData15OutNot(0)	W07
DaslData11OutNot(1)	AJ06	DaslData13OutNot(1)	AC09	DaslData15OutNot(1)	W05
DaslData11OutNot(2)	AG06	DaslData13OutNot(2)	AB07	DaslData15OutNot(2)	W10
DaslData11OutNot(3)	AF05	DaslData13OutNot(3)	AC06	DaslData15OutNot(3)	U06
DaslData12In(0)	AH01	DaslData14In(0)	AA02	DaslData16In(0)	U04
DaslData12In(1)	AG04	DaslData14In(1)	AB03	DaslData16In(1)	T03
DaslData12In(2)	AE02	DaslData14In(2)	Y01	DaslData16In(2)	R02
DaslData12In(3)	AF03	DaslData14In(3)	Y03	DaslData16In(3)	R04
DaslData12InNot(0)	AG02	DaslData14InNot(0)	AB01	DaslData16InNot(0)	U02
DaslData12InNot(1)	AH03	DaslData14InNot(1)	AA04	DaslData16InNot(1)	T01
DaslData12InNot(2)	AF01	DaslData14InNot(2)	AA01	DaslData16InNot(2)	R01
DaslData12InNot(3)	AE04	DaslData14InNot(3)	AA03	DaslData16InNot(3)	R03
DaslData12Out(0)	AE08	DaslData14Out(0)	AB05	DaslData16Out(0)	R10
DaslData12Out(1)	AE06	DaslData14Out(1)	AA05	DaslData16Out(1)	U05
DaslData12Out(2)	AD09	DaslData14Out(2)	Y09	DaslData16Out(2)	R07
DaslData12Out(3)	AD05	DaslData14Out(3)	W06	DaslData16Out(3)	R05
DaslData12OutNot(0)	AF07	DaslData14OutNot(0)	AA07	DaslData16OutNot(0)	R09
DaslData12OutNot(1)	AE05	DaslData14OutNot(1)	AA06	DaslData16OutNot(1)	U07
DaslData12OutNot(2)	AC10	DaslData14OutNot(2)	Y07	DaslData16OutNot(2)	R08
DaslData12OutNot(3)	AD07	DaslData14OutNot(3)	Y05	DaslData16OutNot(3)	T05
DaslData13In(0)	AD01	DaslData15In(0)	W01	DaslData17In(0)	N01
DaslData13In(1)	AD03	DaslData15In(1)	W03	DaslData17In(1)	N03
DaslData13In(2)	AC01	DaslData15In(2)	V01	DaslData17In(2)	M01
DaslData13In(3)	AC04	DaslData15In(3)	U01	DaslData17In(3)	N04
DaslData13InNot(0)	AE01	DaslData15InNot(0)	W02	DaslData17InNot(0)	P01
DaslData13InNot(1)	AE03	DaslData15InNot(1)	W04	DaslData17InNot(1)	P03
DaslData13InNot(2)	AC02	DaslData15InNot(2)	V03	DaslData17InNot(2)	N02
DaslData13InNot(3)	AC03	DaslData15InNot(3)	U03	DaslData17InNot(3)	M03
DaslData13Out(0)	AB09	DaslData15Out(0)	W08	DaslData17Out(0)	P07

Table 28: I/O Signal List, Sorted by Signal Name (Page 4 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
DaslData17Out(1)	P05	DaslData19Out(1)	K07	DaslData21Out(1)	H09
DaslData17Out(2)	N07	DaslData19Out(2)	H07	DaslData21Out(2)	E10
DaslData17Out(3)	N06	DaslData19Out(3)	J05	DaslData21Out(3)	F09
DaslData17OutNot(0)	P09	DaslData19OutNot(0)	K09	DaslData21OutNot(0)	K11
DaslData17OutNot(1)	R06	DaslData19OutNot(1)	K05	DaslData21OutNot(1)	G08
DaslData17OutNot(2)	M05	DaslData19OutNot(2)	J08	DaslData21OutNot(2)	G10
DaslData17OutNot(3)	N05	DaslData19OutNot(3)	J06	DaslData21OutNot(3)	E09
DaslData18In(0)	L02	DaslData20In(0)	E02	DaslData22In(0)	B09
DaslData18In(1)	L03	DaslData20In(1)	C02	DaslData22In(1)	C08
DaslData18In(2)	J01	DaslData20In(2)	E04	DaslData22In(2)	A10
DaslData18In(3)	J03	DaslData20In(3)	A02	DaslData22In(3)	C10
DaslData18InNot(0)	L01	DaslData20InNot(0)	D01	DaslData22InNot(0)	A08
DaslData18InNot(1)	L04	DaslData20InNot(1)	B01	DaslData22InNot(1)	D09
DaslData18InNot(2)	K01	DaslData20InNot(2)	D03	DaslData22InNot(2)	A09
DaslData18InNot(3)	K03	DaslData20InNot(3)	B03	DaslData22InNot(3)	C09
DaslData18Out(0)	M07	DaslData20Out(0)	G06	DaslData22Out(0)	J12
DaslData18Out(1)	L06	DaslData20Out(1)	H05	DaslData22Out(1)	H11
DaslData18Out(2)	L07	DaslData20Out(2)	E07	DaslData22Out(2)	H13
DaslData18Out(3)	L09	DaslData20Out(3)	E06	DaslData22Out(3)	E11
DaslData18OutNot(0)	N08	DaslData20OutNot(0)	F05	DaslData22OutNot(0)	G11
DaslData18OutNot(1)	L05	DaslData20OutNot(1)	G05	DaslData22OutNot(1)	J11
DaslData18OutNot(2)	M09	DaslData20OutNot(2)	E08	DaslData22OutNot(2)	G12
DaslData18OutNot(3)	L08	DaslData20OutNot(3)	F07	DaslData22OutNot(3)	F11
DaslData19In(0)	H01	DaslData21In(0)	A04	DaslData23In(0)	A11
DaslData19In(1)	J04	DaslData21In(1)	C04	DaslData23In(1)	D11
DaslData19In(2)	G02	DaslData21In(2)	A06	DaslData23In(2)	B13
DaslData19In(3)	F03	DaslData21In(3)	D07	DaslData23In(3)	C12
DaslData19InNot(0)	J02	DaslData21InNot(0)	B05	DaslData23InNot(0)	B11
DaslData19InNot(1)	H03	DaslData21InNot(1)	D05	DaslData23InNot(1)	C11
DaslData19InNot(2)	F01	DaslData21InNot(2)	B07	DaslData23InNot(2)	A12
DaslData19InNot(3)	G04	DaslData21InNot(3)	C06	DaslData23InNot(3)	D13
DaslData19Out(0)	L10	DaslData21Out(0)	J10	DaslData23Out(0)	E12

