IBM Packet Routing Switch Serial Interface Converter

Databook

Advance

All information contained in this document is subject to change without notice. The products described in this document are NOT intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change IBM product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. All information contained in this document was obtained in specific environments, and is presented as an illustration. The results obtained in other operating environments may vary.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED ON AN "AS IS" BASIS. In no event will IBM be liable for damages arising directly or indirectly from any use of the information contained in this document.

While the information contained herein is believed to be accurate, such information is preliminary, and should not be relied upon for accuracy or completeness, and no representations or warranties of accuracy or completeness are made.

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

prssi.01
July 10, 2000

Contents

List of Figures

List of Tables

1. General Information

1.1 Features

- Companion to the IBM 28.4 G Packet Routing Switch
- Support for internal (8 ports) and external (16 ports) switch speed expansion mode
- Proprietary 440 Mbps, 8 HSTL pair, data aligned serial link (DASL) switch interface
- 3.52 Gbps aggregate throughput per speed expanded port
- 32-bit Ingress/Egress Protocol Engine Interface Bus (UTOPIA-3 like bus)
- Implements switch plane redundancy system architecture with two independent paths
- 3 packet ingress/6 packet egress shared buffers
- Implements In Band Flow Control (IBFC) via packet header information
- Up to four priority levels packet handling
- Programmable Packet Length (64 to 80 bytes)
- Link liveness packet insertion (Yellow packet)
- 8-bit Processor Interface (with bursting option)
- Internal ABIST
- IEEE 1149.1 standard boundary scan to facilitate circuit board testing
- 1.5 V DASL differential I/Os for DASLs
- 2.5 V Supply Voltage (3.3 V-tolerant I/Os) and 3.3 V LVTTL for the other signal I/Os
- 360 CBGA package
- IBM CMOS 6 SF SA-12E technology

1.2 Description

The IBM Packet Routing Switch Serial Interface Converter (the converter) is a companion chip to the IBM 28.4 G Packet Routing Switch (the switch), which connects the switch's serial link to a protocol engine (PE) on a 32-bit interface bus. The converter attaches to a switch port operating in speed expansion mode up to 4 Gbps, wired on eight DASL pairs running at up to 500 Mbps per pair.

No synchronization is required between input ports. However, packets on a given port are always received or transmitted at a fixed interval equal to the packet length. The converter ingress/egress packet length is programmable from 64 to 80 bytes in increments of four bytes. Input/Output packets to/from the switch are mapped into 4x logical units (LU) of 16 to 20 bytes depending upon the packet length.

The converter is composed of two fully independent data paths (X and Y) to provide a resilient switch fabric.These paths are clocked, reset and controlled independently to support independent activation/deactivation of each switch plane.

Ingress traffic (packets received from a PE interface bus) is routed to both X and Y path switch planes, thereby duplicating packets on both planes. Egress traffic (packets received from switch plane X or Y) is routed to a bus.

The converter normally uses In Band Flow Control (IBFC). The continuous flow of packets transmits necessary information to regulate packet traffic between the PE and the switch planes (in band signalling). Out of Band Flow Control (OBFC) may also be used with IBFC at the interface level, using the traditional operation UTOPIA-3 handshake signals.

The converter extracts output queue grant information from the 16 output queues of the switch, and Memory Grant information corresponding to the four priority watermark levels of the total shared memory. It then provides this congestion control information to the attached PE through an IBFC mechanism. The information is transmitted to the PE in the packet header bytes of either data or idle packets.

Figure 1: Overall Switch Subsystem Configuration

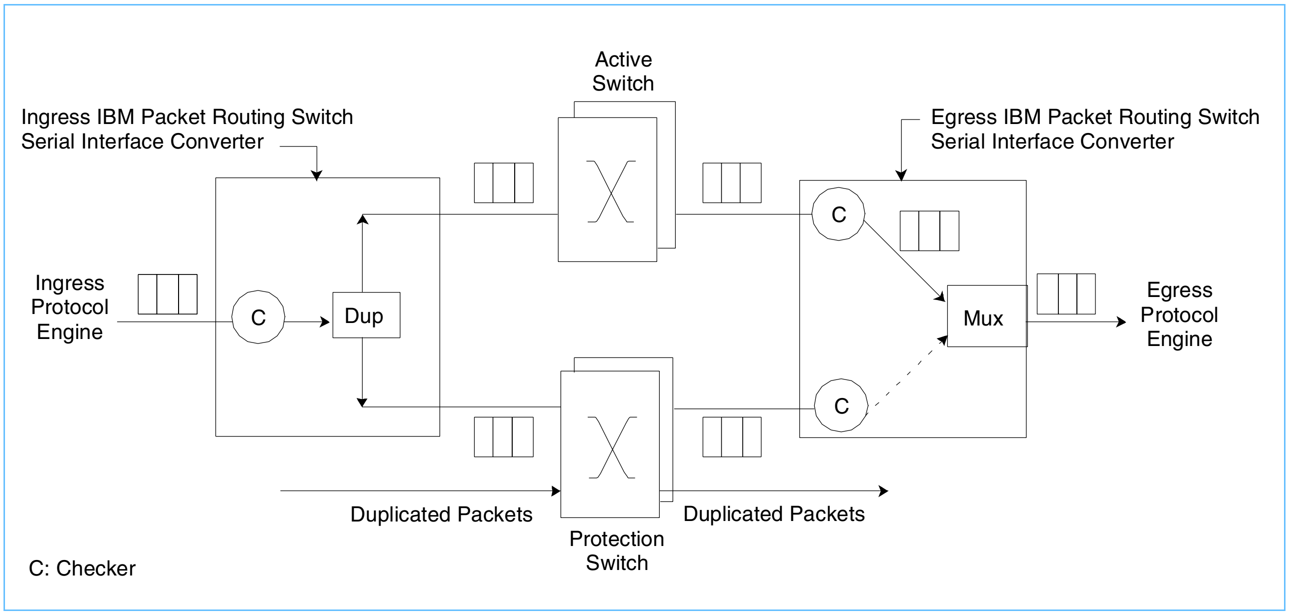
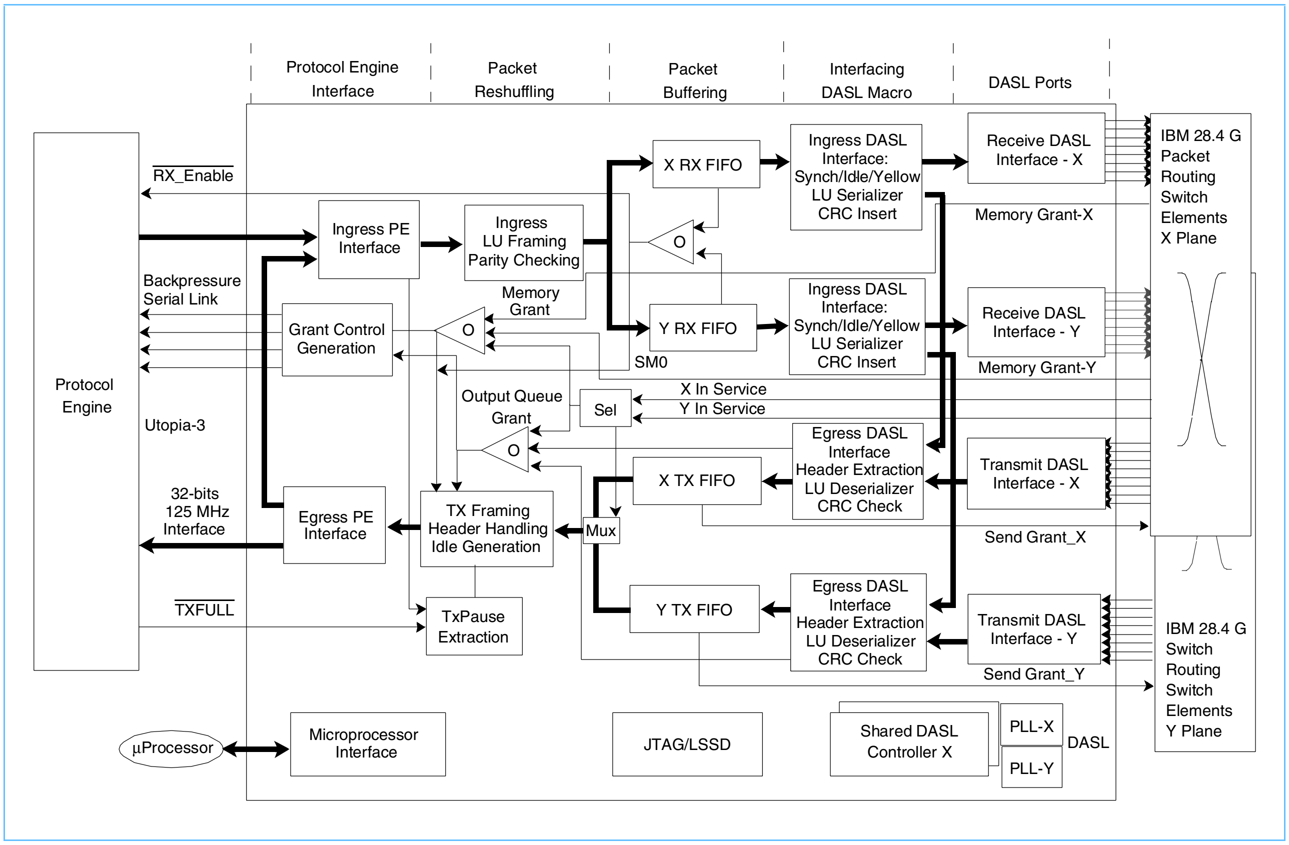


Figure 2: IBM Packet Routing Switch Serial Interface General Data Flow


Protocol Engine Interface
Packet Reshuffling
Packet Buffering
Interfacing DASL Macro
DASL Ports
Protocol Engine
RX_Enable
Backpressure Serial Link
Utopia-3
32-bits 125 MHz Interface
TXFULL
µProcessor
Ingress PE Interface
Grant Control Generation
Egress PE Interface
Ingress LU Framing Parity Checking
Memory Grant
Output Queue Grant
TX Framing Header Handling Idle Generation
TxPause Extraction
Microprocessor Interface
X RX FIFO
Y RX FIFO
O
O
O
SM0
Sel
Mux
X TX FIFO
Y TX FIFO
JTAG/LSSD
Ingress DASL Interface: Synch/Idle/Yellow LU Serializer CRC Insert
Ingress DASL Interface: Synch/Idle/Yellow LU Serializer CRC Insert
Egress DASL Interface Header Extraction LU Deserializer CRC Check
Egress DASL Interface Header Extraction LU Deserializer CRC Check
Shared DASL Controller X
Receive DASL Interface - X
Memory Grant-X
Receive DASL Interface - Y
Memory Grant-Y
X In Service
Y In Service
Transmit DASL Interface - X
Send Grant_X
Transmit DASL Interface - Y
Send Grant_Y
PLL-X
PLL-Y
DASL
IBM 28.4 G Packet Routing Switch Elements X Plane
IBM 28.4 G Switch Routing Switch Elements Y Plane

2. Converter Ingress/Egress Data Flow

The ingress block is the receive path between the protocol engine device (PE) and the rest of the IBM Packet Routing Switch Serial Interface Converter logic. The egress block is the transmit path between the converter transmit logic and the PE. This section provides an overview of functions implemented in the converter ingress and egress data flows. Some functions are duplicated to support two switch planes.

2.1 Ingress Data Flow

2.1.1 Protocol Engine (PE) Ingress Interface

The converter connects to the PE via a 32-bit bus. Ingress data packets are simultaneously routed to PATH X and to PATH Y. Idle packets are inserted in the word stream when there is no data to transfer and are used to maintain a synchronous packet operation in the ingress interface. Idle packets are identified by a bit in the packet qualifier byte. A Receive Start of Packet (RXSOP) synchronized with the data packet is used to delineate packets.

Combined with the RXPRTY signal issued from the PE, the ingress interface checks the parity coherency on each incoming RXDATA [31:0]. A specific bit in the configuration table registers can be set so each parity error issued from the parity checker is reported.

Under the control of the configuration table, RXPRTY_error assertion indicates that the cell that is currently pushed into the ingress reshuffling buffer will be optionally ignored and will not be sent to the ingress FIFO. Cells which are pushed into the ingress FIFOs (X/Y RX FIFOs) are always good and can be treated by the data flow.

2.1.2 Ingress Packet Reshuffling

The ingress logical unit framing block maps incoming data packets into the IBM 28.4 G Packet Routing Switch (switch) logical units (LUs) by moving the five bytes (the packet qualifier and the bit map fields) selected from the configuration register into the master LU, which becomes the switch header information field. The framing block also extracts the IBFC information, discards idle packets, and performs parity checking on the switch header.

2.1.3 Ingress Receive FIFO

The ingress receive FIFOs provide packet synchronization between the 50 - 125 MHz PE interface and the 110 - 125 MHz switch core interface.

2.1.4 Ingress Data Aligned Serial Link Interface (IDI)

The IDI sends packets continuously. Synchronization packets are sent during the DASL synchronization sequence. Data or idle packets are sent once data mode is active. On request, through the configuration table, the LU serializer is filled with a yellow packet and the incoming data packet is buffered while the yellow packet is sent. When there is no data packet to be transmitted to the switch core, the IDI inserts an idle packet, computes the inter idle CRC for each LU, and inserts it as the last byte of each LU.

2.1.5 Data Aligned Serial Link (DASL) Port Serializer

The IDI feeds the DASL port serializer with packets compatible with the switch LU format. (The 16 - 20 byte LU width is set in the configuration register.) The DASL port serializer performs a multi-bit serialization for each LU. Each serial DASL interface line represents a nibble of the LU, so there are two DASL links per byte. The converter provides a total of eight serial links per port (one for each 4-bit nibble) representing a 32-bit wide word.

2.1.6 Egress Data Flow

2.1.7 Data Aligned Serial Link (DASL) Port Deserializer

The DASL egress performs 32-bit deserialization on incoming data and builds LUs for the DASL egress interface. It continuously monitors signal quality on the incoming high speed serial link and performs continuous bit positioning adjustment on the incoming data to maintain synchronization.

2.1.8 Egress Data Aligned Serial Link Interface (EDI)

The DASL's Receive Data Indicator line triggers the LU deserializer logic block which receives a continuous stream of packets. Packet length is programmable from 64 to 80 bytes and is mapped on 4-bytes word boundaries. Therefore, a new packet is received from the DASL every 16 to 20 cycles.

The LU deserializer extracts the switch packet header from the master LU to determine packet type (idle or data) and priority. Idle packets are discarded. Data packets are forwarded (LU format) to the egress buffer interface. The LU deserializer checks the LU CRC after each idle cell, the parity on the switch header, and the type of packet. When an error or yellow packet is detected, and the checker is enabled, the corresponding converter interrupt line is asserted.

2.1.9 Egress FIFO

The egress FIFO interface ignores any idle packets issued from the switch. Only data packets are pushed into the transmit FIFOs (TXFIFOs) for a further word formatting packet operation. The TXFIFOs provide packet synchronization between the 110 - 125 MHz converter and the 50 - 110 MHz protocol engine.

2.1.10 Egress Path Selection

The egress path selection multiplexes packets coming from the X and Y paths to the protocol engine's egress interface bus.The data is selected from the path in service through two interface lines (X_InService and Y_InService) received from the switch core.

2.1.11 Egress Transmit Framing

The packet formatter translates custom formatted packet LUs, reversing the operation performed in the ingress path. Header byte swapping moves the header bytes back to their original position. The converter simultaneously takes the latest available output queue grant flow control data from the EDI and moves it into the corresponding byte positions that were used for output port addressing in the ingress path. The most recent shared memory grant and the output queue grant (OQG) priority (related to the bit map field's OQG) are both stored in the packet qualifier byte that is sent to the PE so it can perform virtual output queuing scheduling for the ingress packets. Idle packets are also generated and sent to the PE in order to maintain continuous flow control information.

2.1.12 Protocol Engine (PE) Egress Interface

The converter connects to the PE’s 32-bit bus via the PE egress interface. Egress transmit framing controls data and idle packet transfer. The PE’s egress interface monitors the behavior of $\overline{\text{TXFULL}}$, controls $\overline{\text{TXENB}}$ accordingly, and generates the Start of Packet to the PE. A parity bit covering the 32-bit word can be generated according to a dedicated bit in the configuration table registers.

3. Functional Description

This chapter describes each functional block of the IBM Packet Routing Switch Serial Interface Converter (the converter). The description is by layer and covers both the ingress and the egress functions because, in general, they are completely symmetrical.