Table 28: I/O Signal List, Sorted by Signal Name (Page 5 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
DaslData23Out(1)	E13	DaslData25Out(1)	F17	DaslData27Out(1)	F23
DaslData23Out(2)	J14	DaslData25Out(2)	G19	DaslData27Out(2)	G23
DaslData23Out(3)	F15	DaslData25Out(3)	E19	DaslData27Out(3)	J23
DaslData23OutNot(0)	G13	DaslData25OutNot(0)	J19	DaslData27OutNot(0)	H21
DaslData23OutNot(1)	F13	DaslData25OutNot(1)	H17	DaslData27OutNot(1)	E23
DaslData23OutNot(2)	G14	DaslData25OutNot(2)	H19	DaslData27OutNot(2)	J22
DaslData23OutNot(3)	E14	DaslData25OutNot(3)	E18	DaslData27OutNot(3)	H23
DaslData24In(0)	A15	DaslData26In(0)	A21	DaslData28In(0)	A26
DaslData24In(1)	C15	DaslData26In(1)	C21	DaslData28In(1)	D25
DaslData24In(2)	A16	DaslData26In(2)	A22	DaslData28In(2)	B27
DaslData24In(3)	C17	DaslData26In(3)	D21	DaslData28In(3)	C28
DaslData24InNot(0)	B15	DaslData26InNot(0)	A20	DaslData28InNot(0)	B25
DaslData24InNot(1)	D15	DaslData26InNot(1)	C20	DaslData28InNot(1)	C26
DaslData24InNot(2)	C16	DaslData26InNot(2)	B21	DaslData28InNot(2)	A28
DaslData24InNot(3)	A17	DaslData26InNot(3)	C22	DaslData28InNot(3)	D27
DaslData24Out(0)	H15	DaslData26Out(0)	G20	DaslData28Out(0)	G24
DaslData24Out(1)	E16	DaslData26Out(1)	E20	DaslData28Out(1)	E25
DaslData24Out(2)	J15	DaslData26Out(2)	G21	DaslData28Out(2)	K23
DaslData24Out(3)	G17	DaslData26Out(3)	F21	DaslData28Out(3)	G26
DaslData24OutNot(0)	G15	DaslData26OutNot(0)	J20	DaslData28OutNot(0)	E24
DaslData24OutNot(1)	E15	DaslData26OutNot(1)	F19	DaslData28OutNot(1)	F25
DaslData24OutNot(2)	K15	DaslData26OutNot(2)	E22	DaslData28OutNot(2)	J24
DaslData24OutNot(3)	E17	DaslData26OutNot(3)	E21	DaslData28OutNot(3)	H25
DaslData25In(0)	B17	DaslData27In(0)	B23	DaslData29In(0)	B29
DaslData25In(1)	C18	DaslData27In(1)	C23	DaslData29In(1)	B31
DaslData25In(2)	B19	DaslData27In(2)	A25	DaslData29In(2)	D29
DaslData25In(3)	D19	DaslData27In(3)	C25	DaslData29In(3)	B33
DaslData25InNot(0)	D17	DaslData27InNot(0)	A23	DaslData29InNot(0)	A30
DaslData25InNot(1)	A18	DaslData27InNot(1)	D23	DaslData29InNot(1)	A32
DaslData25InNot(2)	A19	DaslData27InNot(2)	A24	DaslData29InNot(2)	C30
DaslData25InNot(3)	C19	DaslData27InNot(3)	C24	DaslData29InNot(3)	C32
DaslData25Out(0)	K19	DaslData27Out(0)	G22	DaslData29Out(0)	E26

Table 28: I/O Signal List, Sorted by Signal Name (Page 6 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
DaslData29Out(1)	F27	DaslData31Out(1)	L29	LSSD_C2_CLK	G31
DaslData29Out(2)	F29	DaslData31Out(2)	N26	LSSD_C3_CLK	AJ31
DaslData29Out(3)	G29	DaslData31Out(3)	M25	LSSD_C4_CLK	C29
DaslData29OutNot(0)	E27	DaslData31OutNot(0)	L25	LSSD_SCAN_IN(0)	T09
DaslData29OutNot(1)	E28	DaslData31OutNot(1)	L28	LSSD_SCAN_IN(1)	AE16
DaslData29OutNot(2)	G28	DaslData31OutNot(2)	M27	LSSD_SCAN_IN(10)	AN07
DaslData29OutNot(3)	H29	DaslData31OutNot(3)	L27	LSSD_SCAN_IN(11)	E05
DaslData30In(0)	D31	DebugDataOut(0)	L23	LSSD_SCAN_IN(12)	AL33
DaslData30In(1)	D33	DebugDataOut(1)	N22	LSSD_SCAN_IN(13)	C01
DaslData30In(2)	G30	DebugDataOut(10)	L30	LSSD_SCAN_IN(14)	G33
DaslData30In(3)	F33	DebugDataOut(11)	L31	LSSD_SCAN_IN(15)	C05
DaslData30InNot(0)	E30	DebugDataOut(12)	N29	LSSD_SCAN_IN(16)	AE18
DaslData30InNot(1)	E32	DebugDataOut(13)	N28	LSSD_SCAN_IN(17)	AJ29
DaslData30InNot(2)	F31	DebugDataOut(14)	P25	LSSD_SCAN_IN(18)	AL27
DaslData30InNot(3)	G32	DebugDataOut(15)	P27	LSSD_SCAN_IN(19)	AN03
DaslData30Out(0)	J26	DebugDataOut(2)	G27	LSSD_SCAN_IN(2)	AJ33
DaslData30Out(1)	J28	DebugDataOut(3)	N20	LSSD_SCAN_IN(20)	AL03
DaslData30Out(2)	K25	DebugDataOut(4)	N24	LSSD_SCAN_IN(21)	AL01
DaslData30Out(3)	K29	DebugDataOut(5)	N25	LSSD_SCAN_IN(22)	G03
DaslData30OutNot(0)	H27	DebugDataOut(6)	L33	LSSD_SCAN_IN(23)	C07
DaslData30OutNot(1)	J29	DebugDataOut(7)	L32	LSSD_SCAN_IN(3)	AN29
DaslData30OutNot(2)	L24	DebugDataOut(8)	M29	LSSD_SCAN_IN(4)	AN27
DaslData30OutNot(3)	K27	DebugDataOut(9)	N27	LSSD_SCAN_IN(5)	AL05
DaslData31In(0)	J32	DELAYIn	AA09	LSSD_SCAN_IN(6)	V27
DaslData31In(1)	H31	DELAYOut	M17	LSSD_SCAN_IN(7)	AL31
DaslData31In(2)	K33	DI1	AJ05	LSSD_SCAN_IN(8)	V25
DaslData31In(3)	K31	DI2	A03	LSSD_SCAN_IN(9)	AG33
DaslData31InNot(0)	H33	IOTEST	V07	LSSD_SCAN_MODE	A27
DaslData31InNot(1)	J30	LeakageTest	U31	LSSD_SCAN_OUT(0)	G01
DaslData31InNot(2)	J33	LSSD_A_CLK	E31	LSSD_SCAN_OUT(1)	C33
DaslData31InNot(3)	J31	LSSD_B1_CLK	A29	LSSD_SCAN_OUT(10)	G16
DaslData31Out(0)	L26	LSSD_C1_CLK	T27	LSSD_SCAN_OUT(11)	AG31

Table 28: I/O Signal List, Sorted by Signal Name (Page 7 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
LSSD_SCAN_OUT(12)	E03	MSBusInputAddrBidi(14)	W28	MSBusOutputAddrBidi(2)	M31
LSSD_SCAN_OUT(13)	J16	MSBusInputAddrBidi(15)	AA30	MSBusOutputAddrBidi(20)	U33
LSSD_SCAN_OUT(14)	AG01	MSBusInputAddrBidi(16)	W21	MSBusOutputAddrBidi(21)	U27
LSSD_SCAN_OUT(15)	AG03	MSBusInputAddrBidi(17)	AA31	MSBusOutputAddrBidi(22)	U29
LSSD_SCAN_OUT(16)	C27	MSBusInputAddrBidi(18)	Y31	MSBusOutputAddrBidi(23)	U26
LSSD_SCAN_OUT(17)	A07	MSBusInputAddrBidi(19)	Y25	MSBusOutputAddrBidi(24)	U32
LSSD_SCAN_OUT(18)	AL29	MSBusInputAddrBidi(2)	V33	MSBusOutputAddrBidi(25)	U30
LSSD_SCAN_OUT(19)	AL07	MSBusInputAddrBidi(20)	AA28	MSBusOutputAddrBidi(3)	N30
LSSD_SCAN_OUT(2)	C31	MSBusInputAddrBidi(21)	AC31	MSBusOutputAddrBidi(4)	R28
LSSD_SCAN_OUT(20)	AN05	MSBusInputAddrBidi(22)	AC30	MSBusOutputAddrBidi(5)	R25
LSSD_SCAN_OUT(21)	E33	MSBusInputAddrBidi(23)	W22	MSBusOutputAddrBidi(6)	R24
LSSD_SCAN_OUT(22)	V09	MSBusInputAddrBidi(24)	AA23	MSBusOutputAddrBidi(7)	R20
LSSD_SCAN_OUT(23)	J18	MSBusInputAddrBidi(25)	AA27	MSBusOutputAddrBidi(8)	R26
LSSD_SCAN_OUT(3)	A05	MSBusInputAddrBidi(3)	U25	MSBusOutputAddrBidi(9)	R33
LSSD_SCAN_OUT(4)	C03	MSBusInputAddrBidi(4)	V19	MSBusPacketClockBidi	AB33
LSSD_SCAN_OUT(5)	AN31	MSBusInputAddrBidi(5)	V29	MSBusSyncIn	AC32
LSSD_SCAN_OUT(6)	E01	MSBusInputAddrBidi(6)	V23	MSBusSyncOut	AA25
LSSD_SCAN_OUT(7)	E29	MSBusInputAddrBidi(7)	V21	NotFullyInsertedIn	M19
LSSD_SCAN_OUT(8)	G18	MSBusInputAddrBidi(8)	W30	NotInterruptOut	L18
LSSD_SCAN_OUT(9)	A31	MSBusInputAddrBidi(9)	W26	NotPowerOnResetIn	N18
LSSD_TAP_C1	T07	MSBusOutputAddrBidi(0)	P31	Osc125MhzOut	W32
LSSD_TAP_C2	AG16	MSBusOutputAddrBidi(1)	N31	Osc125MhzOutNot	W33
MemoryGrantOut(0)	A33	MSBusOutputAddrBidi(10)	R32	Osc250MhzOut	P33
MemoryGrantOut(1)	P19	MSBusOutputAddrBidi(11)	R31	Osc250MhzOutNot	N33
MemoryGrantOut(2)	K21	MSBusOutputAddrBidi(12)	R30	Osc500MhzOut	N32
MemoryGrantOut(3)	J21	MSBusOutputAddrBidi(13)	T23	Osc500MhzOutNot	M33
MSBusInputAddrBidi(0)	U23	MSBusOutputAddrBidi(14)	T29	OscillatorIn	T25
MSBusInputAddrBidi(1)	V31	MSBusOutputAddrBidi(15)	R29	PLL_VDDA	T31
MSBusInputAddrBidi(10)	W27	MSBusOutputAddrBidi(16)	T19	ReceiveGrantIn(0)	AA24
MSBusInputAddrBidi(11)	W20	MSBusOutputAddrBidi(17)	T33	ReceiveGrantIn(1)	AA20
MSBusInputAddrBidi(12)	W24	MSBusOutputAddrBidi(18)	U19	ReceiveGrantIn(10)	AB17
MSBusInputAddrBidi(13)	Y29	MSBusOutputAddrBidi(19)	U22	ReceiveGrantIn(11)	AA17

Table 28: I/O Signal List, Sorted by Signal Name (Page 8 of 8)

Signal Name	Grid Position	Signal Name	Grid Position	Signal Name	Grid Position
ReceiveGrantIn(12)	AC17	ReceiveGrantIn(6)	AD21	SendGrantIn(22)	T15
ReceiveGrantIn(13)	AE17	ReceiveGrantIn(7)	AB19	SendGrantIn(23)	U09
ReceiveGrantIn(14)	AC16	ReceiveGrantIn(8)	AA18	SendGrantIn(24)	U13
ReceiveGrantIn(15)	W16	ReceiveGrantIn(9)	AC18	SendGrantIn(25)	U10
ReceiveGrantIn(16)	AL14	RI	AJ03	SendGrantIn(26)	V15
ReceiveGrantIn(17)	AN13	SCCIn(0)	W23	SendGrantIn(27)	V11
ReceiveGrantIn(18)	AB15	SCCIn(1)	L22	SendGrantIn(28)	W14
ReceiveGrantIn(19)	AC15	SCCIn(2)	AN14	SendGrantIn(29)	W13
ReceiveGrantIn(2)	AB23	SCCIn(3)	Y15	SendGrantIn(3)	C14
ReceiveGrantIn(20)	AD13	SendGrantIn(0)	L17	SendGrantIn(30)	W11
ReceiveGrantIn(21)	AC14	SendGrantIn(1)	J17	SendGrantIn(31)	AA10
ReceiveGrantIn(22)	AG09	SendGrantIn(10)	P15	SendGrantIn(4)	A13
ReceiveGrantIn(23)	AB13	SendGrantIn(11)	L12	SendGrantIn(5)	A14
ReceiveGrantIn(24)	AC12	SendGrantIn(12)	J09	SendGrantIn(6)	M15
ReceiveGrantIn(25)	AE09	SendGrantIn(13)	N12	SendGrantIn(7)	L15
ReceiveGrantIn(26)	AC11	SendGrantIn(14)	J07	SendGrantIn(8)	J13
ReceiveGrantIn(27)	AB11	SendGrantIn(15)	N14	SendGrantIn(9)	L13
ReceiveGrantIn(28)	AA12	SendGrantIn(16)	N11	SHIClockIn	AC33
ReceiveGrantIn(29)	AG07	SendGrantIn(17)	N10	SHINotSelectIn	AA32
ReceiveGrantIn(3)	AC23	SendGrantIn(18)	R12	SHISerialDataIn	AA33
ReceiveGrantIn(30)	AA14	SendGrantIn(19)	P13	SHISerialDataOut	Y33
ReceiveGrantIn(31)	AA11	SendGrantIn(2)	L16	TRST	AG18
ReceiveGrantIn(4)	AG25	SendGrantIn(20)	T13		
ReceiveGrantIn(5)	AC21	SendGrantIn(21)	T11		