Figure 3: Converter Functional Block Diagram

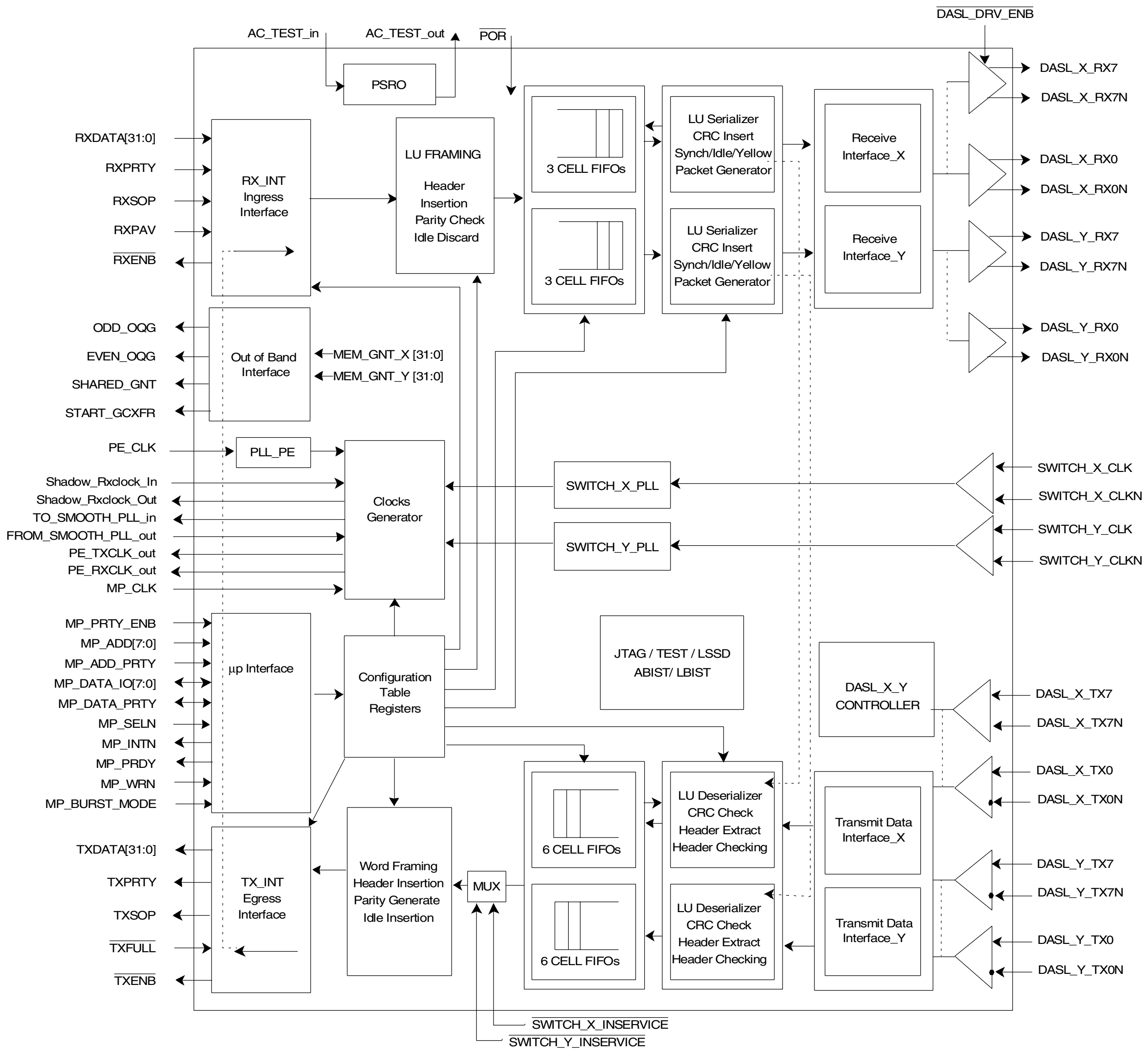
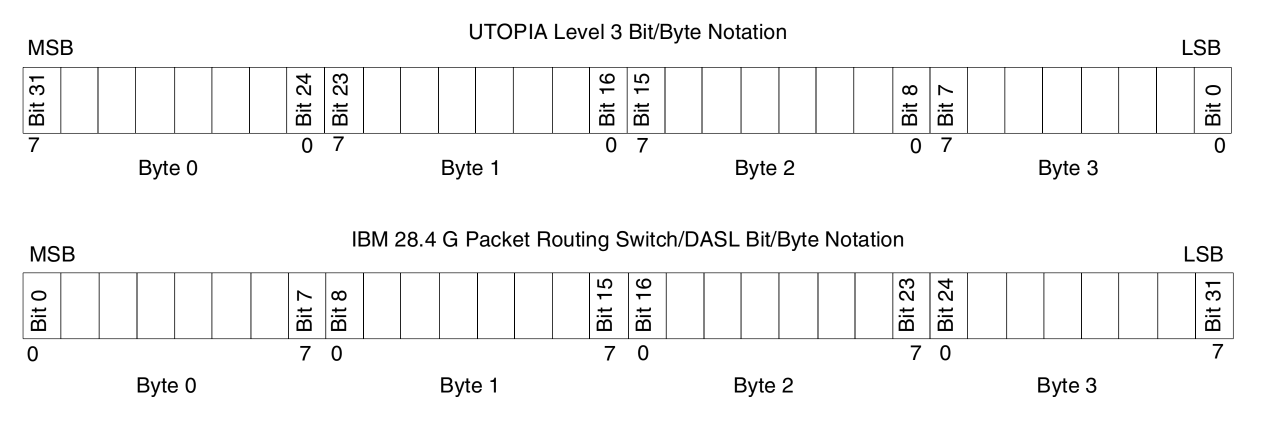


Figure 4: Bit and Byte Notation



The ingress block is the receive path between the protocol engine device (PE) and the rest of the converter logic. The egress block is the transmit path between the converter transmit logic and the PE.

The timing on the UTOPIA-3 interface is the same for out of band flow control (OBFC) and for in band flow control (IBFC) modes except when operating in IBFC mode, the UTOPIA-3 flow control ($\overline{\text{RXENB}}$ at ingress and $\overline{\text{TXFULL}}$ at egress) is not normally used, as the packet header carries all the required information. However if a PE performs an interface level flow control (OBFC), those interface leads can be used for that purpose. In IBFC, fill-up (idle) packets flow through the interface to maintain a continuous stream of flow control information.

3.1 Data Interface Between IBM Packet Routing Switch Serial Interface Converter and Protocol Engine

UTOPIA-3 bit/byte notation is used throughout the protocol engine (PE) interface description. After the byte formatting (which reshuffles the byte position to match the IBM 28.4G Packet Routing Switch (switch) DASL interface), the switch DASL bit/byte notation is used.

3.2 Functional Overview

The IBM Packet Routing Switch Serial Interface Converter (the converter) ingress and egress interfaces are consistent with a subset of the UTOPIA-3 specifications:

- Single PHY interface mode (connection between one converter and one protocol engine)
- Typical operating clock range is 100 - 111 MHz to match the IBM 28.4 G Packet Routing Switch clock rate (up to 111 MHz), but an interface clock rate as low as 50 MHz is supported
- Only 32-bit data paths
- Packet format is 64 - 80 bytes (programmed in the converter at configuration time)
- Only packet level handshake mode (use of signals RXPAV in Receive Path and TXPAV in Transmit Path)

- $\overline{\text{RXENB}}$ and $\overline{\text{TXENB}}$ signals can not be used to perform flow control at an octet level, as is allowed in UTOPIA-2 specifications. Each packet transfer initiated in either the ingress or egress direction will continue to flow until current packet transfer completion, eliminating the ability to insert wait states during the current packet transfer
- The assertion of the $\overline{\text{RXENB}}$ signal depends only on the ingress FIFO filling status and so may be asserted while the RXPAV signal is de-asserted
- Wait states insertion on the bus is only allowed between transmission of two different packets
- All input and output signals are registered. Therefore, a device (either the PE or the converter) responds in not less than two clock cycles after the initiating signal is sent across the interface
- All output signals are generated and all input signals are sampled on low-to-high clock transitions
- All signals are active high, unless the name has an overbar ($\overline{\text{xxx}}$).

3.2.1 Ingress Interface

3.2.1.1 Bus Protocol

The ingress block is the receive path between the protocol engine device (PE) and the IBM Packet Routing Switch Serial Interface Converter (the converter). Data is sent by the PE to the converter according to the following protocol:

- The converter provides the receive clock PE_RXCLK_OUT
- The PE asserts the signal RXPAV when it is ready to send at least one complete packet on the bus
- The converter asserts the signal $\overline{\text{RXENB}}$ when it is ready to receive at least one complete packet
- Receive packet transfer can start once the PE detects $\overline{\text{RXENB}}$ asserted and asserts RXPAV
- The assertion of the signal RXSOP during one clock cycle indicates start of a receive packet transfer
- RXDATA[31:0] is transferred on each low-to-high clock transition and the first data word of the packet is transferred simultaneously with the signal RXSOP
- The converter de-asserts the $\overline{\text{RXENB}}$ signal *two clock cycles before the end of the current packet transfer* to indicate that it can not accept an immediate transfer of the subsequent packet from the PE

This protocol applies if a user wishes to use OBFC mechanisms in addition to IBFC. Under IBFC there is no need for the use of $\overline{\text{RXENB}}$ /RXPAV protocol, as all the flow control is performed in band (through the packet header). Also under IBFC, if there is no data packet to be sent by the protocol engine, it will insert an idle packet that will be discarded by the IBM Packet Routing Switch Serial Interface Converter.

Table 1: Ingress I/O Pin Description

Pin Name	Function
PE_RXCLK_OUT (Output)	Receive Clock (PE_RXCLK_OUT clock) is issued from two clock domains: At POR it is connected to the microprocessor clock until the POR completes. After POR completes, it is generated by either the smooth PLL clock out (derived from SWITCH_PLL X or Y) or by an externally provided clock (for example the 50 - 125 MHz PE clock). PE_RXCLK_OUT source is selected according to the programming of the "PE_RXCLK_OUT_source_I[1:0]" bits in the configuration table registers.
RXDATA[31:0] (Input)	Receive Data is transferred from the PE device to the converter on a 32-bits word basis. LSB MSB RXDATA[0]RXDATA[31]
RXPRTY (Input)	Receive Data Parity Bit is the odd parity bit over the 32 RXDATA bits. The parity_check mode is enabled/disabled by the RXPRTY_enb_l bit in the configuration table registers.
RXPAV (Input, active high)	PE device asserts Receive Packet Available when at least one complete packet is ready to be transmitted on the bus. The signal remains asserted during the current packet transfer (packet level handshake) and indicates, *in the cycle following the last word of the current packet*, if there is (RXPAV asserted) or is not (RXPAV de-asserted) a new packet to transfer. RXPAV must be asserted when operating in IBFC mode.
RXSOP (Input, active high)	PE device asserts Receive Start of Packet for one clock cycle when it starts a packet transfer to indicate the packet's first 32-bits data word.
$\overline{\text{RXENB}}$ (Output, active low)	The converter asserts Receive Enable to indicate its readiness to receive at least one complete packet. The signal remains asserted during the transfer of the current packet (packet level handshake) and, two cycles before the end of the current packet transfer, indicates whether it is (RXENB asserted) or is not (RXENB de-asserted) ready to receive a new complete packet. As this signal is pipelined, the PC devices will respond at least two clock cycles after the RXENB is asserted or de-asserted

3.2.1.2 Ingress Operation and Timing

Figure 5: Ingress Timing for $\overline{\text{RXENB}}$ Deasserted by Converter for 1 Clock Cycle

Ingress Operation for 64-Byte Packet Flow Control

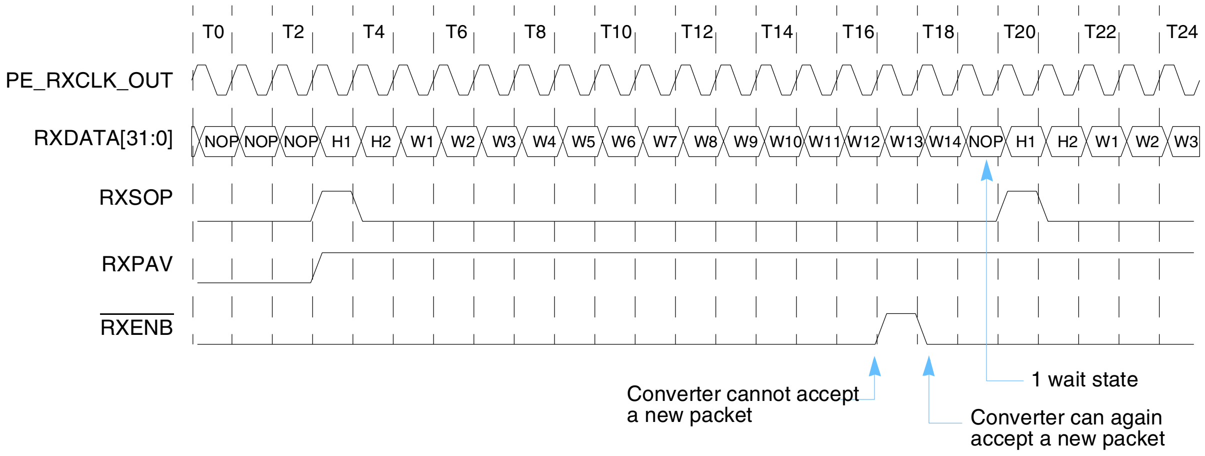

T0
T2
T4
T6
T8
T10
T12
T14
T16
T18
T20
T22
T24
PE_RXCLK_OUT
RXDATA[31:0]
NOP NOP NOP H1 H2 W1 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 NOP H1 H2 W1 W2 W3
RXSOP
RXPAV
RXENB
1 wait state
Converter cannot accept a new packet
Converter can again accept a new packet

Figure 6: Ingress Timing for $\overline{\text{RXENB}}$ Deasserted by Converter for 3 Clock Cycles

Ingress Operation for 64-Byte Packet Flow Control

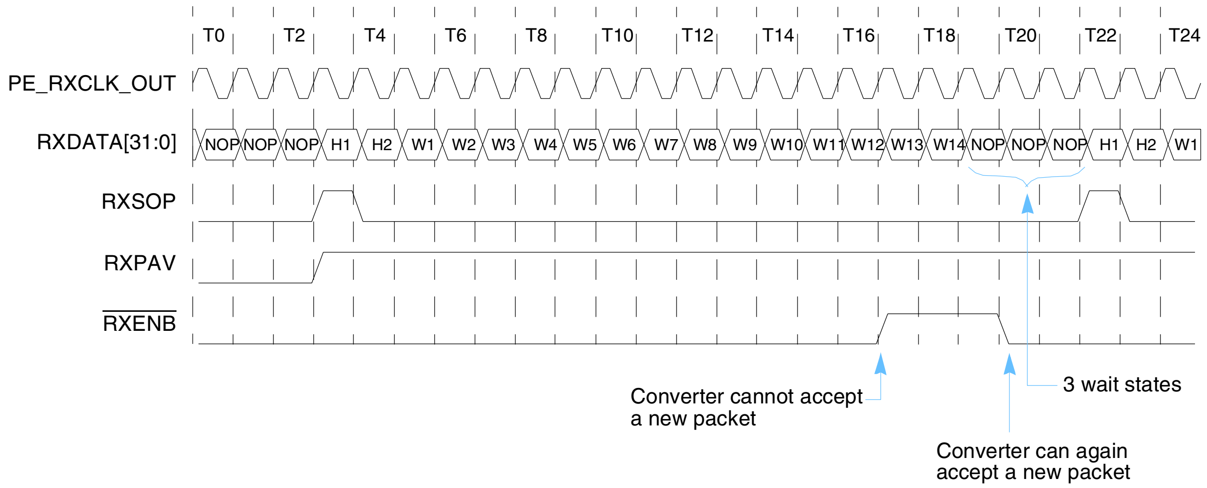

T0
T2
T4
T6
T8
T10
T12
T14
T16
T18
T20
T22
T24
PE_RXCLK_OUT
RXDATA[31:0]
NOP NOP NOP H1 H2 W1 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 NOP NOP NOP H1 H2 W1
RXSOP
RXPAV
RXENB
Converter cannot accept a new packet
3 wait states
Converter can again accept a new packet

Figure 7: Ingress Timing for RXPAV Deasserted by Protocol Engine for 1 Clock Cycle

Ingress Operation for 64-Byte Packet Flow Control

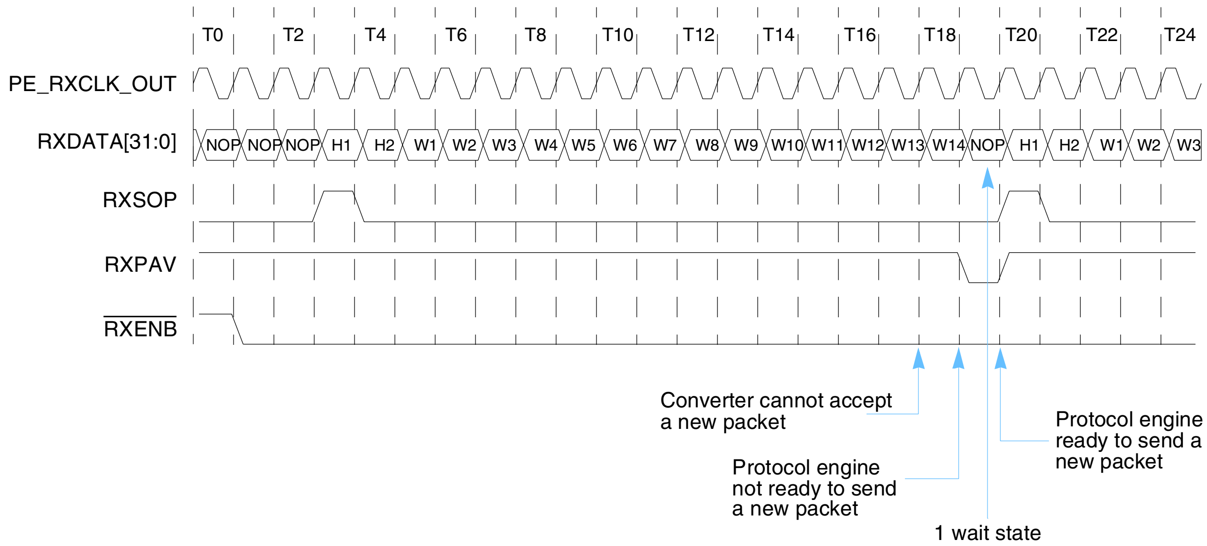

T0
T2
T4
T6
T8
T10
T12
T14
T16
T18
T20
T22
T24
PE_RXCLK_OUT
RXDATA[31:0]
NOP NOP NOP H1 H2 W1 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 NOP H1 H2 W1 W2 W3
RXSOP
RXPAV
RXENB
Converter cannot accept a new packet
Protocol engine not ready to send a new packet
Protocol engine ready to send a new packet
1 wait state