Table 29: I/O Signal List, Sorted by Grid Position (Page 1 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
A02	DaslData20In(3)	A33	MemoryGrantOut(0)	AB09	DaslData13Out(0)
A03	DI2	AA01	DaslData14InNot(2)	AB11	ReceiveGrantIn(27)
A04	DaslData21In(0)	AA02	DaslData14In(0)	AB13	ReceiveGrantIn(23)
A05	LSSD_SCAN_OUT(3)	AA03	DaslData14InNot(3)	AB15	ReceiveGrantIn(18)
A06	DaslData21In(2)	AA04	DaslData14InNot(1)	AB17	ReceiveGrantIn(10)
A07	LSSD_SCAN_OUT(17)	AA05	DaslData14Out(1)	AB19	ReceiveGrantIn(7)
A08	DaslData22InNot(0)	AA06	DaslData14OutNot(1)	AB23	ReceiveGrantIn(2)
A09	DaslData22InNot(2)	AA07	DaslData14OutNot(0)	AB25	DaslData00OutNot(1)
A10	DaslData22In(2)	AA08	DaslData13Out(2)	AB27	DaslData00Out(0)
A11	DaslData23In(0)	AA09	DELAYIn	AB33	MSBusPacketClockBidi
A12	DaslData23InNot(2)	AA10	SendGrantIn(31)	AC01	DaslData13In(2)
A13	SendGrantIn(4)	AA11	ReceiveGrantIn(31)	AC02	DaslData13InNot(2)
A14	SendGrantIn(5)	AA12	ReceiveGrantIn(28)	AC03	DaslData13InNot(3)
A15	DaslData24In(0)	AA14	ReceiveGrantIn(30)	AC04	DaslData13In(3)
A16	DaslData24In(2)	AA17	ReceiveGrantIn(11)	AC05	DaslData13Out(3)
A17	DaslData24InNot(3)	AA18	ReceiveGrantIn(8)	AC06	DaslData13OutNot(3)
A18	DaslData25InNot(1)	AA20	ReceiveGrantIn(1)	AC07	DaslData13OutNot(0)
A19	DaslData25InNot(2)	AA23	MSBusInputAddrBidi(24)	AC09	DaslData13OutNot(1)
A20	DaslData26InNot(0)	AA24	ReceiveGrantIn(0)	AC10	DaslData12OutNot(2)
A21	DaslData26In(0)	AA25	MSBusSyncOut	AC11	ReceiveGrantIn(26)
A22	DaslData26In(2)	AA26	DaslData00OutNot(0)	AC12	ReceiveGrantIn(24)
A23	DaslData27InNot(0)	AA27	MSBusInputAddrBidi(25)	AC14	ReceiveGrantIn(21)
A24	DaslData27InNot(2)	AA28	MSBusInputAddrBidi(20)	AC15	ReceiveGrantIn(19)
A25	DaslData27In(2)	AA30	MSBusInputAddrBidi(15)	AC16	ReceiveGrantIn(14)
A26	DaslData28In(0)	AA31	MSBusInputAddrBidi(17)	AC17	ReceiveGrantIn(12)
A27	LSSD_SCAN_MODE	AA32	SHINotSelectIn	AC18	ReceiveGrantIn(9)
A28	DaslData28InNot(2)	AA33	SHISerialDataIn	AC21	ReceiveGrantIn(5)
A29	LSSD_B1_CLK	AB01	DaslData14InNot(0)	AC23	ReceiveGrantIn(3)
A30	DaslData29InNot(0)	AB03	DaslData14In(1)	AC24	DaslData01Out(0)
A31	LSSD_SCAN_OUT(9)	AB05	DaslData14Out(0)	AC25	DaslData00Out(2)
A32	DaslData29InNot(1)	AB07	DaslData13OutNot(2)	AC26	DaslData00OutNot(2)

Table 29: I/O Signal List, Sorted by Grid Position (Page 2 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
AC27	DaslData00Out(1)	AE10	DaslData10OutNot(2)	AF23	DaslData04Out(1)
AC28	DaslData00Out(3)	AE11	DaslData09Out(3)	AF25	DaslData03Out(1)
AC29	DaslData00OutNot(3)	AE12	DaslData09OutNot(2)	AF27	DaslData01Out(2)
AC30	MSBusInputAddrBidi(22)	AE14	DaslData08OutNot(0)	AF29	DaslData02Out(1)
AC31	MSBusInputAddrBidi(21)	AE15	DaslData07OutNot(0)	AF31	DaslData00InNot(3)
AC32	MSBusSyncIn	AE16	LSSD_SCAN_IN(1)	AF33	DaslData00In(2)
AC33	SHIClockIn	AE17	ReceiveGrantIn(13)	AG01	LSSD_SCAN_OUT(14)
AD01	DaslData13In(0)	AE18	LSSD_SCAN_IN(16)	AG02	DaslData12InNot(0)
AD03	DaslData13In(1)	AE19	DaslData06Out(2)	AG03	LSSD_SCAN_OUT(15)
AD05	DaslData12Out(3)	AE20	DaslData05Out(2)	AG04	DaslData12In(1)
AD07	DaslData12OutNot(3)	AE22	DaslData04Out(0)	AG05	DaslData11Out(3)
AD09	DaslData12Out(2)	AE23	DaslData04OutNot(1)	AG06	DaslData11OutNot(2)
AD11	DaslData10Out(2)	AE24	DaslData03Out(0)	AG07	ReceiveGrantIn(29)
AD13	ReceiveGrantIn(20)	AE26	DaslData01OutNot(2)	AG08	DaslData10Out(3)
AD15	DaslData07Out(0)	AE28	DaslData01OutNot(3)	AG09	ReceiveGrantIn(22)
AD19	DaslData06OutNot(2)	AE29	DaslData01Out(3)	AG10	DaslData10Out(0)
AD21	ReceiveGrantIn(6)	AE30	DaslData00In(3)	AG11	DaslData09Out(2)
AD23	DaslData03OutNot(0)	AE31	DaslData00In(1)	AG12	DaslData09Out(0)
AD25	DaslData01OutNot(0)	AE32	DaslData00InNot(2)	AG13	DaslData08Out(2)
AD27	DaslData01Out(1)	AE33	DaslData00In(0)	AG14	DaslData08Out(0)
AD29	DaslData01OutNot(1)	AF01	DaslData12InNot(2)	AG15	DaslData07Out(2)
AD31	DaslData00InNot(1)	AF03	DaslData12In(3)	AG16	LSSD_TAP_C2
AD33	DaslData00InNot(0)	AF05	DaslData11OutNot(3)	AG17	DaslData07Out(1)
AE01	DaslData13InNot(0)	AF07	DaslData12OutNot(0)	AG18	TRST
AE02	DaslData12In(2)	AF09	DaslData10OutNot(3)	AG19	DaslData06OutNot(0)
AE03	DaslData13InNot(1)	AF11	DaslData09OutNot(3)	AG20	DaslData05OutNot(2)
AE04	DaslData12InNot(3)	AF13	DaslData09OutNot(0)	AG21	DaslData05OutNot(0)
AE05	DaslData12OutNot(1)	AF15	DaslData07OutNot(2)	AG22	DaslData04OutNot(2)
AE06	DaslData12Out(1)	AF17	DaslData06OutNot(3)	AG23	DaslData04OutNot(0)
AE08	DaslData12Out(0)	AF19	DaslData06Out(0)	AG24	DaslData03OutNot(2)
AE09	ReceiveGrantIn(25)	AF21	DaslData04Out(2)	AG25	ReceiveGrantIn(4)

Table 29: I/O Signal List, Sorted by Grid Position (Page 3 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
AG26	DaslData03OutNot(1)	AJ09	DaslData10Out(1)	AK15	DaslData07In(3)
AG28	DaslData02Out(0)	AJ10	DaslData10OutNot(0)	AK17	DaslData07In(0)
AG29	DaslData02OutNot(1)	AJ11	DaslData09OutNot(1)	AK19	DaslData06InNot(1)
AG30	DaslData01InNot(0)	AJ12	DaslData08OutNot(2)	AK21	DaslData05InNot(1)
AG31	LSSD_SCAN_OUT(11)	AJ13	DaslData08OutNot(3)	AK23	DaslData04In(3)
AG32	DaslData01In(1)	AJ14	DaslData08Out(1)	AK25	DaslData03InNot(3)
AG33	LSSD_SCAN_IN(9)	AJ15	DaslData07Out(3)	AK27	DaslData03In(1)
AH01	DaslData12In(0)	AJ16	DaslData07OutNot(3)	AK29	DaslData02InNot(0)
AH03	DaslData12InNot(1)	AJ17	DaslData06Out(3)	AK31	DaslData01InNot(2)
AH05	DaslData11Out(2)	AJ18	DaslData06Out(1)	AK33	DaslData01InNot(3)
AH07	DaslData11Out(1)	AJ19	DaslData06OutNot(1)	AL01	LSSD_SCAN_IN(21)
AH09	DaslData10OutNot(1)	AJ20	DaslData05OutNot(3)	AL02	DaslData11InNot(3)
AH11	DaslData09Out(1)	AJ21	DaslData05Out(1)	AL03	LSSD_SCAN_IN(20)
AH13	DaslData08Out(3)	AJ22	DaslData05Out(0)	AL04	DaslData10InNot(3)
AH15	DaslData08OutNot(1)	AJ23	DaslData04Out(3)	AL05	LSSD_SCAN_IN(5)
AH17	DaslData07OutNot(1)	AJ24	DaslData03Out(2)	AL06	DaslData10In(1)
AH19	DaslData05Out(3)	AJ25	DaslData03OutNot(3)	AL07	LSSD_SCAN_OUT(19)
AH21	DaslData05OutNot(1)	AJ26	DaslData02OutNot(2)	AL08	DaslData09InNot(3)
AH23	DaslData04OutNot(3)	AJ27	DaslData02Out(2)	AL09	DaslData09In(1)
AH25	DaslData03Out(3)	AJ28	DaslData02Out(3)	AL10	DaslData09InNot(1)
AH27	DaslData02OutNot(3)	AJ29	LSSD_SCAN_IN(17)	AL11	DaslData08In(3)
AH29	DaslData02OutNot(0)	AJ30	DaslData01In(2)	AL12	DaslData08InNot(1)
AH31	DaslData01In(0)	AJ31	LSSD_C3_CLK	AL14	ReceiveGrantIn(16)
AH33	DaslData01InNot(1)	AJ32	DaslData01In(3)	AL15	DaslData07InNot(3)
AJ01	BISTTESTM3	AJ33	LSSD_SCAN_IN(2)	AL16	DaslData07In(1)
AJ02	DaslData11InNot(2)	AK01	DaslData11In(2)	AL17	DaslData06InNot(3)
AJ03	RI	AK03	DaslData11In(0)	AL18	DaslData06InNot(2)
AJ04	DaslData11InNot(0)	AK05	DaslData10In(3)	AL19	DaslData06In(1)
AJ05	DI1	AK07	DaslData10InNot(1)	AL20	DaslData05In(3)
AJ06	DaslData11OutNot(1)	AK09	DaslData09In(3)	AL21	DaslData05InNot(3)
AJ07	DaslData11OutNot(0)	AK11	DaslData08InNot(3)	AL22	DaslData05In(1)
AJ08	DaslData11Out(0)	AK13	DaslData08In(1)	AL23	DaslData04InNot(3)

Table 29: I/O Signal List, Sorted by Grid Position (Page 4 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
AL24	DaslData04In(1)	AN07	LSSD_SCAN_IN(10)	B13	DaslData23In(2)
AL25	DaslData04InNot(1)	AN08	DaslData09In(2)	B15	DaslData24InNot(0)
AL26	DaslData03In(3)	AN09	DaslData09In(0)	B17	DaslData25In(0)
AL27	LSSD_SCAN_IN(18)	AN10	DaslData09InNot(0)	B19	DaslData25In(2)
AL28	DaslData03InNot(1)	AN11	DaslData08InNot(2)	B21	DaslData26InNot(2)
AL29	LSSD_SCAN_OUT(18)	AN12	DaslData08In(0)	B23	DaslData27In(0)
AL30	DaslData02In(0)	AN13	ReceiveGrantIn(17)	B25	DaslData28InNot(0)
AL31	LSSD_SCAN_IN(7)	AN14	SCCIn(2)	B27	DaslData28In(2)
AL32	DaslData02In(1)	AN15	DaslData07InNot(2)	B29	DaslData29In(0)
AL33	LSSD_SCAN_IN(12)	AN16	DaslData07InNot(1)	B31	DaslData29In(1)
AM01	DaslData11In(3)	AN17	DaslData06In(3)	B33	DaslData29In(3)
AM03	DaslData11In(1)	AN18	DaslData06In(2)	C01	LSSD_SCAN_IN(13)
AM05	DaslData10In(2)	AN19	DaslData06In(0)	C02	DaslData20In(1)
AM07	DaslData10In(0)	AN20	DaslData05In(2)	C03	LSSD_SCAN_OUT(4)
AM09	DaslData09InNot(2)	AN21	DaslData05InNot(2)	C04	DaslData21In(1)
AM11	DaslData08In(2)	AN22	DaslData05InNot(0)	C05	LSSD_SCAN_IN(15)
AM13	DaslData08InNot(0)	AN23	DaslData04In(2)	C06	DaslData21InNot(3)
AM15	DaslData07In(2)	AN24	DaslData04In(0)	C07	LSSD_SCAN_IN(23)
AM17	DaslData07InNot(0)	AN25	DaslData04InNot(0)	C08	DaslData22In(1)
AM19	DaslData06InNot(0)	AN26	DaslData03InNot(2)	C09	DaslData22InNot(3)
AM21	DaslData05In(0)	AN27	LSSD_SCAN_IN(4)	C10	DaslData22In(3)
AM23	DaslData04InNot(2)	AN28	DaslData03In(0)	C11	DaslData23InNot(1)
AM25	DaslData03In(2)	AN29	LSSD_SCAN_IN(3)	C12	DaslData23In(3)
AM27	DaslData03InNot(0)	AN30	DaslData02In(2)	C14	SendGrantIn(3)
AM29	DaslData02InNot(2)	AN31	LSSD_SCAN_OUT(5)	C15	DaslData24In(1)
AM31	DaslData02InNot(3)	AN32	DaslData02In(3)	C16	DaslData24InNot(2)
AM33	DaslData02InNot(1)	B01	DaslData20InNot(1)	C17	DaslData24In(3)
AN02	DaslData11InNot(1)	B03	DaslData20InNot(3)	C18	DaslData25In(1)
AN03	LSSD_SCAN_IN(19)	B05	DaslData21InNot(0)	C19	DaslData25InNot(3)
AN04	DaslData10InNot(2)	B07	DaslData21InNot(2)	C20	DaslData26InNot(1)
AN05	LSSD_SCAN_OUT(20)	B09	DaslData22In(0)	C21	DaslData26In(1)
AN06	DaslData10InNot(0)	B11	DaslData23InNot(0)	C22	DaslData26InNot(3)

Table 29: I/O Signal List, Sorted by Grid Position (Page 5 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
C23	DaslData27In(1)	E05	LSSD_SCAN_IN(11)	F07	DaslData20OutNot(3)
C24	DaslData27InNot(3)	E06	DaslData20Out(3)	F09	DaslData21Out(3)
C25	DaslData27In(3)	E07	DaslData20Out(2)	F11	DaslData22OutNot(3)
C26	DaslData28InNot(1)	E08	DaslData20OutNot(2)	F13	DaslData23OutNot(1)
C27	LSSD_SCAN_OUT(16)	E09	DaslData21OutNot(3)	F15	DaslData23Out(3)
C28	DaslData28In(3)	E10	DaslData21Out(2)	F17	DaslData25Out(1)
C29	LSSD_C4_CLK	E11	DaslData22Out(3)	F19	DaslData26OutNot(1)
C30	DaslData29InNot(2)	E12	DaslData23Out(0)	F21	DaslData26Out(3)
C31	LSSD_SCAN_OUT(2)	E13	DaslData23Out(1)	F23	DaslData27Out(1)
C32	DaslData29InNot(3)	E14	DaslData23OutNot(3)	F25	DaslData28OutNot(1)
C33	LSSD_SCAN_OUT(1)	E15	DaslData24OutNot(1)	F27	DaslData29Out(1)
D01	DaslData20InNot(0)	E16	DaslData24Out(1)	F29	DaslData29Out(2)
D03	DaslData20InNot(2)	E17	DaslData24OutNot(3)	F31	DaslData30InNot(2)
D05	DaslData21InNot(1)	E18	DaslData25OutNot(3)	F33	DaslData30In(3)
D07	DaslData21In(3)	E19	DaslData25Out(3)	G01	LSSD_SCAN_OUT(0)
D09	DaslData22InNot(1)	E20	DaslData26Out(1)	G02	DaslData19In(2)
D11	DaslData23In(1)	E21	DaslData26OutNot(3)	G03	LSSD_SCAN_IN(22)
D13	DaslData23InNot(3)	E22	DaslData26OutNot(2)	G04	DaslData19InNot(3)
D15	DaslData24InNot(1)	E23	DaslData27OutNot(1)	G05	DaslData20OutNot(1)
D17	DaslData25InNot(0)	E24	DaslData28OutNot(0)	G06	DaslData20Out(0)
D19	DaslData25In(3)	E25	DaslData28Out(1)	G08	DaslData21OutNot(1)
D21	DaslData26In(3)	E26	DaslData29Out(0)	G10	DaslData21OutNot(2)
D23	DaslData27InNot(1)	E27	DaslData29OutNot(0)	G11	DaslData22OutNot(0)
D25	DaslData28In(1)	E28	DaslData29OutNot(1)	G12	DaslData22OutNot(2)
D27	DaslData28InNot(3)	E29	LSSD_SCAN_OUT(7)	G13	DaslData23OutNot(0)
D29	DaslData29In(2)	E30	DaslData30InNot(0)	G14	DaslData23OutNot(2)
D31	DaslData30In(0)	E31	LSSD_A_CLK	G15	DaslData24OutNot(0)
D33	DaslData30In(1)	E32	DaslData30InNot(1)	G16	LSSD_SCAN_OUT(10)
E01	LSSD_SCAN_OUT(6)	E33	LSSD_SCAN_OUT(21)	G17	DaslData24Out(3)
E02	DaslData20In(0)	F01	DaslData19InNot(2)	G18	LSSD_SCAN_OUT(8)
E03	LSSD_SCAN_OUT(12)	F03	DaslData19In(3)	G19	DaslData25Out(2)
E04	DaslData20In(2)	F05	DaslData20OutNot(0)	G20	DaslData26Out(0)