Figure 8: Ingress Timing for RXPAV Deasserted by Protocol Engine for 3 Clock Cycles

Ingress Operation for 64-Byte Packet Flow Control

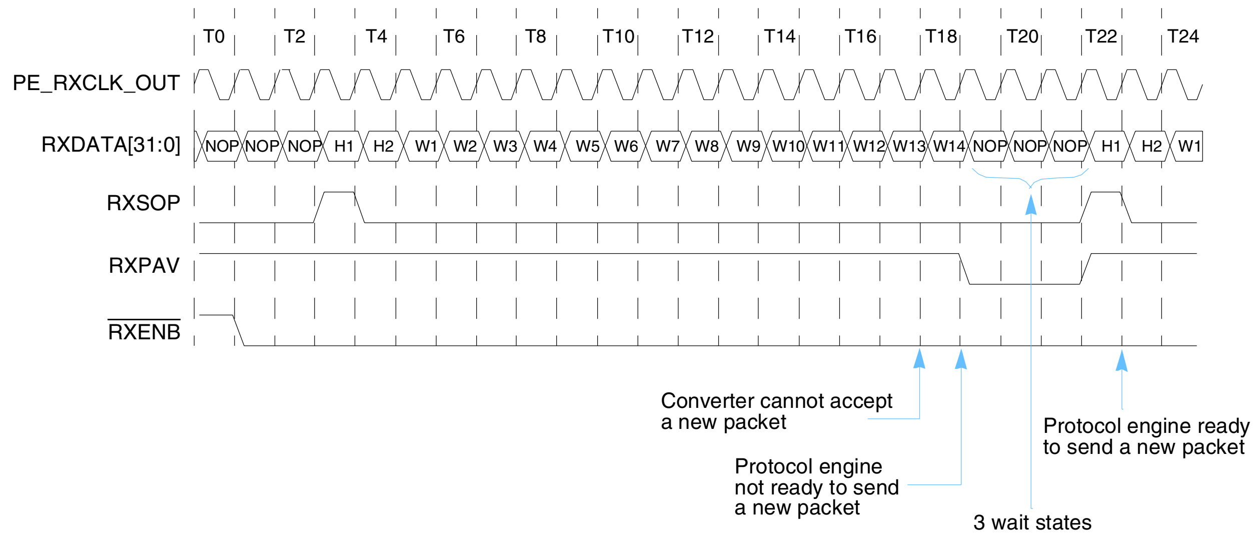
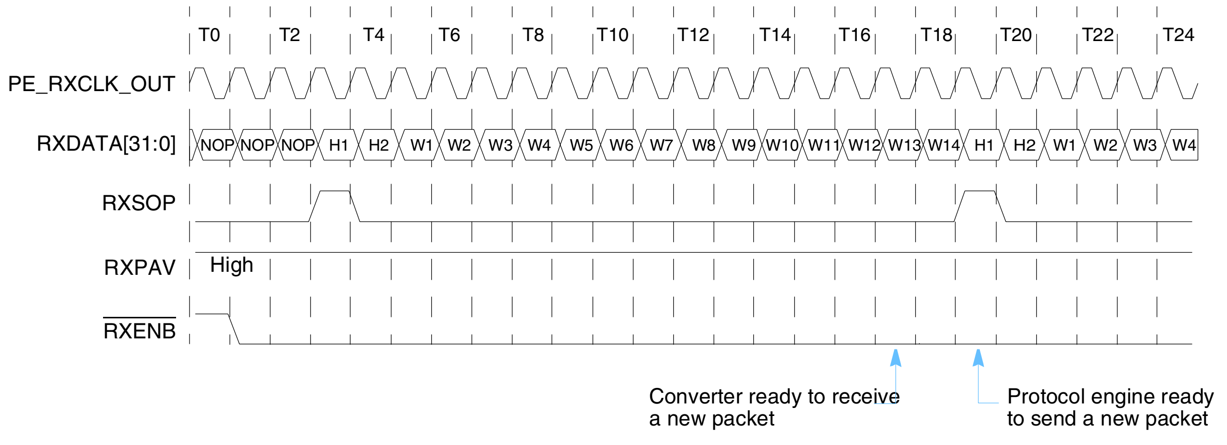

T0
T2
T4
T6
T8
T10
T12
T14
T16
T18
T20
T22
T24
PE_RXCLK_OUT
RXDATA[31:0]
NOP NOP NOP H1 H2 W1 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 NOP NOP NOP H1 H2 W1
RXSOP
RXPAV
RXENB
Converter cannot accept a new packet
Protocol engine not ready to send a new packet
Protocol engine ready to send a new packet
3 wait states

Figure 9: Ingress Packets in Back-to-Back from Protocol Engine

Receive Operation for 64-Byte Back-to-Back Packets


T0
T2
T4
T6
T8
T10
T12
T14
T16
T18
T20
T22
T24
PE_RXCLK_OUT
RXDATA[31:0]
NOP NOP NOP H1 H2 W1 W2 W3 W4 W5 W6 W7 W8 W9 W10 W11 W12 W13 W14 H1 H2 W1 W2 W3 W4
RXSOP
RXPAV
High
RXENB
Converter ready to receive a new packet
Protocol engine ready to send a new packet

3.2.2 Egress Interface

3.2.2.1 Bus Protocol

The egress block is the transmit path between the IBM Packet Routing Switch Serial Interface Converter (the converter) and the protocol engine (PE). The converter sends data according to the following protocol:

- The converter provides the transmit clock called PE_TXCLK_OUT
- The PE asserts the TXPAV signal when it is ready to receive at least one packet from the converter
- When the converter is ready to send at least one packet on the bus and the PE device has asserted the signal TXPAV, it starts the transfer by simultaneously asserting the signal TXSOP and $\overline{\text{TXENB}}$
- TXDATA[31:0] is transferred on each low-to-high clock transition, the first data word of the packet being transferred simultaneously with the signal TXSOP
- The PE device de-asserts the TXPAV signal *at least four cycles before the end of the current packet transfer* to indicate that it cannot accept an immediate transfer of the subsequent packet

This protocol applies when a user wishes to use OBFC mechanisms in addition to IBFC if the attached PE has two sets of buffers. The small buffer allows the link layer to absorb the lack of synchronization between the UTOPIA-3 interface and the rest of the PE chip. The large buffer absorbs traffic burst. Under pure IBFC operation, because all the flow control is performed in band (through the packet header), there is no need for the use of $\overline{\text{TXENB}}$ / $\overline{\text{TXFULL}}$ (TXPAV) protocols. Also under IBFC, if there is no data packet to be sent by the converter, it will insert an idle packet that will be discarded by the PE.

Table 2: Egress I/O Pin Description

Pin Name	Type	Function
PE_TXCLK_OUT	Output	Transmit clock (PE_TXCLK_OUT clock) is issued from two clock domains: At POR it is connected to the microprocessor clock until the POR is completed. Generated after POR completion by either the smooth PLL clock out (derived from either SWITCH_PLL X or Y), or by an externally provided clock (for example the PE clock that ranges from 50 - 125 MHz). PE_TXCLK_OUT source is selected according to the programming of the “PE_TXCLK_OUT_source_l[1:0]” bits as given in the configuration table registers.
TXDATA[31:0]	Output	Transmit Data is transferred from the converter to the PE on a 32-bits word basis. TXDATA[31] is the MSB, TXDATA[0] is the LSB.
TXPRTY (Output)	Output	Transmit Data Parity bit serves as odd parity bit over the 32 TXDATA bits. Parity generation mode is enabled/disabled by the TXPRTY_enb_l bit in the configuration table registers.
TXPAV TxFULL	Input	These are the same signal, but can have two names. Transmit Packet Available/TXFULL is asserted by the PE device when it is ready to receive at least one complete packet. During a packet transfer, the PE device has at least four cycles before the end of the current packet transfer to assert TXPAV if it can accept immediate transfer of the subsequent packet, or to de-assert TXPAV if it cannot. Once the converter detects TXPAV de-asserted, it may only transmit four more 32-bits data words to the PE device. It is recommended that the PE device keeps TXPAV signal asserted until four cycles before the end of the packet transfer. TXPAV and TXFULL lines have the same timing. TXPAV asserted is equivalent to TXFULL de-asserted.
TXSOP	Output	Transmit Start Of Packet is asserted for one clock cycle by the converter when it starts a packet transfer to indicate the first available 32-bits data word of the packet.
TXENB	Output	The converter asserts Transmit Enable to indicate that valid 32-bits data words are on the bus. When TXPAV is asserted during a packet transfer (at least four clock cycles before the end of the current packet transfer), the converter indicates *one clock cycle after the last word of the current packet* if it is (TXENB asserted) or is not (TXENB de-asserted) ready to send a new complete packet. When TXPAV is de-asserted during a packet transfer (at least four clock cycles before the end of the current packet transfer) TXENB is de-asserted four clock cycles afterwords to stop the transfer.

3.2.2.2 Egress Operation and Timing

Figure 10: TXFULL Timing

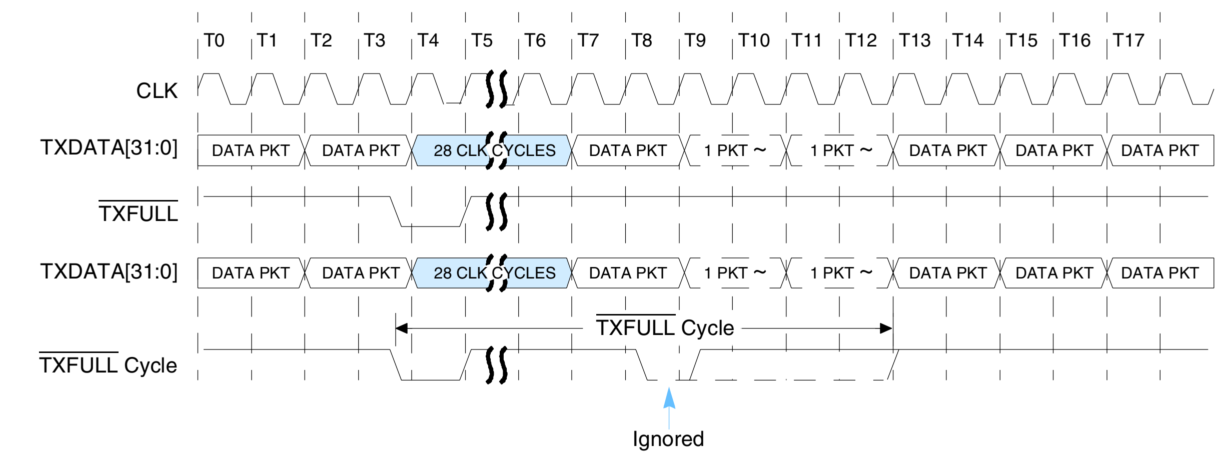
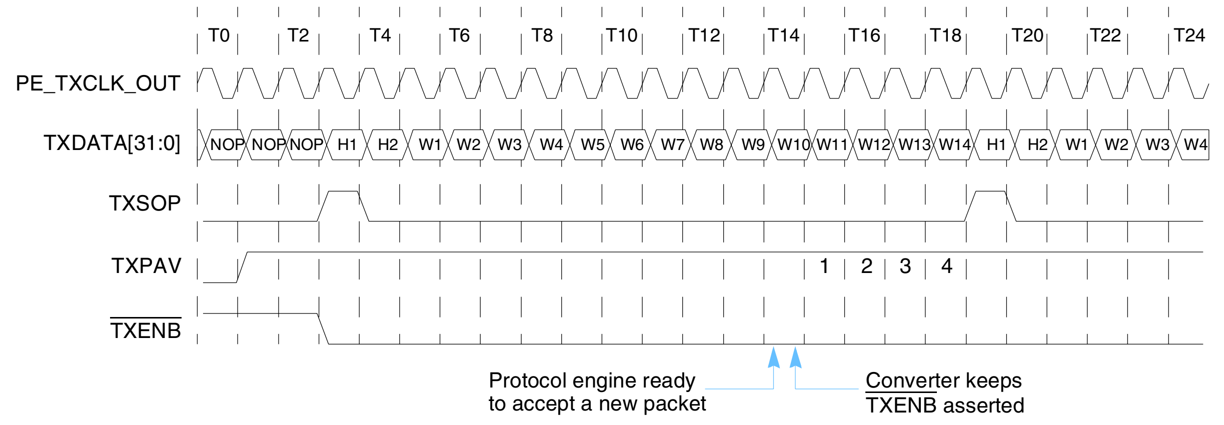


Figure 11: Egress Timing for Back-to-Back Packets
Transmit Operation for 64-Byte Back-t0-Back Packet Transmission



3.2.3 TXFULL Timing

When the PE asserts TXFULL to stop the flow of packets, its effect is extended for 28 PE clock cycles from the end of the current packet. After the first packet following the deassertion of TXFULL by the PE, the UDASL inserts two empty slots of idle (no data) before the traffic resumes. If the PE resasserts TXFULL upon the reception of the first packet during the TXFULL cycle, it will be ignored until the end of the complete cycle.

3.3 Packet Reshuffling

3.3.1 Ingress Receive Logical Unit Framing

The Ingress LU formatter is the interface between the PE interface logic and the ingress FIFO (RXFIFO) which translates various types of ingress packet formats into the IBM 28.4 G Packet Routing Switch (switch) LU format is defined in the IBM Packet Routing Switch Serial Interface Converter (the converter) configuration registers.

Data inputs are in a 4-byte format. The ingress LU formatter extracts TxPause flow control information from the incoming packet qualifier byte. The incoming packet is comprised of 16 to 20 32-bit words. Packet Reshuffling stores the packet in the switch format in four LUs which, when fully generated, are forwarded to the RXFIFO.

The Ingress LU Formatter:

- Discard idle packets
- Extract Send Grant flow control information from the incoming PQ byte
- Modify the packet qualifier byte into switch format and compute the associated parity
- Change the position of the bit map bytes in the packet header
- Know of any bus parity error through the PE interface RXPRTY_error
- May not forward a packet to the RXFIFOs in case of any bus or header error (depending upon a configuration register setting)

3.3.1.1 Header Bytes Reshuffling

The header bytes are moved according to the content of the "byte positioning in LU formatter" configuration fields. These allow any byte contained in the four words to be moved to any other byte position in the master LU.

3.3.1.2 Input/Output Packet Format

The following tables highlight the position of the different information fields in an incoming packet. All bytes in the first four words can be repositioned in the switch header according to the formatter field in the configuration registers. The data source is based on a word/byte coordinate in a nibble. The first two bits correspond to the word selection; the last two bits correspond to the byte selection.

Example of Header Bytes in Sequence

Byte	Byte 0	Byte 1	Byte 2	Byte 3
	MSB			LSB
Bit	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Word 0	Packet Qualifier	BM[31:24] BM1	BM[23:16] BM2	BM[15:8] BM3
Word 1	BM[7:0] BM4			
Word 2				
Word 15				

When the bitmap field is defined for 32-bits but only 16 are used, only bits 16 through 31 of the BM field are required to identify the 16 switch core ports. The others are considered to be part of the payload. The parity sub-field of the Packet Qualifier field is calculated across the first three or five bytes (depending upon the configuration).

For example, with header bytes in sequence the content of the bit positioning registers (@C8 for ingress and @C4 for the egress) will be 0, 1, 2, 3, 1, as shown in the table below:

Header Bytes	Header Bytes in Incoming Packets		Content of Bit Positioning Register	Notes
Packet qualifier	word 0	byte 0	0	
Bit Map 1	word 0	byte 1	1	
Bit Map 2	word0	byte 2	2	
Bit Map 3	word0	byte 3	3	
Bit Map 4	word1	byte 0	1	1

1. This last index is not 4 because during the second move operation this byte was moved from word 1 position 0 to word 0 position 1

The full packet is stored in the ingress LU formatter. Reshuffling starts when the first five words are received. They are stored again in another set of five word buffers. The header information on which the switch acts to route the packet to the appropriate output port is moved to the appropriate byte location in the master LU during the cycles required to store the current packet's remaining words. Byte swapping is done according to the 16 bits stored in the configuration register @C8 (byte positioning in LU formatter).

Example of Bit Map Field in Single Word

Byte	Byte 0	Byte 1	Byte 2	Byte 3
	MSB			LSB
Bit	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Word 0	Packet Qualifier			
Word 1	BM[31:24] BM1	BM[23:16] BM2	BM[15:8] BM3	BM[7:0] BM4
Word 2				
Word 15				

When 16 (or 8)-port mode is used, only bits 31 - 16 of the BM field are required to identify the 16-switch core ports. The others are considered to be part of the payload. The outgoing packet is made of four logical units (LU0, LU1, LU2, and LU3), each 16-bytes wide. The following tables show how the bytes of the incoming packet are rearranged inside the four logical units. Each line represents the content of each LU.

8x8 and 16x16 Switch LU Output Format:

Container		C00	C01	C02	C03	C04	C05	C06	C07	C08	C09	C10	C11	C12	C13	C14	C15
Master LU0	Byte 0	PQ	BM1	BM2	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU1	Byte 1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU2	Byte 2	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU3	Byte 3	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

32x32 Switch LU Output Format:

Container	C00	C01	C02	C03	C04	C05	C06	C07	C08	C09	C10	C11	C12	C13	C14	C15
Master LU0	PQ	BM1	BM2	BM3	BM4	D	D	D	D	D	D	D	D	D	D	D
Slave LU1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU2	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU3	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

When the switch is in 32x32 mode, all bytes of the Bitmap are used (BM1, BM2, BM3, and BM4) for both port addressing and to compute the header parity. When the switch is in 16x16 mode, only BM1 and BM2 are used while BM3 and BM4 are considered as payload and are ignored for the computation of the header parity. When the switch is in 08x08 mode, only BM1 is used while BM2 must be ’00’ so it does not impact the computation of the parity in packet qualifier.

3.3.1.3 Port Addressing

The following table shows the switch output port addressing (a binary 1 in the bit map field indicates that the packet should be sent to the corresponding switch output port).

Table 3: Output Port Bitmap Fields

UTOPIA-3 Bit Notation	7	6	5	4	3	2	1	0
IBM Packet Routing Switch/ DASL Bit	0	1	2	3	4	5	6	7
Output port bit map 1	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7
Output port bit map 2	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15
Output port bit map 3	Port 16	Port 17	Port 18	Port 19	Port 20	Port 21	Port 22	Port 23
Output port bit map 4	Port 24	Port 25	Port 26	Port 27	Port 28	Port 29	Port 30	Port 31

3.3.2 Nested TxPause Extraction

In IBFC, The bits PQ (26,27 and 30,31) of the packet qualifier are monitored to extract the converter egress flow control information (TxPause). The TxPause information is used by the PE to flow control the switch’s output port. The converter interprets those bits and translates them into Sent Grant to the switch. The use of TxPause is dependent upon the setting of register @C8 bit 30 of the configuration table register. When Bit 3 (Send Grant per Priority) of configuration register table @C4 is not enabled, any bit set to '0' in the TxPause field is sufficient to stop transmission to the attached protocol engine.

3.3.3 Ingress Packets for IBFC

Figure 12: Example of Converter Ingress Idle Packet

Byte	Byte 0								Byte 1								Byte 2								Byte 3							
	MSB																															LSB
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word 0	Packet Qualifier								1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
Word 1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
Word 2	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
Word 15	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0

Table 4: Packet Qualifier for Ingress Idle Packet

Idle Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	TxPause Bit 0	TxPause Bit 1	0	0	TxPause Bit 2	TxPause Bit 3	0	0
TxPause Priority Bitmap								
Highest	1	0			0	0		
Medium High	0	1			0	0		
Medium Low	0	0			1	0		
Low	0	0			0	1		
All	1	1			1	1		
None	0	0			0	0		

Table 5: Packet Qualifier for Ingress Data Packet

Data Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	TxPause Bit 0	TxPause Bit 1	Active Bit ‘1’	Backup Bit ‘1’	TxPause Bit 2	TxPause Bit 3	Packet Priority Bits	
TxPause Priority Bitmap								
Highest	1	0			0	0		
Medium High	0	1			0	0		
Medium Low	0	0			1	0		
Low	0	0			0	1		
All	1	1			1	1		
None	0	0			0	0		

Table 6: Packet Qualifier for Ingress Data Packet

Data Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	0	Header Parity	Color/Type	Color/Type	0	0	Packet Priority Bits	
Color/Type								
Red Complemented			0	1				
Red			1	0				
Blue			1	1				
Packet Priority								
Highest							0	0
Medium High							0	1
Medium Low							1	0
Lowest							1	1

3.3.4 Egress Packet Formatter

The Egress Packet Formatter is the interface between the egress FIFO and the PE interface logic which translates the switch LU format into the egress packet format that matches the format expected by the PE.

The Egress Packet Formatter:

- Changes the position of the proper data bytes in the packet which have been moved in the ingress path
- Inserts the switch Output Queue Grant & Shared Memory Flow control information at the byte position occupied by the bit map and some of the packet qualifier bits
- Inserts idle packets when there is no data packet to be sent

Figure 13: Example of Converter Egress Idle Packet

Byte	Byte 0								Byte 1								Byte 2								Byte 3							
	MSB																															LSB
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Word 0	Packet Qualifier								1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
Word 1	Output Queue Grant [31:24]								Output Queue Grant [23:16]								Output Queue Grant [15:8]								Output Queue Grant [7:0]							
	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
Word 15	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0

Table 7: Packet Qualifier for Egress Idle Packet

Idle Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	Shared Memory Grant Priority Level		Shared Memory Status Hold = '0' Grant = '1'	Output Queue Grant Priority Indicator		0	0	0
Memory Grant Priority Level Definition								
Highest	0	0						
Medium High	0	1						
Medium Low	1	0						
Lowest	1	1						
Output Queue Grant Priority Indicator								
Highest				0	0			
Medium High				0	1			
Medium Low				1	0			
Lowest				1	1			

Table 8: Packet Qualifier for Egress Data Packet

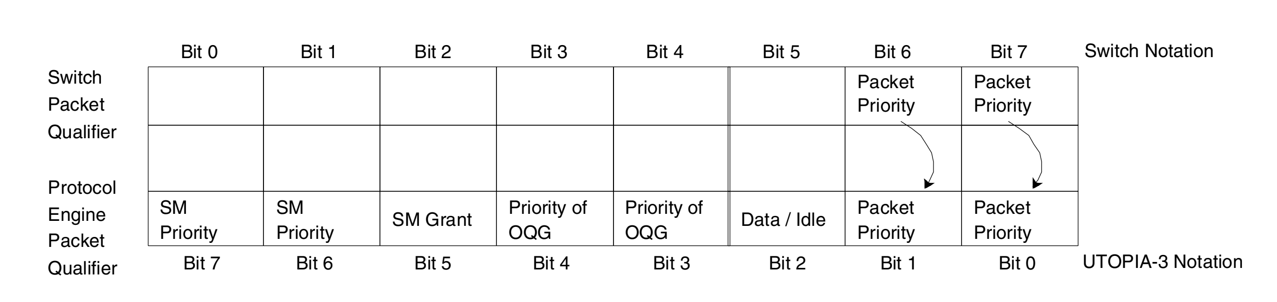
Data Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	Shared Memory Grant Priority Level		Shared Memory Status Hold = ‘0’ Grant = ‘1’	Output Queue Grant Priority Indicator		0	0	0
Memory Grant Priority Level Definition								
Highest	0	0						
Medium High	0	1						
Medium Low	1	0						
Lowest	1	1						
Output Queue Grant Priority Indicator								
Highest				0	0			
Medium High				0	1			
Medium Low				1	0			
Lowest				1	1			
Data Packet Priority Bit								
Highest							0	0
Medium High							0	1
Medium Low							1	0
Lowest							1	1

3.3.4.1 Building the Egress Packet Qualifier Byte

- Shared Memory Information
  - The bit PQ(2) is replaced by the shared memory information (SM) whose priority level corresponds to shared memory grant priority levels PQ(0) and PQ(1) that cycle one priority after another.

- Output Queue Grant Priority Indicator
  - The first two bits PQ(3) and PQ(4) are loaded from the Output Queue Grant Priority Level values.
  - The other bits PQ(6) and PQ(7) are unchanged, as they belong to the data information contained in the packet.

Figure 14: IBM 28.4 Packet Routing Switch (switch) Packet Qualifier Bits Reshuffling



	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Switch Notation
Switch Packet Qualifier							Packet Priority	Packet Priority	
Protocol Engine Packet Qualifier	SM Priority	SM Priority	SM Grant	Priority of OQG	Priority of OQG	Data / Idle	Packet Priority	Packet Priority	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	UTOPIA-3 Notation

- Output Queue Grant Bytes Description
  - The IBM Packet Routing Switch Serial Interface Converter egress block provides the OQG values. There are 16 or 32 bits for each priority.
  - For the IBM 28.4 Packet Routing Switch, the first OQG sent is an idle packet's priority level 00. The OQG mechanism gives the priority level in the idle packet's qualifier byte. The other OQGs are sent in data packets (or idle packets) and the process is looped. In every idle packet sent, the flywheel mechanism gives the priority level to check the position in the loop.
  - The latest refreshed OQG bits are inserted into the packet header to be sent to the protocol engine.

Table 9: Packet Qualifier for Egress Data Packet

Data Packet	Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24
Bit Meaning	0	Header Parity	Color/Type	Color/Type	0	0	Packet Priority Bits	
Color/Type								
Red Complemented			0	1				
Red			1	0				
Blue			1	1				
Packet Priority								
Highest							0	0
Medium High							0	1
Medium Low							1	0
Lowest							1	1

3.4 Packet Buffering

3.4.1 X and Y Path Receive FIFO (RXFIFO)

The RXFIFO block:

- Writes the data coming from the ingress formatter (clocked on the PE clock (URXCLK)) to the FIFO
- Reads the data from FIFO to the X and Y paths to the ingress DASL interface (IDI) block, clocked on the X/Y path clock (accounting for the size of the packet in steps of four bytes)
- Detects and reports a FIFO almost full event, whose threshold is programmable through configuration registers

- Depending upon the configuration table, it either de-asserts RXENB (if it is in use), or de-grants Shared Memory priority 0 (highest) to create room in the RXFIFO when there is an indication of FIFO almost full.

The FIFO is 256 bytes wide, or three 80-bytes packets. A 3-packet buffer temporarily stores incoming data packets, thereby allowing the insertion of a yellow packet without requiring any link level flow control. This absorbs any clock difference between the PE and the switch

3.4.2 X/Y Path Transmit FIFO (TXFIFO)

The X and Y path TXFIFO block:

- Writes the data coming from the X or Y path TX EXTRACTION BLOCK, as clocked on DASL_X_CLK or DASL_Y_CLK respectively, to the FIFO
- Reads the data from the FIFO to the path selection block, clocked on UTXCLK (PE clock)
- Detects and reports a FIFO almost full event, whose threshold is programmable
- Detects and reports a FIFO not empty, whose threshold is programmable
- Stops its data flow when the path selection block decides

Data packets are never written in the unused path’s FIFO.

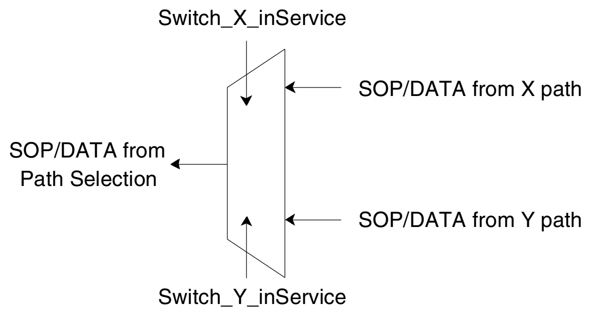
When the FIFO is almost full, send grant to the switch is de-asserted.

The FIFO is 480 bytes wide, i.e. six 80-bytes packets. Six packets of buffer in the TX FIFO take into account the switch’s latency in reacting to the send grant (three packets in the worst case). This is true when asserting or de-asserting send grant.

3.5 Path Selection Block

This block selects the X or Y FIFO depending on the switch control’s X_inService or Y_inService inputs. Only the selected path’s TXFIFO is filled with packets, the other one is empty. Switching from one plane to the other is done on a packet boundary. However, there can be duplicated or missing packets.

Figure 15: Path Selection



-X_inService	True	False	True	False
-Y_inService	False	True	True	False
Status of the selector	Path X is selected	Path Y is selected	No path selected	No path selected

The attached microprocessor polls those bits to check the validity of the path selection.

Force path selection	0	1	1
Select X/Y 0 select X path 1 select Y path	Don’t care X_/Y_inService is valid	0	1
Status of the selector	Selection based on physical wires	Path X is selected	Path Y is selected

Use this scheme to force any path with the force bit to perform a switch-over at switch core level and to verify that the X_/Y_inService lines operate properly.

3.6 Interfacing Data Aligned Serial Link (DASL) Macro

3.6.1 Ingress Data Aligned Serial Link (DASL) Interface

The ingress DASL interface logic (IDI) is the interface between the ingress FIFO and the DASL macro. The IDI feeds the DASL macro with data packets following a DASL packet request. When no packet is available to serve the DASL, the IDI should provide an idle packet to the DASL macro. Data packets sent to the DASL macro come from the ingress FIFO. The IDI is the packet clock provider to the output logic of the ingress FIFO and to the DASL macro. The IDI also triggers the ingress FIFO output scheduler operation.

During the initial training sequence required to synchronize the remote switch DASL, the IDI provides ’synchronization packets’. The IDI is also designed to insert link liveness packets (liveness function covered by yellow packets) into the flow of packets sent to the DASL box following a request from the control interface. When data mode is activated, the IDI is fed with any data packet, otherwise it generates an idle packet (lack of data packet). Idle packets carry CRC (one per LU) to protect logical unit transport media (IBM Packet Routing Switch Serial Interface Converter and IBM 28.4 G Packet Routing Switch).

3.6.2 Functions

- Generate Idle Packet: no data available from ingress FIFO
- Generate Synchronization Packet (when DASL training sequence is being run)
- Generate Yellow Packet (when requested by control interface)
- Forward Data Packets
- LU protection CRC insert (within idle packets)
- Provide Packet Clock to ingress FIFO
- Provide Packet Request to ingress FIFO Output Scheduler
- DASL interface LU/Byte/Bit map compatible

3.6.3 Packets Format

3.6.3.1 Idle Packets

Idle packets are generated by IDI logic when no data packet is available from the ingress path. Idle packets have the following format:

Master LU	PQ	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC

The Master LU must be routed to the switch operating as master.

Note: Yellow packets sent to the switch are considered idle packets and therefore contain CRC trailers.

3.6.3.2 Idle Packet CRC Computing

Idle packet CRC detects physical media errors, so 4x LUs must be covered. The last byte of every LU within an idle packet carries a CRC byte protecting each LU.

The CRC polynom is X8+X4+X3+X2+1. CRC register is initialized (software configuration) depending on the LU depth.

Considering an LU, the CRC byte when inserted is computed from the end of the previous idle packet sent on the DASL interface (end of cell boundary, CRC byte), and up to the current idle packet last byte (byte preceding CRC location).

3.6.3.3 Synchronization Packets Format

Synchronization packets synchronize DASLs. Characteristics of this packet allows the remote DASL operating as receiver to recover bit transition and packet delineation (packet clock recovery). Synchronization packet format is as follows:

Master LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’33’
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’33’
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’33’
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’33’

3.6.3.4 Data Packets

- Data Packet Format Switch 8x8 and 16x16

Master LU	PQ	BM0	BM1	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

- Data Packet Format Switch 32x32

Master LU	PQ	BM0	BM1	BM2	BM3	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

3.6.4 Egress Data Aligned Serial Link (DASL) Interface (EDI)

The IBM Packet Routing Switch Serial Interface Converter’s packets have a fixed length of 64 - 80 bytes (based on the setting in the configuration register) and are mapped on 4-byte words. Therefore a new packet is received every 16 to 20 clock cycles from the DASL. When a complete packet has been received, the 16 to 20 words are transferred to the egress FIFO.

The EDI analyzes the IBM 24.8 G Packet Routing Switch (switch) packet header from the master LU to indicate the presence of an idle packet or data packet and the packet priority. Idle packets are discarded when received, (no packet write request is presented to the egress FIFO). The EDI block also checks the LU CRC (mapped into Idle Packets LU trailer). When an error is detected, the converter interrupt line is asserted (if checker enabled) and the Idle CRC error counter is incremented.

The EDI checks switch header parity. An error count is incremented when an error is detected and the processor interruption line is asserted (may be masked). Optional "error" packet discard function is provided.

The EDI performs switch grant extraction from the switch packet header. A grant information bit map is carried into the switch output packet header and is reported per priority. Therefore grant information is refreshed over four packet periods (if the four priorities are in use) for a given priority. A flywheel counter mapped into the Idle Packet Qualifier synchronizes the converter with the switch counter.

The EDI GRANT extraction mechanism is synchronized on the incoming idle packet grant flywheel counter. When a desynchronization problem is detected between the switch counter and the EDI grant flywheel counter, an error is reported and the EDI flywheel counter is automatically resynchronized. In order to minimize the latency of the switch Grant mechanism, the Grant vector is extracted before the completion of the packet reception.

The EDI performs switch memory grant synchronization. Memory Grant information and Destination Grant Vectors are passed to the in band grant generation block for insertion into the egress packets. The EDI is also the protocol engine interface wrap point.

EDI functions:

- Detect idle packets received from the switch
- Synchronize on switch Output Queue Grant counter
- Output Queue Grant Extraction (one to four priorities traffic)
- Yellow Packet Received Indication (Maskable error Interruption)
- LU CRC checking (Maskable error interruption)
- Header Parity checking (Maskable error interruption) with optional discard
- Data Packet Received Detection
- Wrap Indication Input line (Grant Indication Bypass)

3.6.4.1 Packets Format

Idle Packets

Idle packets received from EDI have the following format:

16x16 Switch Interface

Container	C00	C01	C02	C03	C04	C05	C06	C07	C08	C09	C10	C11	C12	C13	C14	C15
Master LU	PQ	OQG1	OQG2	x’00’	x’00’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’00’	x’00’	x’00’	x’00’	CRC
Slave LU	x’CC’	x’00’	x’00’	x’00’	x’00’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’00’	x’00’	x’00’	x’00’	CRC
Slave LU	x’CC’	x’00’	x’00’	x’00’	x’00’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’00’	x’00’	x’00’	x’00’	CRC
Slave LU	x’CC’	x’00’	x’00’	x’00’	x’00’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’00’	x’00’	x’00’	x’00’	CRC

For idle packets PQ is mapped as follows:

Bit_ 0	Bit_1	Bit_2	Bit_3	Bit_4	Bit_5	Bit_6	Bit_7
0	Header Parity PQ OQG1 OQG2	0	0	00 (Blue Colored Packet) 01 (Red Colored Packet)		00 Highest Priority 01 10 11 Lowest Priority Flywheel Counter Synchronization	

Parity Bit is EVEN parity over 3 bytes PQ, OQG1 and OQG2

32x32 SWITCH Interface

Master LU	PQ	OQG1	OQG2	OQG3	0QG4	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC
Slave LU	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	x’CC’	CRC

In the packet qualifier the parity bit is EVEN parity over 5 bytes PQ, OQG1, OQG2, OQG3 and OQG4

Note: The content of PQ byte for yellow packets received from the switch is ’01001000’. It does not contain flywheel synchronization information because it is generated by the switch control. In the switch, egress yellow packets are considered data packets and therefore do not contain link CRC information fields. Yellow packets are detected and an interruption is reported when the detection function is enabled. When detection is disabled, the yellow packet is considered a normal time fill packet (idle packet).