Table 29: I/O Signal List, Sorted by Grid Position (Page 6 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
G21	DaslData26Out(2)	J04	DaslData19In(1)	K09	DaslData19OutNot(0)
G22	DaslData27Out(0)	J05	DaslData19Out(3)	K11	DaslData21OutNot(0)
G23	DaslData27Out(2)	J06	DaslData19OutNot(3)	K15	DaslData24OutNot(2)
G24	DaslData28Out(0)	J07	SendGrantIn(14)	K19	DaslData25Out(0)
G26	DaslData28Out(3)	J08	DaslData19OutNot(2)	K21	MemoryGrantOut(2)
G27	DebugDataOut(2)	J09	SendGrantIn(12)	K23	DaslData28Out(2)
G28	DaslData29OutNot(2)	J10	DaslData21Out(0)	K25	DaslData30Out(2)
G29	DaslData29Out(3)	J11	DaslData22OutNot(1)	K27	DaslData30OutNot(3)
G30	DaslData30In(2)	J12	DaslData22Out(0)	K29	DaslData30Out(3)
G31	LSSD_C2_CLK	J13	SendGrantIn(8)	K31	DaslData31In(3)
G32	DaslData30InNot(3)	J14	DaslData23Out(2)	K33	DaslData31In(2)
G33	LSSD_SCAN_IN(14)	J15	DaslData24Out(2)	L01	DaslData18InNot(0)
H01	DaslData19In(0)	J16	LSSD_SCAN_OUT(13)	L02	DaslData18In(0)
H03	DaslData19InNot(1)	J17	SendGrantIn(1)	L03	DaslData18In(1)
H05	DaslData20Out(1)	J18	LSSD_SCAN_OUT(23)	L04	DaslData18InNot(1)
H07	DaslData19Out(2)	J19	DaslData25OutNot(0)	L05	DaslData18OutNot(1)
H09	DaslData21Out(1)	J20	DaslData26OutNot(0)	L06	DaslData18Out(1)
H11	DaslData22Out(1)	J21	MemoryGrantOut(3)	L07	DaslData18Out(2)
H13	DaslData22Out(2)	J22	DaslData27OutNot(2)	L08	DaslData18OutNot(3)
H15	DaslData24Out(0)	J23	DaslData27Out(3)	L09	DaslData18Out(3)
H17	DaslData25OutNot(1)	J24	DaslData28OutNot(2)	L10	DaslData19Out(0)
H19	DaslData25OutNot(2)	J26	DaslData30Out(0)	L12	SendGrantIn(11)
H21	DaslData27OutNot(0)	J28	DaslData30Out(1)	L13	SendGrantIn(9)
H23	DaslData27OutNot(3)	J29	DaslData30OutNot(1)	L15	SendGrantIn(7)
H25	DaslData28OutNot(3)	J30	DaslData31InNot(1)	L16	SendGrantIn(2)
H27	DaslData30OutNot(0)	J31	DaslData31InNot(3)	L17	SendGrantIn(0)
H29	DaslData29OutNot(3)	J32	DaslData31In(0)	L18	NotInterruptOut
H31	DaslData31In(1)	J33	DaslData31InNot(2)	L22	SCCIn(1)
H33	DaslData31InNot(0)	K01	DaslData18InNot(2)	L23	DebugDataOut (0)
J01	DaslData18In(2)	K03	DaslData18InNot(3)	L24	DaslData30OutNot(2)
J02	DaslData19InNot(0)	K05	DaslData19OutNot(1)	L25	DaslData31OutNot(0)
J03	DaslData18In(3)	K07	DaslData19Out(1)	L26	DaslData31Out(0)

Table 29: I/O Signal List, Sorted by Grid Position (Page 7 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
L27	DaslData31OutNot(3)	N18	NotPowerOnResetIn	R08	DaslData16OutNot(2)
L28	DaslData31OutNot(1)	N20	DebugDataOut(3)	R09	DaslData16OutNot(0)
L29	DaslData31Out(1)	N22	DebugDataOut(1)	R10	DaslData16Out(0)
L30	DebugDataOut(10)	N24	DebugDataOut(4)	R12	SendGrantIn(18)
L31	DebugDataOut(11)	N25	DebugDataOut(5)	R20	MSBusOutputAddrBidi(7)
L32	DebugDataOut(7)	N26	DaslData31Out(2)	R24	MSBusOutputAddrBidi(6)
L33	DebugDataOut(6)	N27	DebugDataOut(9)	R25	MSBusOutputAddrBidi(5)
M01	DaslData17In(2)	N28	DebugDataOut(13)	R26	MSBusOutputAddrBidi(8)
M03	DaslData17InNot(3)	N29	DebugDataOut(12)	R28	MSBusOutputAddrBidi(4)
M05	DaslData17OutNot(2)	N30	MSBusOutputAddrBidi(3)	R29	MSBusOutputAddrBidi(15)
M07	DaslData18Out(0)	N31	MSBusOutputAddrBidi(1)	R30	MSBusOutputAddrBidi(12)
M09	DaslData18OutNot(2)	N32	Osc500MhzOut	R31	MSBusOutputAddrBidi(11)
M15	SendGrantIn(6)	N33	Osc250MhzOutNot	R32	MSBusOutputAddrBidi(10)
M17	DELAYOut	P01	DaslData17InNot(0)	R33	MSBusOutputAddrBidi(9)
M19	NotFullyInsertedIn	P03	DaslData17InNot(1)	T01	DaslData16InNot(1)
M25	DaslData31Out(3)	P05	DaslData17Out(1)	T03	DaslData16In(1)
M27	DaslData31OutNot(2)	P07	DaslData17Out(0)	T05	DaslData16OutNot(3)
M29	DebugDataOut(8)	P09	DaslData17OutNot(0)	T07	LSSD_TAP_C1
M31	MSBusOutputAddrBidi(2)	P13	SendGrantIn(19)	T09	LSSD_SCAN_IN(0)
M33	Osc500MhzOutNot	P15	SendGrantIn(10)	T11	SendGrantIn(21)
N01	DaslData17In(0)	P19	MemoryGrantOut(1)	T13	SendGrantIn(20)
N02	DaslData17InNot(2)	P25	DebugDataOut(14)	T15	SendGrantIn(22)
N03	DaslData17In(1)	P27	DebugDataOut(15)	T19	MSBusOutputAddrBidi(16)
N04	DaslData17In(3)	P31	MSBusOutputAddrBidi(0)	T23	MSBusOutputAddrBidi(13)
N05	DaslData17OutNot(3)	P33	Osc250MhzOut	T25	OscillatorIn
N06	DaslData17Out(3)	R01	DaslData16InNot(2)	T27	LSSD_C1_CLK
N07	DaslData17Out(2)	R02	DaslData16In(2)	T29	MSBusOutputAddrBidi(14)
N08	DaslData18OutNot(0)	R03	DaslData16InNot(3)	T31	PLL_VDDA
N10	SendGrantIn(17)	R04	DaslData16In(3)	T33	MSBusOutputAddrBidi(17)
N11	SendGrantIn(16)	R05	DaslData16Out(3)	U01	DaslData15In(3)
N12	SendGrantIn(13)	R06	DaslData17OutNot(1)	U02	DaslData16InNot(0)
N14	SendGrantIn(15)	R07	DaslData16Out(2)	U03	DaslData15InNot(3)