Idle Packet CRC Computing

Idle packet CRC detects physical media errors, so the 4x LUs must be covered. The last byte of each LU for each idle packet carries a CRC byte protecting each LU. The four LU CRC bytes are cumulative between two idle packets.

The CRC polynom is X8+X4+X3+X2+1. The CRC register is initialized (configuration table address 08 @ 28, bits 24-31) depending on the LU depth.

3.6.4.2 Data Packets

The data packet format can be made of LUs of 16 to 20 bytes long depending on the configuration register.

Data Packet Format Switch 8x8 and 16x16

Master LU	PQ	OQG0	OQG1	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

Data Packet Format Switch 32x32

Master LU	PQ	OQG0	OQG1	OQG2	OQG3	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Slave LU	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

3.6.5 IBM 28.4G Packet Routing Switch (switch) in band Output Queue Grant Information

Bytes BM1 and BM2 carry the output queue grant coming from the switch. This information is carried for all 16 output ports simultaneously for a given priority. Consecutive packets, either data or idle, carry a different priority of the output queue grant bits, cycling from 0 to the highest priority value that is enabled. For instance, when two priorities are enabled, it takes two packets to transmit the output queue grant information.

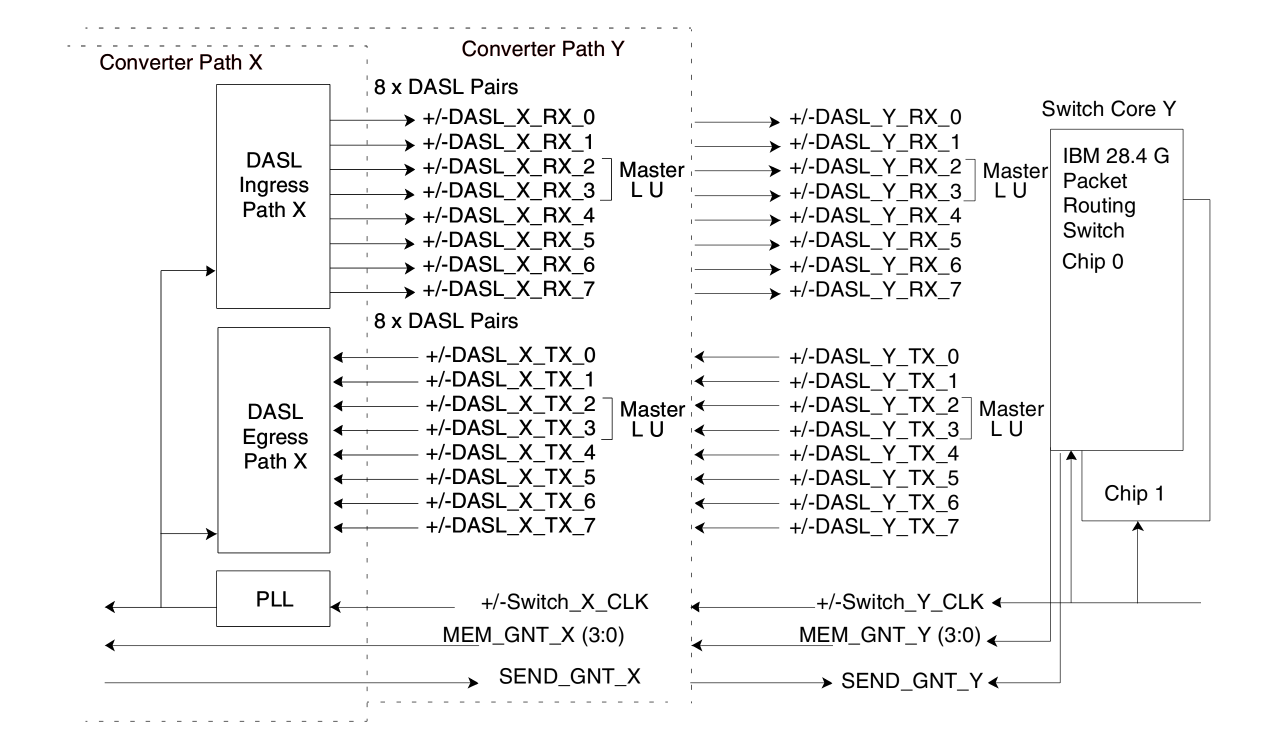
Table 10: Output Queue Grant Bit Map Fields

UTOPIA-3 Bit Notation	7	6	5	4	3	2	1	0
DASL/IBM 28.4G Packet Routing Switch Bit	0	1	2	3	4	5	6	7
Output Queue Grant 1	Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7
Output Queue Grant 2	Port 8	Port 9	Port 10	Port 11	Port 12	Port 13	Port 14	Port 15

When a bit of this OQG field is set to ’1’, data can be sent to the corresponding output queue. When set to ’0’ no data should be sent to that output queue. The switch sets (8-port mode) OQG1 to ’FF’ when operating in internal speed expansion. If a yellow packet is detected when detection is enabled, an interruption is reported. If detection is not enabled the yellow packet is considered an idle packet.

3.6.6 IBM Packet Routing Switch Serial Interface Converter (the converter) Switch Interface

Figure 16: Converter Interface Lines



3.7 Egress & Ingress Interface Diagnostic Functions

3.7.1 Loopbacks

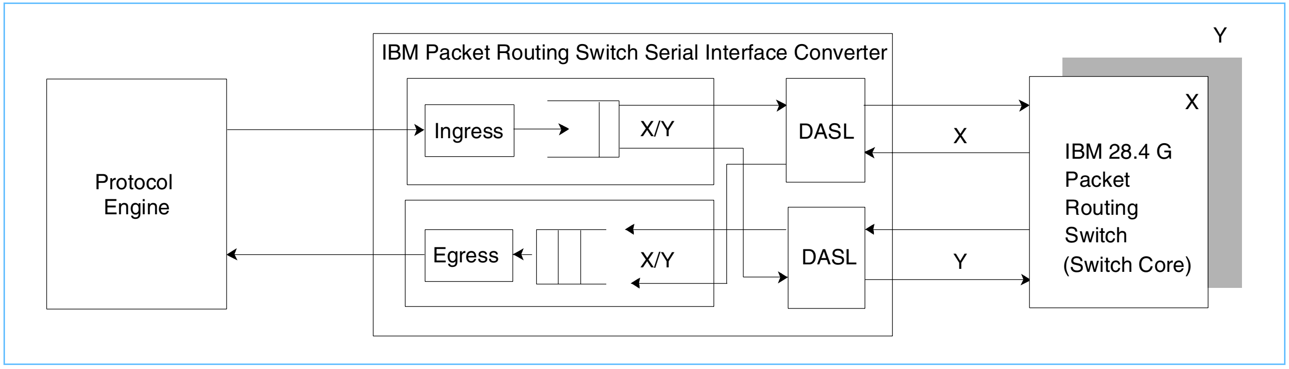
Loopbacks are controlled through configuration registers. There are two possible paths, X and Y, for each loopback, but an exclusive choice must be made and a path reset executed before the loopback is performed. The IBM Packet Routing Switch Serial Interface Converter (the converter) supports two loopback modes:

- Protocol engine (PE) X/Y loopback: the PE sends and checks data and the grant mechanism must be bypassed (forced to all ones through configuration register)
  - Protocol engine X domain loopback (register @00 bit 0 set to '1')
  - Protocol engine Y domain loopback (register @20 bit 0 set to '1')
- Switch loopback: data is initiated from the switch and the switch control verifies the overall operation
  - Switch X and Switch Y domain loopbacks: (register @C4 bit 1 set to '1'). Depending on the status of the force path and select X/Y bits of registers A0 bits 21 and 22 respectively, it is possible to use either the inService line or the select X/Y by means of bit 22 to decide which plane is selected. In the latter case, plane X is used if set to '0' and plane Y is used if set to '1'.

3.7.1.1 Normal Operating Mode

Data sent by the PE to the switch core is transferred to the ingress interface where it is parity checked, LU formatted, and queued to both planes for transmission by the DASL. Data transmitted by the switch core is checked for framing consistency, header parity, queued, and word formatted for transmission by the egress interface to the PE.

Figure 17: Configuration in Normal Operating Mode

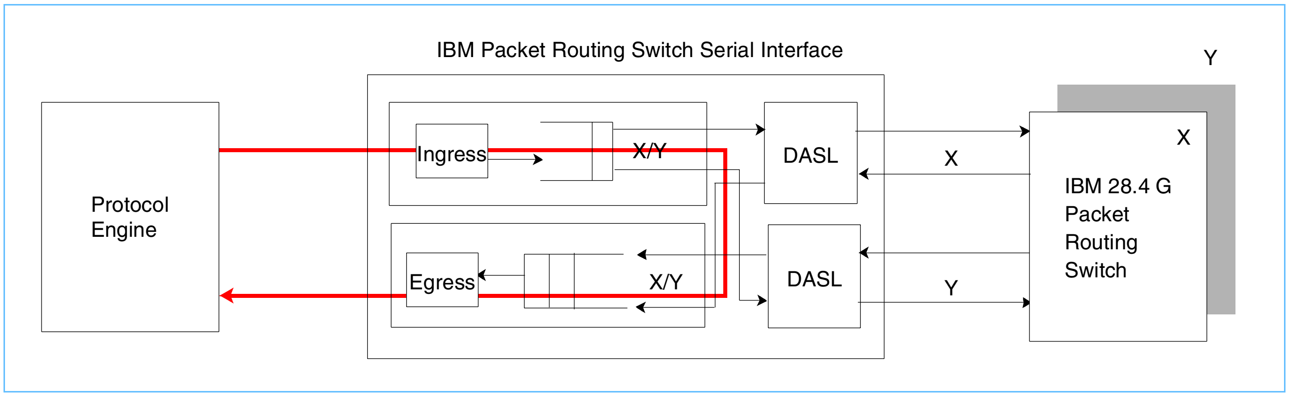

Protocol Engine
IBM Packet Routing Switch Serial Interface Converter
Ingress
X/Y
Egress
X/Y
DASL
DASL
X
Y
Y
X
IBM 28.4 G Packet Routing Switch (Switch Core)

3.7.1.2 Protocol Engine X/Y Loopback

Protocol engine X/Y loopback provides a connection from the ingress protocol engine bus to the egress protocol engine bus. This loopback mode can be activated on either path X or path Y but not on both simultaneously. The converter in loopback mode is internally disconnected from the DASL macro for both the data and control buses and does not test the DASL macro. The loopback is initiated on a packet boundary.

Note: A switch plane (or substitute) clock must be present on the converter’s switch interface.

Figure 18: Protocol Engine Loopback Through Path X or Path Y


IBM Packet Routing Switch Serial Interface
Protocol Engine
Ingress
X/Y
Egress
X/Y
DASL
DASL
X
Y
Y
X
IBM 28.4 G Packet Routing Switch

Perform the following to execute the PE X/Y domain loopback (starting from operational mode):

1. Reset the path to be set in loopback.
2. Force X or Y switch plane in service (switch control can also decide in this mode).
3. Select the clock source (switch or microprocessor) for the plane to be wrapped.
4. Set the PE loopback bit corresponding to the plane to be set in loopback.
5. Initiate the traffic from the PE.

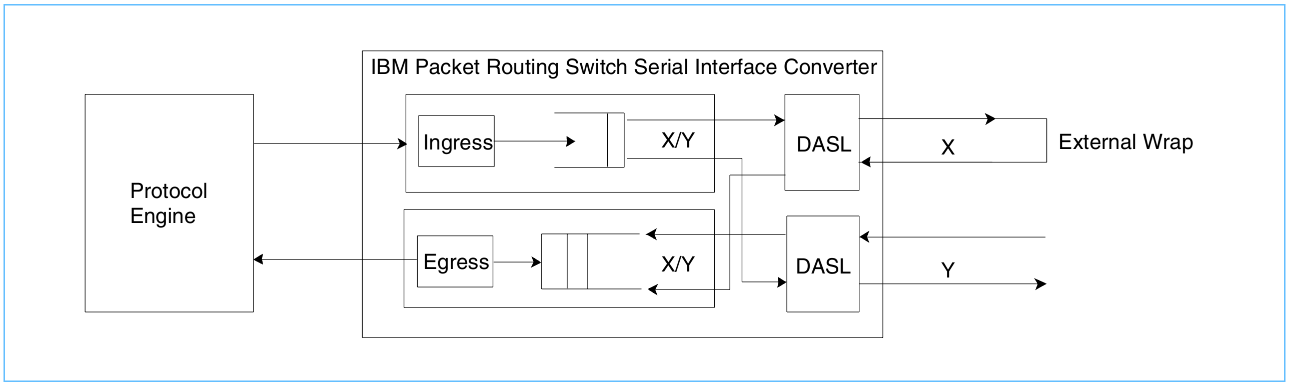
The data transfer must have been stopped for long enough and in such a way that all buffers are empty before the loopback is initiated. If the buffers are not empty, reset the path on which the loopback is to be performed. To execute the loopback, the converter must be set in OBFC mode to avoid modifying the packet qualifier byte.

3.7.1.3 Protocol Engine (PE) External Loopback

The PE bus is wrapped via the DASL interface. The switch is disconnected from the converter. A wrap plug is connected to the DASL port corresponding to the path to be tested.

Note: The switch plane (or substitute) clock must be selected.

Figure 19: Protocol Engine External Loopback Through Path X



The following operational sequence must be performed to execute PE external loopback:

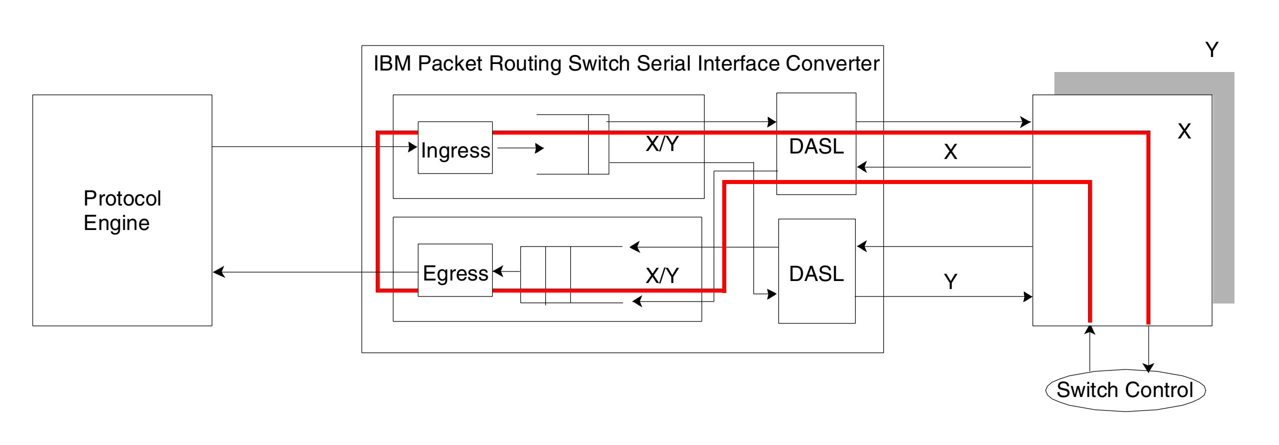
1. Reset the path to be set in loopback.
2. Force X or Y switch plane in service.
3. Select the clock source (switch or microprocessor) for the plane to be wrapped.
4. Synchronize the DASL.
5. Initiate the traffic from the PE.

The PE external loopback can be performed without switch clock by using the microprocessor clock and properly programming the PLLs (the PLL must be reset to acquire its newly programmed value). In all cases the objective is to be in the frequency range of 100 - 125 MHz to allow DASL operation.

3.7.1.4 Switch X/Y Loopback

Switch X/Y loopback provides a connection from the egress input to the ingress output to the ingress input through either the X or the Y path. During this test $\overline{\text{RXENB}}$ is de-asserted so the PE cannot send data.The result of this test is obtained through the switch control.

Figure 20: Switch X Loopback



Perform the following operational sequence to execute the switch loopback (starting from operational mode):

1. Set the converter in OBFC mode.
2. Force X or Y switch plane in service or let the switch control decide which switch plane is in service.
3. Set the switch loopback bit corresponding to the plane to be set in loopback.
4. Initiate the traffic from the switch control.

The data transfer must have been stopped for long enough and in such a way that all buffers are empty before the loopback is initiated. If the buffers are not empty, reset the path on which the loopback is to be performed. To execute the loopback, the converter must be set in OBFC mode to avoid modifying the packet qualifier byte.