Table 29: I/O Signal List, Sorted by Grid Position (Page 8 of 8)

Grid Position	Signal Name	Grid Position	Signal Name	Grid Position	Signal Location
U04	DaslData16In(0)	V11	SendGrantIn(27)	W20	MSBusInputAddrBidi(11)
U05	DaslData16Out(1)	V15	SendGrantIn(26)	W21	MSBusInputAddrBidi(16)
U06	DaslData15OutNot(3)	V19	MSBusInputAddrBidi(4)	W22	MSBusInputAddrBidi(23)
U07	DaslData16OutNot(1)	V21	MSBusInputAddrBidi(7)	W23	SCCIn(0)
U08	DaslData15Out(3)	V23	MSBusInputAddrBidi(6)	W24	MSBusInputAddrBidi(12)
U09	SendGrantIn(23)	V25	LSSD_SCAN_IN(8)	W26	MSBusInputAddrBidi(9)
U10	SendGrantIn(25)	V27	LSSD_SCAN_IN(6)	W27	MSBusInputAddrBidi(10)
U13	SendGrantIn(24)	V29	MSBusInputAddrBidi(5)	W28	MSBusInputAddrBidi(14)
U19	MSBusOutputAddrBidi(18)	V31	MSBusInputAddrBidi(1)	W30	MSBusInputAddrBidi(8)
U22	MSBusOutputAddrBidi(19)	V33	MSBusInputAddrBidi(2)	W32	Osc125MhzOut
U23	MSBusInputAddrBidi(0)	W01	DaslData15In(0)	W33	Osc125MhzOutNot
U25	MSBusInputAddrBidi(3)	W02	DaslData15InNot(0)	Y01	DaslData14In(2)
U26	MSBusOutputAddrBidi(23)	W03	DaslData15In(1)	Y03	DaslData14In(3)
U27	MSBusOutputAddrBidi(21)	W04	DaslData15InNot(1)	Y05	DaslData14OutNot(3)
U29	MSBusOutputAddrBidi(22)	W05	DaslData15OutNot(1)	Y07	DaslData14OutNot(2)
U30	MSBusOutputAddrBidi(25)	W06	DaslData14Out(3)	Y09	DaslData14Out(2)
U31	LeakageTest	W07	DaslData15OutNot(0)	Y15	SCCIn(3)
U32	MSBusOutputAddrBidi(24)	W08	DaslData15Out(0)	Y25	MSBusInputAddrBidi(19)
U33	MSBusOutputAddrBidi(20)	W09	DaslData15Out(2)	Y29	MSBusInputAddrBidi(13)
V01	DaslData15In(2)	W10	DaslData15OutNot(2)	Y31	MSBusInputAddrBidi(18)
V03	DaslData15InNot(2)	W11	SendGrantIn(30)	Y33	SHISerialDataOut
V05	DaslData15Out(1)	W13	SendGrantIn(29)		
V07	IOTEST	W14	SendGrantIn(28)		
V09	LSSD_SCAN_OUT(22)	W16	ReceiveGrantIn(15)		

9. Electrical Characteristics

Table 30: Absolute Maximum Ratings

Symbol	Parameter	Min	Typ	Max	Units	Notes
V_{DD}	Supply Voltage	-0.5	2.5	4.3	V	1
V_{IN}	Input Voltage	-0.5		V_{DD}+0.6	V	1, 4
V_{OUT}	Output Voltage	-0.5		V_{DD}	V	1
Θ_{JA}	Thermal impedance junction to ambient package Airflow = 0		10.8		°C/W	2
Θ_{JA}	Thermal impedance junction to ambient package Airflow = 100 FPM		9.8		°C/W	2
Θ_{JA}	Thermal impedance junction to ambient package Airflow = 100 FPM w/heat sink				°C/W	2
Θ_{JC}	Thermal impedance junction to case package		0.20		°C/W	3
T_S	Storage Temperature	-25		150	°C	1
T_A	Operating Junction Temperature Range	0		125	°C	1
	Electrostatic Discharge	-3000	6000	3000	V	1

1. Permanent device damage might occur if the above absolute maximum ratings are exceeded. Functional operation should be restricted to the conditions listed in *Table 31: Recommended Operating Conditions* on page 129. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.
2. Modules mounted to a 2S2P card, 1 oz. copper, size 63.5 mm x 76.2 mm. Flow on both sides of card, vertical orientation.
3. Θ_{JC} represents the difference between junction temperature and the temperature at the top center of the outside surface of the component package divided by the power applied to the component mounted to the test card.
4. V_{DD} max at 2.75 V

Table 31: Recommended Operating Conditions

Symbol	Parameter (for TTL compatible I/Os)		Min	Typ	Max	Units	Notes
V_{DD}	Supply Voltage		2.1	2.5	2.7	V	1, 2, 3
V_{DR}	Supply Voltage		2.1	2.5	2.7	V	1, 2, 3
V_{DD2}	Supply Voltage		1.46	1.5	1.54	V	1, 2, 3
V_{IH}	Input Up Level			1.4	1.7	V	
V_{IL}	Input Down Level		0.7	1.4		V	
V_{OH}	High Level Output Voltage (V_{DD}=min, I_{OH}=-4mA)		2.0	2.3	VDD	V	
V_{OL}	Low Level Output Voltage (V_{DD}=min, I_{OL}=6mA I_{OL}=8mA)		GND	0.2	0.4	V	
I_{IL}	Receiver maximum input leakage Low Level Input Current at LPDL	No pull-up/down			0	µA	4
		With pull-up			0	µA	
		With pull-down			-250	µA	
I_{IH}	Receiver maximum input leakage High Level Input Current at MPUL	No pull-up/down			0	µA	4
		With pull-up			400	µA	
		With pull-down			0	µA	
C_I	Input Capacitance (V_{DD}=nom)				5	pF	

1. For power up, the +2.5 V supply must be activated either prior to or concurrently with the +1.5 V supply. The device might be damaged if the 1.5 V supply is activated for an extended period of time while the 2.5 V supply remains at 0 V level.
2. For power down, the +2.5 V supply must be deactivated either after or concurrently with the +1.5 V supply. The device might be damaged if the 1.5 V supply is activated for an extended period of time while the 2.5 V supply remains at 0 V level.
3. Up to 400 mV of negative voltage can be tolerated during the sequence of any two supplies. There is no time delay requirement, only the negative voltage restriction.
4. Applies to receivers/bidis without pullup or pulldown resistors.

Table 32: Power Dissipation

Supply	Power		Current	
	Nom	Max	Nom	Max
2.5 V	19.8 W	23.2 W	7.9 A	8.6 A
1.5 V	1.7 W	1.8 W	1.1 A	1.2 A

Table 33: Electrical Characteristics for DASL I/Os

Parameter	Ref Signal	Min	Typ	Max	Units	Notes
Rising Transition Rate of the Output	DASL Driver	0.14	0.21	0.38	ns	1
Falling Transition Rate of the Output	DASL Driver	0.19	0.28	0.33	ns	1
Max Input Pin Cap	DASL Receiver		2.5		pF	1

1. The DASL I/Os are compatible with high speed transceiver logic (HSTL) differential interface standard defined in JEDEC document JESD8-6.

Table 34: Clocks

Parameter		Min	Max	Units	Notes
Internal Clock Frequency		100	125	MHz	
OscillatorIn	Internal PLL	50	62.5		1, 2

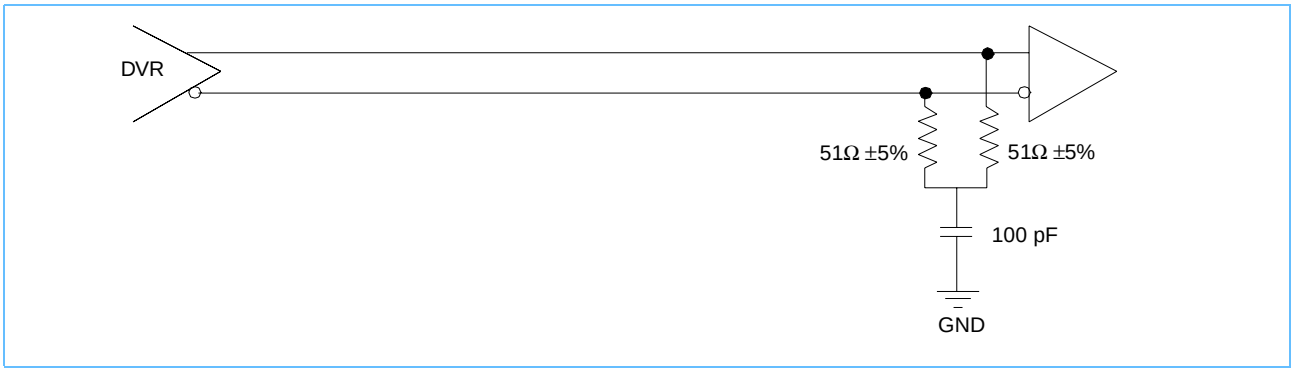
1. The OscillatorIn input must have a tolerance of ±100 ppm (0.01%), a duty cycle of 40 - 60%, and a phase jitter: ±150 ps (cycle to cycle) maximum.
2. The skew between Master and Slave OscillatorIn input pin is ±250 ps maximum.

10. Line Termination

10.1 DASL and SYS_CLK

The DASL data inputs (DATA_IN_*nn*_Q(*i*) and DATA_IN_*nn*_QN(*i*)) and the clock reference (SYS_CLK_Q and SYS_CLK_QN) used differential HSTL EIA/JEDEC (EIA/JESD8-6) standard compliant IO books. The following figure gives the recommended termination for those receiver lines.

Figure 20: DASL Termination

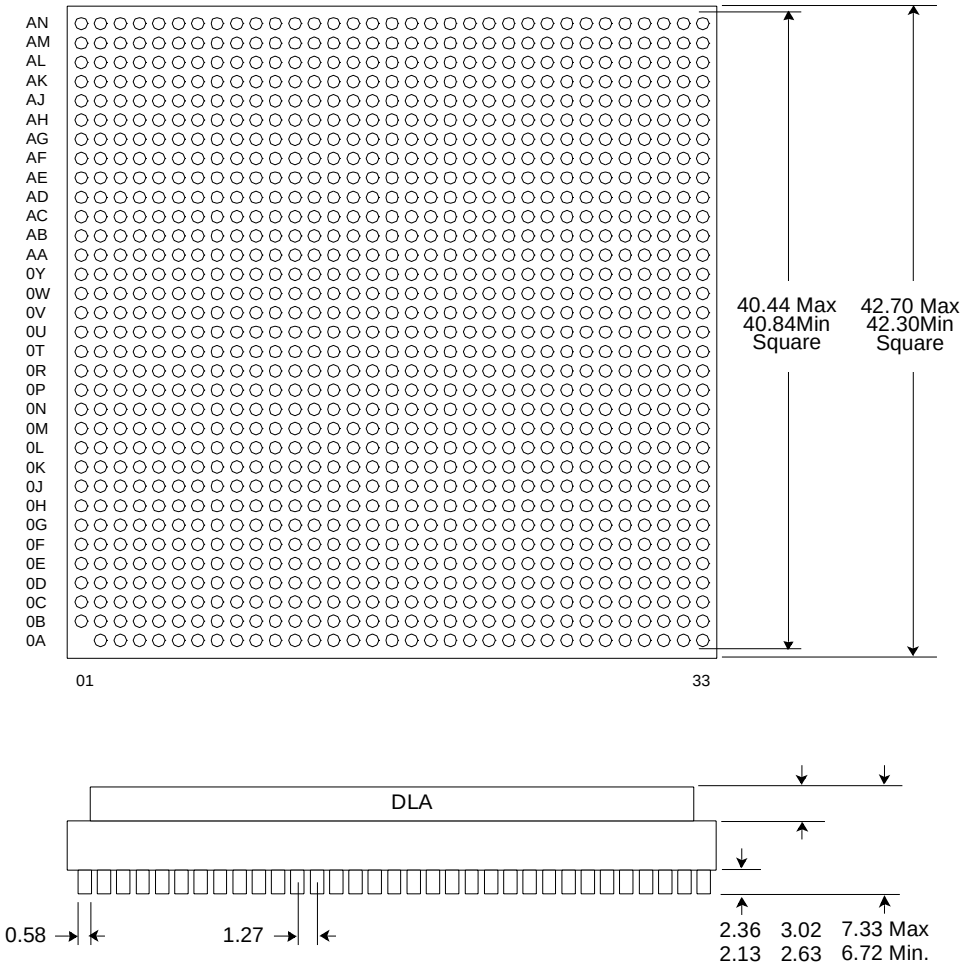


The termination network has to be placed within 2.5 cm of the receiver device.

Note: For the SYS_CLK_Q/QN differential pair, both termination resistors can be directly connected to GND, without the use of a capacitor.

11. Mechanical Information

Figure 21: Physical Dimensions



All measurements are millimeters unless otherwise noted.

Table 35: Package Information

Package Type	Ceramic Column Grid Array (CCGA) with Direct Lid Attach (DLA)
Leads	1088
Power Supply	Dual: V_{DD}2.5 V V_{DR}2.5 V for DASL receiver (treated as VDD at card level; no special requirements) V_{DD2}1.5 V

12. References

1. IEEE 1491.1 Specification, IEEE Standard Test Access Port and Boundary Scan Architecture
2. CMOS5S6 Semi-Custom User's Guide. IBM Document Number PL-SC-C56, October 30, 1998
3. EIA/JEDEC Standard High Speed Transceiver Logic (HSTL) A 1.5 V output Buffer Supply Voltage Based Interface Standard for Digital Integrated Circuits EIA/JESD8-6 August 1995
4. IBM Packet Routing Switch PRS28.4G, Databook, 8/26/99

Revision Log

Revision Date	Contents of Modification
06/01/00	Initial release (revision 00).
08/31/00	First release (01). Changed reference to document type from Databook to Datasheet. Changed page header labels from IBM Packet Routing Switch PRS64G to IBM Packet Routing Switch and from IBM32SW0640DSLCDA250 to PRS64G. Changed references to IBM 64G Packet Routing Switch to IBM Packet Routing Switch PRS64G or PRS64G as appropriate.