3.8 Clocks Generator Description

Figure 21: Clocks Distribution Diagram

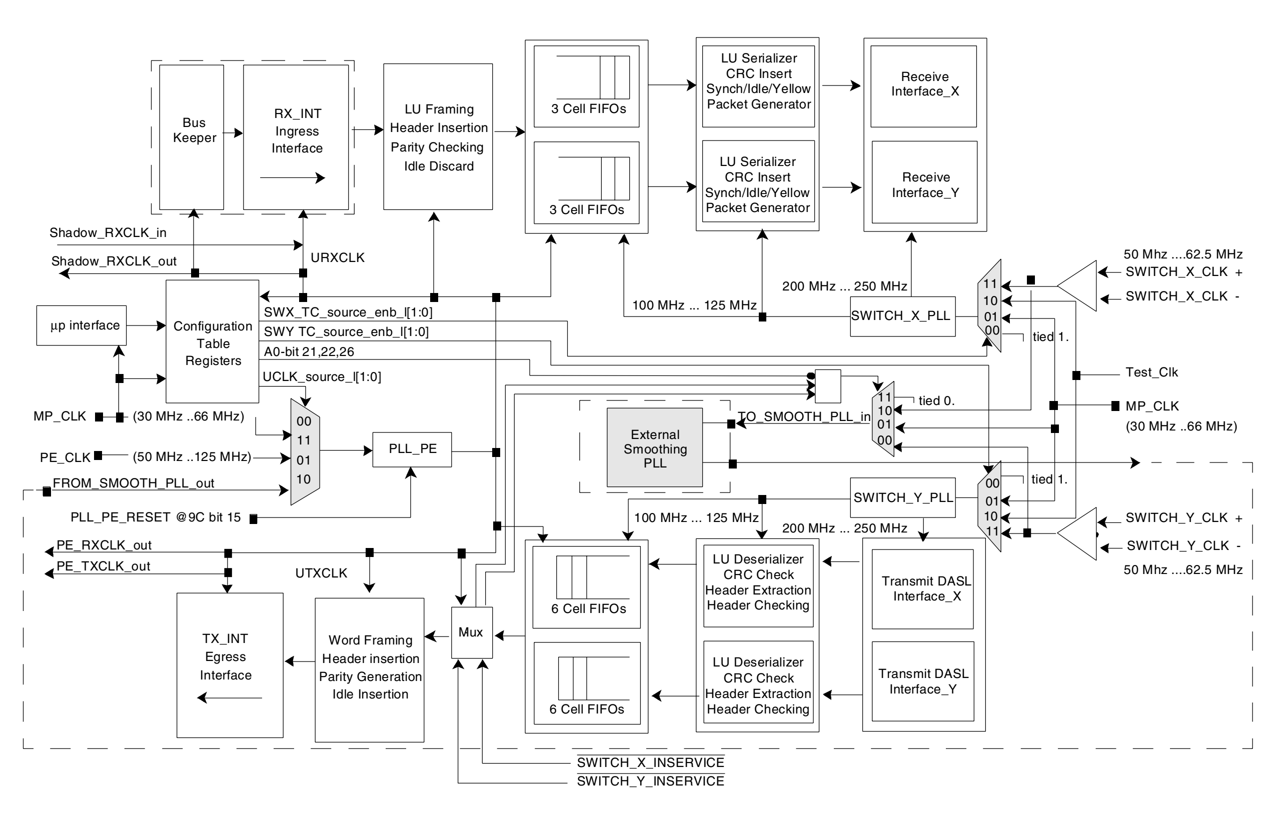


Table 11: Selecting the Signal That Appears on the TO_SMOOTH_PLL_IN Signal

SW_X_IN	SW_Y_IN	SelectX_Y @A0-22	Forcepath @A0-21	Smooth_select _enable_l@A0-26	To_Smooth_ PLL_In
X	X	0	1	1	SWITCH_X
X	X	1	1	1	SWITCH_Y
0	0	X	0	1	MP_CLK
0	1	X	0	1	SWITCH_Y
1	0	X	0	1	SWITCH_X
1	1	X	0	1	MP_CLK
X	X	X	X	0	0

Table 12: External Clocks Description

Clock Name	Speed	Description
SWITCH_X_CLK +/-	50 - 62.5 MHz	Differential input receiver clock lines. The associated PLL (SWITCH_X_PLL) delivers both 200 - 250 MHz (DASL_X250_CLK) and 100 - 125 MHz (DASL_X125_CLK) to clock the DASL_X and the path X glue logic. An additional input is provided to connect an external oscillator TEST_CLCK or the MP_CLK instead of the SWITCH_X_CLK on the PLL_X input. SWX_TC_source_enb_l[1:0] is located in the configuration table registers (@A0 bits 25 - 24).
SWITCH_Y_CLK +/-	50 - 62.5 MHz	Differential input receiver clock lines. The associated PLL (SWITCH_Y_PLL) delivers both 200 - 250 MHz (DASL_Y250_CLK) and 100 - 125 MHz (DASL_Y125_CLK) to clock the DASL_Y and the path Y glue logic. An additional input is provided to connect an external oscillator TEST_CLK or the MP_CLK instead of the SWITCH_Y_CLK on the PLL_Y input. SWY_TC_source_enb_l[1:0] is located in the configuration table registers (@A0 bits 15 - 14).
PE_CLK	50 and 125 MHz	Single-ended receiver clock line. This is the transfer/synchronization clock issued from the PE to synchronize transfers on RXDATA[31:0]. It is connected to the PE_PLL which delivers the URXCLK /UTXCLK for the ingress/egress interfaces. The PE_CLK input is controlled by PLL_PE_RESET (@9C bit 15) and is turned in BYPASS mode in case PLL_PE_RESET = ’1’. Programming PLL_PE_RESET at ’1’ disables the PE_PLL from delivering the necessary clocks for the ingress/egress interfaces.
MP_CLK	30 - 66 MHz	Single-ended receiver clock line. This is a free running clock which must be available at POR when the MP_CLK directly clocks the configuration table registers to provide correct register map initialization.
PE_RXCLK_out/ PE_TXCLK_out		Protocol engine clocks. They are single-ended output driver clock lines that allow the transfer and synchronization of the RXDATA[31:0] and the TXDATA[31:0] to and from the converter. They are derived from clock sources selected through the setting of register @A0, bits 21, 22, and 26 (To_smooth_PLL_In) and bits 30-31 (UCLK_Source_1).
Shadow_RXCLK_in	50 and 125 MHz	Single-ended receiver clock line. It is applied to the ingress interface by programming the RXDATA_KEEPER register @C0 bit 2 at '1'. When the RXDATA_KEEPER is equal to '0', Shadow_RxClk_Out is routed to the ingress PE interface (this setting is used for switch loopback application).
Shadow_RxClk_Out		A single-ended output driver clock line derived directly (not going through any clock tree) from PE PLL output. It connects to the Shadow_RxClk_In through either the PE chip or the board (via a delay) to provide the best sampling point of the received data from the protocol engine.

3.8.1 IBM Packet Routing Switch Serial Interface Converter Internal Clocks Description

The converter clock tree is articulated around three PLLs and the microprocessor (MP_CLK) clock. Seven clock domains exist to clock the converter. The clock generator acts as a programmable selector. UTXCLK and URXCLK internal clock trees are generated from several clock sources that are selected by programming the "UCLK_source_l [1:0]" and "XY_SW_source_enb_l [1:0]" in the configuration table registers.

3.8.2 IBM Packet Routing Switch Serial Interface Converter External Traffic:

External traffic from the converter and the IBM 28.4 G Packet Routing Switch (switch) is timed using the two edges of the clock issued from the SWITCH_X_PLL or the SWITCH_Y_PLL. SWITCH_X/Y_PLLs deliver two clocks which are 100 - 125 MHz (DASL_X125_CLK) and 200 - 250 MHz (DASL_X250_CLK) respectively. The X/Y transmit and the X/Y receive DASLs are clocked by DASL_X250_CLK and DASL_Y250_CLK respectively. Receive and transmit data is then input and output at a frequency of 400 - 500 MHz. At each occurrence of the DASL_X/Y250_CLK clock edge, transition data is sent to or received from the switch.

3.9 IBM Packet Routing Switch Serial Interface Converter RESET Scheme Description

The converter allows total and selective reset. PATHX and PATHY can be reset together or independently. The ingress and egress interfaces and their associated FIFOs are reset independently. External reset, programmable reset, and power-on-reset are implemented in the converter.

3.9.1 Reset Strategy

The converter is reset when power-on-reset (POR) is active. T/S drivers are in Hi-Z state and bidirectional I/Os are sourced to receiver mode. The µp_interface and configuration table register are reset, and then the PE interface, FIFOs and their associated logic, and PATH_X/Y are reset.

When RESET_X is active, PATH_X is reset. When RESET_Y is active, PATH_Y is reset. DASL _X/Y differential drivers are in Hi-Z state.

The µP_interface initiates the reset by a write access to the bit that needs to be reset. A second write access is necessary to restore the bit to its inactive position.

The following table shows the different reset cases:

Table 13: Register Reset Settings

Module	POR	@A0 bit 0	@A0 bit 1	via BIT Configuration
Configuration Table Registers	X			
MP_INTERFACE	X			
PE_PLL	X			@9C bit 15
UTOPIA-3 PE Interface	X			@A0 bit 8
LU_FRAMING	X			@A0 bit 8
PATH_X INGRESS FIFO	X	X		@08 bit 10
PATH_Y INGRESS FIFO	X		X	@28 bit 10
WORD_UNFRAMING	X			@A0 bit 8
SWITCH_X_PLL	X			@90 bit 15
PATH_X EGRESS FIFO	X	X		@08 bit 11
PATH_Y EGRESS FIFO	X		X	@28 bit 11
RESET PATH_X_SDC	X	X		@A0 bit 4
RESET PATH_Y_SDC	X		X	@A0 bit 5
SWITCH_Y_PLL	X			@94 bit 15
RESET X_DASL	X	X		@A0 bit 2
RESET Y_DASL	X	X		@A0 bit 3

3.9.2 Power-On-Reset (POR) Procedure

The POR action uses the external MP_CLK microprocessor clock to reset the chip. All drivers are set to high impedance during the POR. The register table controlling those output lines must be reconfigured after POR.

After POR action, the converter must be re-configured via the µP_interface. The PLLs are switched into BYPASS mode and neither PATH X nor PATH Y are configured.

POR external signal is an asynchronous signal. POR initiates when the POR signal input is going down. POR procedure stops when the POR signal input is held up. POR must be asserted for at least 10 MP_CLK cycles to insure proper chip reset.

The following table shows pin/pad TEST I/O initialization values (the values during and after reset):

Table 14: I/O Initialization Values

Pin Name	Value in System	Description
CE1_A	1	LSSD test A clock
CE1_B	1	LSSD test B clock
CE1_C1	1	LSSD test C clock
CE1_C2	1	LSSD test C clock
CE0_IO	0	Used to force the JTAG EXTEST operation
CEO_Scan	0	A and B clock gating, used also to Clock Splitter GATE input
CE0_TEST	0	Used to control Boundary Scan feature
TEST_B2	1	LSSD test B clock (if required)
TEST_C3	1	LSSD test C clock (if required)
DI1	1	Driver Inhibit (for non test I/O)
DI2	1	Driver Inhibit (for test I/O)
RI	1	Receiver Inhibit
TDI	1	JTAG serial input
TCK	1	JTAG clock
TMS	1	JTAG control signal
TRST	1	JTAG asynchronous reset

3.9.3 Path Reset

Individual PATH X and PATH Y resets are performed by addressing an appropriated bit in the COMMON_CONFIG_REGISTER @A0 (CCR). SWITCH_X_PLL or SWITCH_Y_PLL must run during the PATH X/Y reset action. PLLs are not affected by PATH X/Y resets, but the PLL must be reset when an external loopback is initiated with a new clock source (PLL needs a delay to lock). Once a reset is established, the software must reconfigure the CCR @A0 bits 0, 1, 2, 3 into the system mode.

Table 15: Path Resets

Reset Name	Impact	Reset
PATH X	No impact on PATH Y	Program CCR @A0 bit 0 at ‘1’ in the configuration table registers.
PATH Y	no impact on PATH X	Program CCR @A0 bit 1 at ‘1’ in the configuration table registers.
DASL_X	no impact on DASL_Y	Program CCR @A0 bit 2 at ‘1’ in the configuration table registers.
DASL_Y	no impact on DASL_X	Program CCR @A0 bit 3 at ‘1’ in the configuration table registers.

3.9.4 PLL Reset

Once a reset is established, the software must reconfigure the SWITCH_PLL X/Y bit or PLL_PE bit 15 to ‘0’ (system mode).

Table 16: System Mode PLL Resets

Reset Name	Impact	Reset
SWITCH_X_PLL	Exclusive - No impact on other PLLs	Program @90 SWITCH_PLL_X bit 15 at ‘1’ in the configuration table registers.
SWITCH_Y_PLL	Exclusive - No impact on other PLLs	Program @94 SWITCH_PLL_Y bit 15 at ‘1’ in the configuration table registers.
PLL_PE	Exclusive - No impact on other PLLs	Program @9C PLL_PE bit 15 at ‘1’ in the configuration table registers.

3.9.5 Ingress/Egress Interface Reset

Ingress and egress interfaces are reset on software request. An interface reset restores the internal scheduler to the idle phase. Transfer data is lost and internal registers are swapped to their reset position. Signals that drive the PE outputs are switched so to hold the PE interface drivers in inactive state.

Both ingress and egress PE interfaces are reset simultaneously by programming @A0 CONFIG_reg bit 8 at ‘1’ in the configuration table registers. Once the reset is established, the software must reconfigure the @A0 CONFIG_reg bit 8 at ‘0’ (system mode).

3.10 Microprocessor Interface Description

The IBM Packet Routing Switch Serial Interface Converter (the converter) chip is initialized and controlled via a processor interface which works on an 8-bit data bus and operates in two modes. The external input pin MP_BURST_SEL selects the operational mode (low - 8-bits mode and high - 32-bits mode):

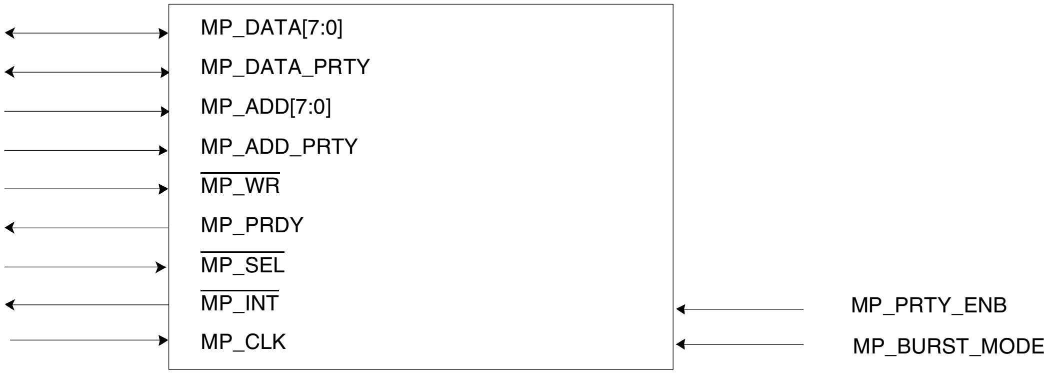
- 8-bit mode (byte mode): The converter registers are considered single 8-bits registers and are addressed via MP_ADD[7:0] address bus signals. Each register access is a single-beat access of one byte.
- 32-bit mode (burst mode): The converter registers are considered as single 32-bits registers and are addressed via MP_ADD[7:2] address bus signals. Each access is a burst access of four bytes. The burst order is the following: data bits [7:0], data bits [15:8], data bits [23:16], and data bits [31:24].

3.10.1 The microprocessor interface:

- Provides read/write access to all chip registers
- Provides DASL picocode downloading
- Provides error reporting
- Collects the converter interrupts and pass them to the attached processor.
- Monitors all interrupt signals generated by other converter functional blocks and, when one is asserted, latches and holds the value until the interrupt event register is read and reset.
- Provides the necessary handshake protocol to interface the attached processor, including address bus decoding, wait state insertion, data bus drivers control, and optional parity checking on both the data and address busses.

3.10.2 Processor Interface Lines

Figure 22: IBM Packet Routing Switch Serial Interface Converter Processor Interface Lines



3.10.3 Processor Interface I/O Lines Description

The processor interface is synchronized to the external processor clock. This clock operates at a different frequency than the switch fabric clock.

Pin Name	I/O	Width	Description
MP_DATA_[7:0]	BiDi	8 bits	bidirectional Data Bus - Hi-Z when no access being processed. MSB LSB 7 0
MP_DATA_PRTY	BiDI	1 bit	Data Byte ODD Parity Bit - Hi-Z when no access being processed. The converter checks during write operation and generates during read operation. Parity checker can be disabled.
MP_ADD[7:0]	In	8 bits	Address Bus MSB LSB 7 0
MP_ADD_PRTY	In	1 bit	Address Bus ODD Parity - converter checks during read and write access. Default is parity checker disabled.
$\overline{\text{MP\_WR}}$	In	1 bit	Read/Write line Low High Read Operation Write Operation
MP_PRDY	Out		Ready signal - Asserted High when data is valid on the bus (for read) or when data is written into converter (for write). Driven by converter. Hi-Z when converter not selected
$\overline{\text{MP\_SEL}}$	In		IBM Packet Routing Switch Serial Interface Converter Selected when LOW
MP_CLK	In		Processor Interface clock Fmax 66 MHz
$\overline{\text{MP\_INT}}$	Out		Converter Processor Interrupt. Asserted low when interruption is pending

3.10.4 32-Bit Mode Processor Interface Timing

Figure 23: Processor Read Access in 32-bit Burst Mode

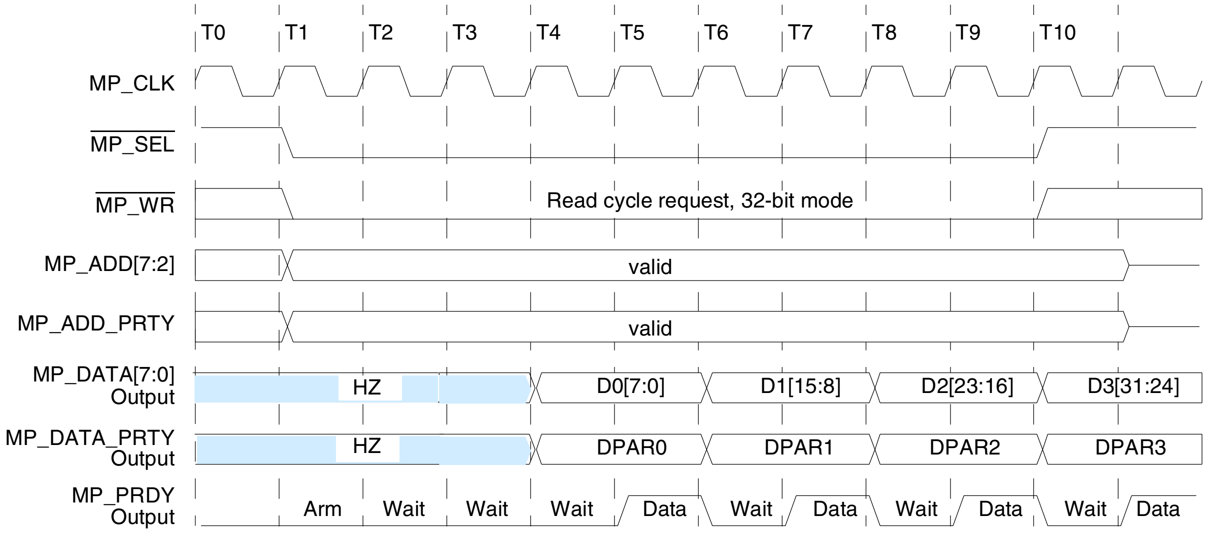
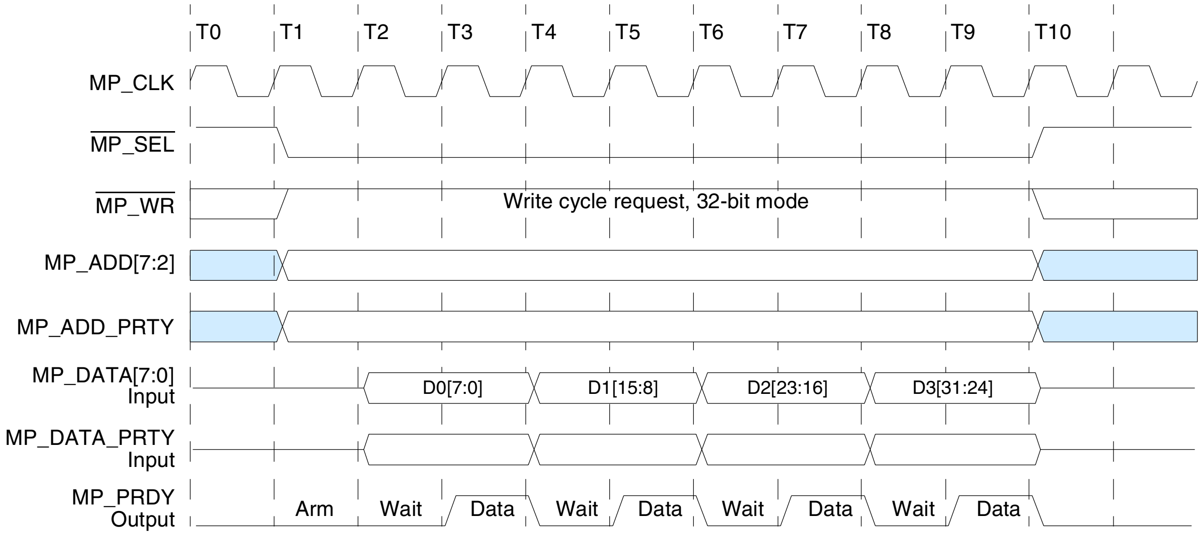


Figure 24: Processor Write Access in 32-bit Burst Mode



3.10.5 8-Bit Mode Processor Interface Timing

Figure 25: Processor Read Access in 8-bit Byte Mode

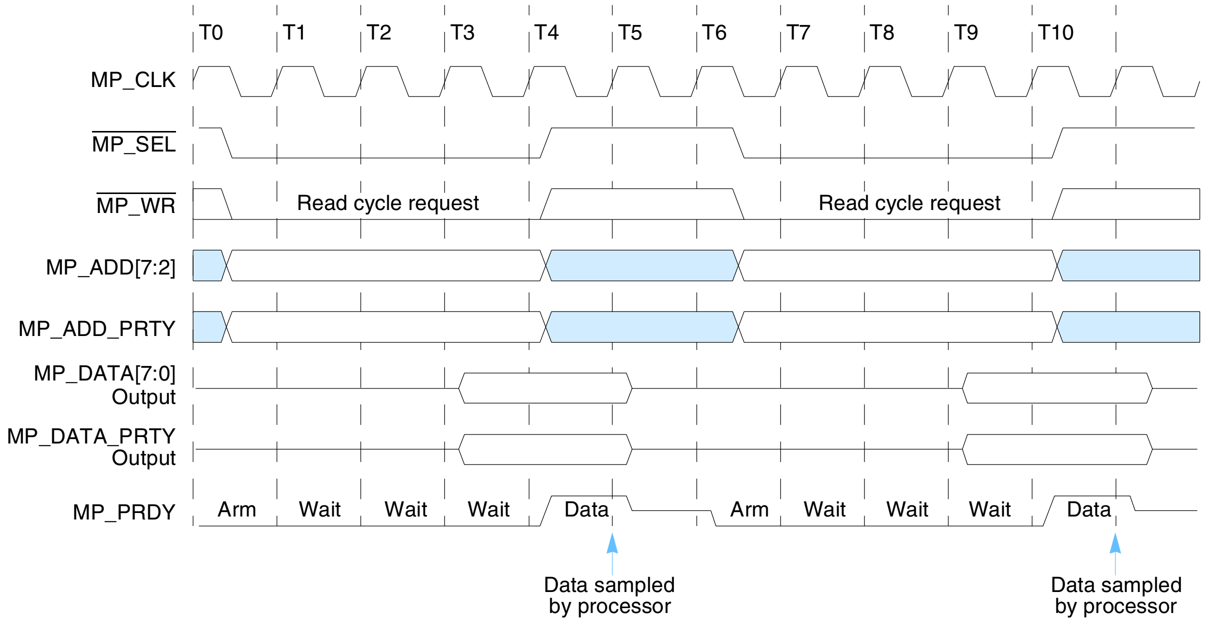
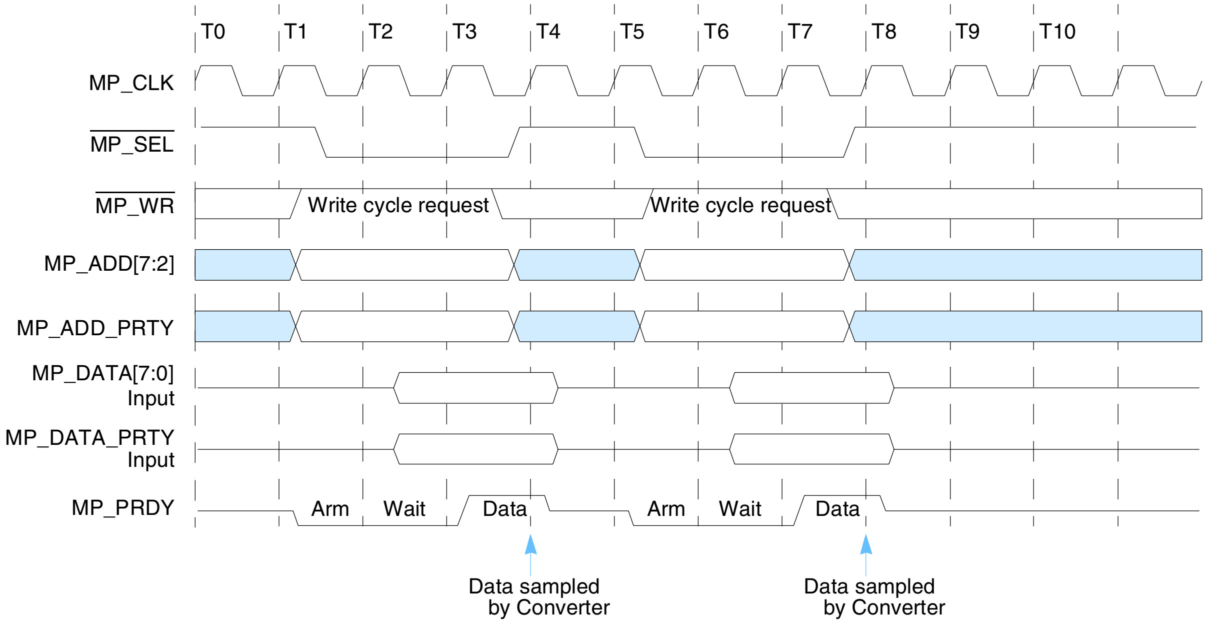


Figure 26: Processor Write Access in 8-bit Byte Mode



Note: MP_SEL can be active one more cycle without impact on the processor access, however the minimum deactivate time must remain two clock cycle times.

4. Converter Configuration Table Registers

This module contains all the configuration registers that the converter needs to operate.

- Registers are referenced from @00 up to @FF.
- There are two ways to access the configuration table registers:
  - During the POR procedure, some registers can be preset to start the system as predetermined.
  - Using the microprocessor interface, the configuration table register can be either READ or WRITE.
- Write access to a read-only register is not valid and does not change the held value of that register.
- All errors are logged (not first failure data capture).
- All bits in configuration table registers are active binary 1.
- The registers can be accessed either in 8-bit (byte) mode or in 32-bit (burst) mode through bursts of 4x 8-bits modes (for I960 Processors). In 32-bit mode, the least significant byte is sent first, then the second least significant byte, and so forth.
- The registers are based on little endian notation.

Figure 27: Register Mapping

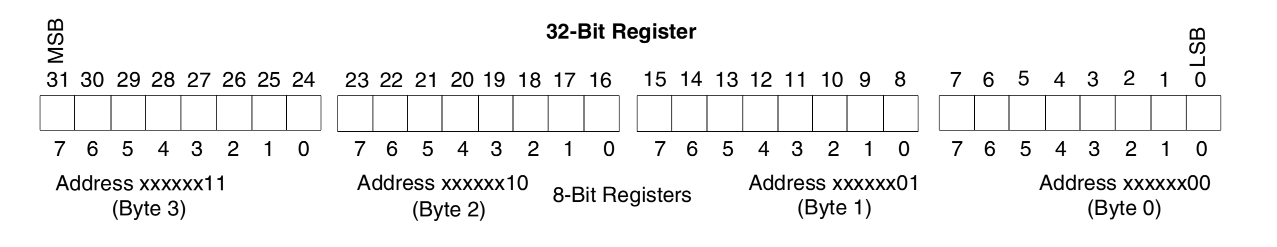


Figure 28: Register Addressing

7	6	5	4	3	2	1	0	Address Bits
0	0	0						Plane X Clock Domain
0	0	1						Plane Y Clock Domain
0	1	0						DASL Plane X Clock Domain
0	1	1						DASL Plane Y Clock Domain
1	0							Microprocessor Clock Domain
1	1							Protocol Engine Clock Domain
Register Selection Per Clock Domain						Byte Selection In a Register		

4.1 Error Detection, Reporting, and Interrupt Registers

The converter uses a common strategy for error detection, error reporting, and interrupt generation:

- Each error is detected by an individual checker (for instance parity checker)
- An error must be individually enabled in the CHECKER_ENABLE_REGISTER to be reported into EVENT_REGISTER
- An error must be individually enabled in the INTERRUPT_ENABLE register to generate an interrupt
- The INTERRUPT_REGISTER_INDIRECTION is the first register to read when an interrupt is raised. The interrupt cause can either be present in the register itself or be via an indirection to another EVENT_REGISTER
- For some specific errors (Parity and CRC), it is possible to discard or not discard the corresponding packet, depending on the setting of the corresponding configuration register

The interrupt routine is as follows:

1. The processor reads the INTERRUPT_REGISTER_INDIRECTION for the interrupt cause (either directly or indirectly via a second read into the EVENT_REGISTER flagged by INTERRUPT_REGISTER_INDIRECTION)
2. After processing the interrupt routine, the processor resets only the EVENT_REGISTER. The reset is under mask, so the whole byte of a register can be reset by writing x’FF’ inside, or it can be reset a single bit at a time by writing a ’1’ at the bit position in the byte to be reset. However two bits of a byte cannot be reset simultaneously (applicable only to Event registers @ 10, 30, 88)

4.1.1 Register Map

Register Name	Word Address	Byte Address
Setup1_X PATH	x‘00’	x‘00 to 03’
Setup 2_X PATH	x’04’	x’04 to 07’
Control_X PATH	x’08’	x’08 to 0B’
CRC_Error_Count_X	x’0C’	x’0C to 0F’
Event 1 _X	x’10’	x’10 to 1B’
Event 1 Checker Enable _X	x’14’	x’14 to 17’
Interrupt Enable _X	x’18’	x’18 to 1B’
Setup1_Y PATH	x’20’	x’20 to 23’
Setup2_Y PATH	x’24’	x’24 to 27’
Control_Y PATH	x’28’	x’28 to 2B’
CRC_Error_Count_Y	x’2C’	x’2C to 2F’
Event 1 _Y	x’30’	x’30 to 3B’
Event 1 Checker Enable _Y	x’34’	x’34 to 37’
Interrupt Enable _Y	x’38’	x’38 to 3B’
DASL M3 Picocode X	x’40’	x’40 to 43’
Shared DASL Controller: SDC_Debug_CNTL X	x’44’	x’44 to 47’
Shared DASL Controller: SDC_Debug_Data_In X	x’48’	x’48 to 4B’
Shared DASL Controller: SDC Debug_Data_Out X	x’4C’	x’4C to 4F’
Shared DASL Controller: SDC_Debug_Data_Address X	x’50’	’50 to 53’
SHared DASL Controller: SDC_ Status_Reg X	x’54’	’54 to 57’
DASL M3 Picocode Y	x’60’	x’60 to 63’
Shared DASL Controller: SDC_Debug_CNTL Y	x’64’	x’64 to 67’
Shared DASL Controller: SDC_Debug_Data_In Y	x’68’	x’68 to 6B’
Shared DASL Controller: SDC Debug_Data_Out Y	x’6C’	x’6C to 6F’
Shared DASL Controller: SDC_Debug_Data_Address Y	x’70’	’70 to 73’
Shared DASL Controller: SDC_ Status_Reg Y	x’74’	’74 to 77’
Event 2 Checker Enable _X and _Y	x’80’	x’80 to 83’
Event 2 Interrupt Enable_X and _Y	x’84’	x’84 to 87’
Event 2_X and _Y	x’88’	x’88 to 8B’
Test Status _X_Y	x’8C’	x’8C to 8F’
Switch X PLL	x’90’	x’90 to 93’
Switch Y PLL	x’94’	x’94 to 97’
Chip ID	x’98’	x’98 to 9B’
PE PLL Register	x’9C’	x’9C to 9F’
Common Control Register	x’A0’	x’A0 to A3’
Interrupt Register Indirection	x’A4’	x’A4 to A7’
Ingress_PE_Interface	x’C0’	x’C0 to C3’
Egress_PE_Interface	x’C4’	x’C4 to C7’
PE (Common)	x’C8’	x’C8 to CB’
PARITY_Error_count	x’CC’	x’CC to CF’

4.1.1.1 Setup 1_X PATH Register

Unused	XDASLConf1	XDASLConf0	Unused	XDiscPZHd	XColor	XbadPrty	XbadCRC	XGrantForce	XGrantDis	XPELoop
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path X)	All ’0’s
Address in Word Mode	x'00’
Address in Byte Mode	x'00 to 03’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description	Notes
31-24	7-0		Unused	
23-16	7-0	XDASLConf1	DASL configuration register (physical subport, = 32-bit port unique hardware version number, synchronization mode). Configure to x’07’	1
15-8	7-0	XDASLConf0	DASL configuration register (physical subport, unique hardware version number ‘10001’, synchronization mode ‘1’). Configure to x’89’	1
7	7		Unused	
6	6	XDiscPZHd	1 Do not discard packets with IBM 28.4 G Packet Routing Switch header parity in error on switch egress (EDI) 0 Discard packets	
5	5	XColor	1 Force red in IBM 28.4 G Packet Routing Switch Header for idle packets to switch core 0 Force blue idle packets at IDI	
4	4	XbadPrty	1 Force a bad parity on idle packets in the IBM 28.4 G Packet Routing Switch header to switch core 0 Normal operation	
3	3	XbadCRC	1 Force a bad Switch CRC insertion on Ingress side. 0 Normal operation	
2	2	XGrantForce	1 Force Output Queue and Memory Grant to 1(always granted) during PE loopback to bypass flow conrol 0 Normal operation (use flow control)	
1	1	XGrantDis	1 Force Output Queue and Memory Grant to 0 (no grant) during PE loopback to block the flow 0 Normal operation	
0	0	XPELoop	1 PE interface loopback via path X logic 0 No loopback	

1. See Appendix: *Appendix A: Data Aligned Serial Link (DASL)* on page 129

4.1.1.2 Setup 2_X Path Register

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

Reset Output Status (Power-on-Reset, Software Reset Path X)	'7807 3807'
Address in Word Mode	x'04'
Address in Byte Mode	x'04 to 07'
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31	7		Unused
30-24	6-0	XEgThFull	Egress FIFO almost full threshold (word)
23	7		Unused
22-16	6-0	XEgThEmpt	Egress FIFO almost empty threshold (word)
15-14	7-6		Unused
13-8	5-0	XInThFull	Ingress FIFO almost full threshold (word)
7-6	7-6		Unused
5-0	5-0	XInThEmpt	Ingress FIFO almost empty threshold (word)

The settings of Set Up Register 2 depend on the packet size, the PE clock, the switch clock, and the flow control mechanism in use.

Example 1: for a 64-bytes packet and a PE and switch using the same range clock frequency, the setting might be:

- Ingress buffer with out of band flow control ($\overline{\text{RXENB}}$)
  - Almost empty: x'10' for a 16 word-packet (packet of 64 bytes) or above to avoid underrun
  - Almost full: x'2C' or below to avoid overrun

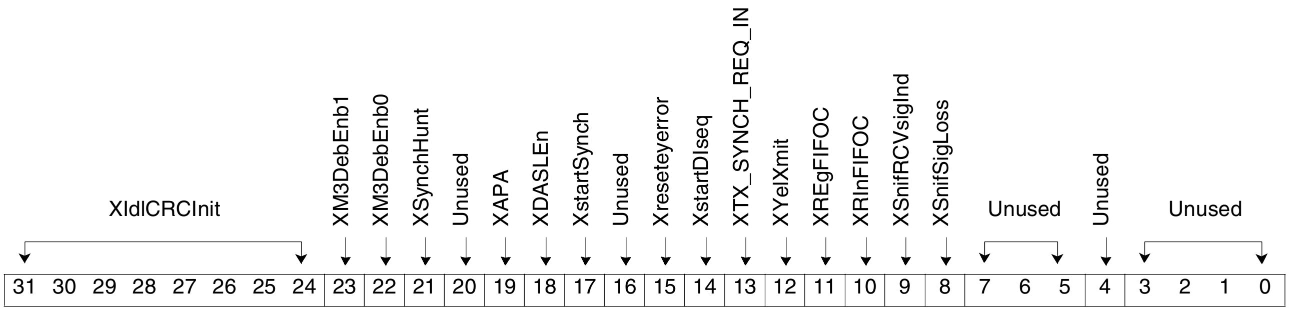
- Egress buffer
  - Almost empty: x'10' or above to avoid underrun
  - Almost full: x'3D' or below to avoid overrun

Example 2: for an 80-bytes packet and a protocol engine and switch clock at 110 MHz, the setting might be:

- Ingress buffer with out of band flow control (RXENB)
  - Almost empty: x'14' for a 20 word-packet (packet of 80 bytes) or above to avoid underrun
  - Almost full: x'28' or below to avoid overrun

- Egress buffer
  - Almost empty: x'14' or above to avoid underrun
  - Almost full: x'30' or below to avoid overrun

4.1.1.3 Control _X PATH Register



Reset Output Status (Power-on-Reset, Software Reset Path X)	‘0000 0C00’
Address in Word Mode	x'08’
Address in Byte Mode	x'08 to 0B’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	XIdlCRCInit	Initial FCS register value for Idle Packet CRC Computation Ingress Side (to match the LU size). LU LengthCRC Init Register in Hex 16 D0 17 DD 18 04 19 FA 20 07
23	7	XM3DebEnb1	1 Disable DASL debug bus 1 0 Enable DASL debug bus 1 (reserved)
22	6	XM3DebEnb0	1 Enable DASL debug bus 0. See Register @8C for the displayed signals. 0 Disable routing to debug bus.
21	5	XSynchHunt	1 Start receiver synchronization for bit / nible / byte / packet alignment 0 Stop receiver syncronization process
20	4		Unused
19	3	XAPA	1 Indicates on APAN_X I/O that the port is not synchronized 0 Written by the microprocessor to indicate the port is fully synchronized: mapped to APAN_X I/O
18	2	XDASLEn	1 Enable DASL port operation for transmitter and receiver 0 Disable DASL port operation
17	1	XstartSynch	1 Start transmission of synchronization packets on ingress side 0 Stop transmission of packets
16	0		Unused
15	7	Xreseteyerror	1 Reset DASL eye error detector (SDC port quality). See register @10 bit 6 for error. 0 Normal Operation
14	6	XstartDIseq	1 Starts DASL Interface Sequencer (on during normal operation’ 0 Stops DASL Interface Sequencer

Bits / Word	Bits / Bytes	Name	Description
13	5	XTX_SYNCH_REQ_IN	Must be written at ‘1’ for reset and then at ‘0’ to start DASL transmitter (emulates a pulse that resets DASL transmitter state machine).
12	4	XYelXmit	1 Tells IDI to send one yellow packet on the ingress path to the switch (must be reset (set to ‘0’) by software and rewritten afterwards to send the next yellow packet.) 0 Normal Operation
11	3	XREgFIFOC	1 Reset Egress FIFO Counter (for debug purpose) 0 Normal Operation
10	2	XRInFIFOC	1 Reset ingress FIFO Counter (used for debug purpose) 0 Normal Operation
9	1	XSnifRCVsigInd	Sniffer in real time for DASL Receive Data Indicator. Read the status (see reg @10 bit 13) (used for debug purpose)
8	0	XSnifSigLoss	Sniffer in real time for DASL loss of signal detection. Read the status. (see also reg @10 bit 5) (used for debug purpose)
7-5	7-5		Unused
4	4		Unused
3-0	3-0		Unused

4.1.1.4 X Plane Parity and CRC_Error_Count_X Registers

The microprocessor has the highest priority when writing into this register, so it may at any time overwrite the current value even if the counter is being incremented by the source of an error.

Note: All the counters do not wrap around.

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

Reset Output Status (Power-on-Reset, Software Reset Path X)	All ’0’s
Address in Word Mode	x'0C’
Address in Byte Mode	x'0C to 0F’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-12	7-4		Unused
11-8	3-0	XHPRY	IBM 28.4 Routing Packet Switch Header Parity error Count (on Egress from switch). Reset by writing 0.
7-4	7-4		Unused
3-0	3-0	XIErrCount	Idle Packet CRC error counter (on Egress from IBM 28.4 Routing Packet Switch). Reset by writing 0.

4.1.1.5 X Plane Events 1 Register (Event 1 _X)

Unused								Unused								Unused	XTX_SYNCH_REQ_OUT	XDARDInd	XUMTX	XUMRX	XM3 IRQ	XESBover	XISBover	XDASLPEr	XDPrtErr	XDsigLoss	XYellRcv	Unused	XGrtPty	XICRCErr	XHPty
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path X) All '0's

Address in Word Mode x'10'

Address in Byte Mode x'10 to 1B'

Access Type Read/Write

Bits/ Word	Bits / Bytes	Name	Description	Notes
31-24	7-0		Unused	
23-16	7-0		Unused	
15	7		Unused	
14	6	XTX_SYNCH_REQ_OUT	1 Transmit Synchronization request from the SDC to observe the status of DASL (when the picocode needs to reset DASL transmitter) 0 Normal mode	2
13	5	XDARDInd	1 DASL receiver is synchronized and ready to receive data. It is read by the microprocessor to assert APA @08 bit 19 to inform the switch that the adapter is synchronized and it can stop the transmission of the synchronization packets. Remains 'ON' if nothing abnormal occurs. 0 DASL receiver is not synchronized	1
12	4	XUMRX	1 At least one data packet has been sent to IDI to the switch (debug purpose) 0 No data packet has been forwarded by IDI to the switch	2
11	3	XUMTX	1 At least one data packet has been detected at EDI from the switch (debug purpose) 0 No data packet has been detected by EDI	2
10	2	XM3 IRQ	1 SDC Interruption: a parity error has occurred during the transfer between the SDC and the instruction or data store. Remains 'ON' as long as the condition is present and the DASL is not reset. 0 SDC operation is error free.	1
9	1	XESBover	1 Egress Buffer Overrun, occurs upon abnormal condition and requires chip re-initialization. 0 No buffer overrun (normal operation)	1, 3

1. I = an interrupt is raised because of error or status change.
2. S = status and is reported through polling.
3. In case of buffer overrun all packets are discarded.

Bits/ Word	Bits / Bytes	Name	Description	Notes
8	0	XISBover	1 Ingress Buffer Overrun, occurs upon abnormal condition and requires chip re-initialization. 0 No buffer overrun (normal operation)	1,3
7	7	XDASLPEr	1 DASL port error detected (returned when a port cannot be synchronized because the minimum eye opening criteria is not met or a data misalignment has been detected). Remains on as long as condition is present. 0 Port is synchronized and is not experiencing error.	1
6	6	XDPrtErr	1 DASL eye error detected. Remains 'ON' as long as the condition persists and register @08 bit 15 is not activated. 0 No eye error (receiver eye is wide open)	1
5	5	XDsigLoss	1 DASL loss of signal is detected. Remains 'ON' as long as the condition is present. 0 DASL receiver detects valid signal	1
4	4	XYellRcv	1 EDI detects a yellow packet coming from the switch (the bit is cleared by writing 0) 0 No yellow packet detected by EDI	1
3	3		Unused	
2	2	XGrtPty	1 Output Queue Grant flywheel counter is desynchronized 0 Output Queue Grant flywheel counter is synchronized	1
1	1	XICRCErr	1 LU CRC error is detected in switch idle packet (EDI) 0 No LU CRC is detected in EDI	1
0	0	XHPty	1 Parity error is detected on switch header on egress side (EDI) 0 No switch header parity error is detected	1

1. I = an interrupt is raised because of error or status change.
2. S = status and is reported through polling.
3. In case of buffer overrun all packets are discarded.

Figure 29: DI and Data Aligned Serial Link (DASL) Startup Sequence Path X

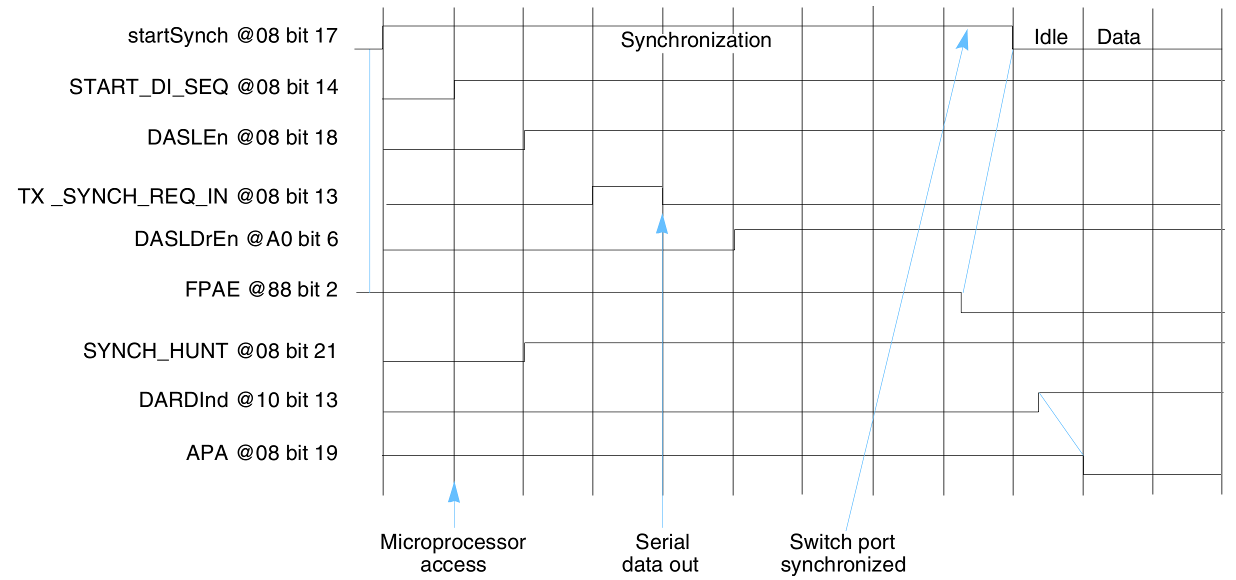
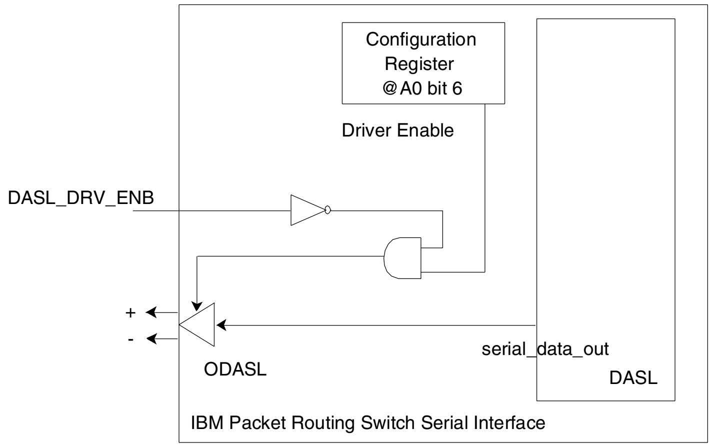


Figure 30: Enabling of Data Aligned Serial Link (DASL) Data Transmission and Reception Path X


Configuration Register @A0 bit 6
Driver Enable
DASL_DRV_ENB
+
-
ODASL
serial_data_out
DASL
IBM Packet Routing Switch Serial Interface

4.1.1.6 X Plane Event 1 Checker Enable Register (Event 1 Checker Enable_X)

Unused	Unused	Unused	XDARDIndEn	Unused	XESBoverEn	XISBoverEn	XDASLPErn	XDPrtErrEn	XDsiglossEn	XYellRcvEn	Unused	XGrtPtyEn	XICRCErrEn	XHPtyEn
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14	13	12 11 10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path X)	All ’0’s
Address in Word Mode	x'14’
Address in Byte Mode	x'14 to 17’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-14	7-6		Unused
13	5	XDARDIndEn	Enable DASL Receive Data Indicator
12-10	4-2		Unused
9	1	XESBoverEn	Enable Egress Buffer Overrun Checker
8	0	XISBoverEn	Enable Ingress Buffer Overrun Checker
7	7	XDASLPErn	Enable DASL Port in Error
6	6	XDPrtErrEn	Enable egress DASL Interface EYE Error Flag
5	5	XDsiglossEn	Enable DASL Loss of Signal Detection Flag
4	4	XYellRcvEn	Enable egress Yellow Packet Reception Checker
3	3		Unused
2	2	XGrtPtyEn	Enable egress Grant Priority Error Flag Checker
1	1	XICRCErrEn	Enable egress IBM 28.4 Routing Packet Switch Idle Packet CRC Error Checker
0	0	XHPtyEn	Enable egress IBM 28.4 Routing Packet Switch Header 0 parity error Flag Checker

Note: To enable the checker corresponding to an event, set ’1’ in the corresponding bit position.

4.1.1.7 Interrupt Enable _X Register

Unused								Unused								Unused		XDARDIndE	Unused		XM3Int	XESBovE	XISBovE	XDASLPEE	XDPrtErrE	XDsigLossE	XYellRcvE	Unused	XGrtPtyE	XICRCErrE	XHPtyE
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path X)	All ’0’s
Address in Word Mode	x'18’
Address in Byte Mode	x'18 to 1B’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-14	7-6		Unused
13	5	XDARDIndE	Enable DASL Receive Data Indicator Interrupt
12-11	4-3		Unused
10	2	XM3Int	Enable M3 interrupt
9	1	XESBovE	Enable egress Buffer Overrun Interrupt
8	0	XISBovE	Enable ingress Buffer Overrun Interrupt
7	7	XDASLPEE	Enable DASL Port in error Interrupt
6	6	XDPrtErrE	Enable egress DASL Interface EYE error Flag Interrupt
5	5	XDsigLossE	Enable DASL loss of signal detection Interrupt
4	4	XYellRcvE	Enable egress yellow packet reception Interrupt
3	3		Unused
2	2	XGrtPtyE	Enable egress Grant Priority Error Flag Interrupt
1	1	XICRCErrE	Enable egress IBM 28.4 Routing Packet Switch Idle Packet CRC Error Interrupt
0	0	XHPtyE	Enable egress IBM 28.4 Routing Packet Switch Header 0 parity error Flag Interrupt
Note: To enable the interrupt a ’1’ must be set in the corresponding bit position.			

4.1.1.8 Setup 1_Y PATH Register

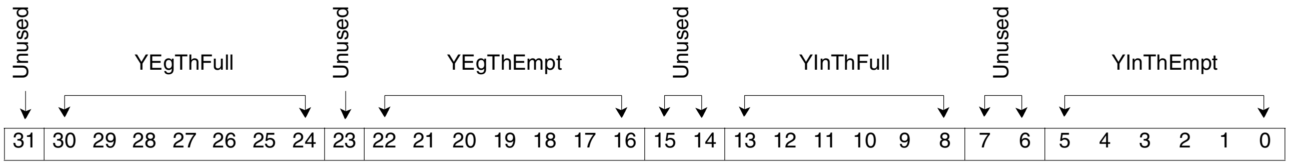
Unused								YDASLConf1								YDASLConf0								Unused	YDiscPZHd	YColor	YbadPrty	YbadCRC	YGrantForce	YGrantDis	YPELoop
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path Y)	All ’0’s
Address in Word Mode	x'20’
Address in Byte Mode	x'20 to 23’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description	Notes
31-24	7-0		Unused	
23-16	7-0	YDASLConf1	DASL configuration register (physical subport = 32-bit port, unique hardware version number, synchronization mode.) Configure to x’07’	1
15-8	7-0	YDASLConf0	DASL configuration register (physical subport, unique hardware version number = ‘10001’, synchronization mode = 1). Configure to x’89’	1
7	7		Unused	
6	6	YDiscPZHd	1 Do not discard packets with IBM 28.4 G Packet Routing Switch header parity in error on switch egress (EDI) 0 Discard packets	
5	5	YColor	1 Force red in IBM 28.4 G Packet Routing Switch Header for Idle packets to switch core 0 Force blue idle packets at IDI	
4	4	YbadPrty	1 Force a bad parity on idle packets in the IBM 28.4 G Packet Routing Switch header to switch core 0 Normal operation	
3	3	YbadCRC	1 Force a bad Switch-CRC insertion on Ingress side. 0 Normal operation	
2	2	YGrantForce	1 Force Output Queue and Memory Grant to 1 (always granted) during PE loopback to bypass flow control 0 Normal operation (use flow control)	
1	1	YGrantDis	1 Force Output Queue and Memory Grant to 0 (no grant) during PE Loopback to block the flow 0 Normal operation	
0	0	YPELoop	1 PE interface loopback via path Y logic 0 Normal operation	

1. See Appendix: *Appendix A: Data Aligned Serial Link (DASL)* on page 129

4.1.1.9 Setup 2_Y PATH Registers



Reset Output Status (Power-on-Reset, Software Reset Path Y)	‘7807 3807’
Address in Word Mode	x'24’
Address in Byte Mode	x'24 to 27’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31	7		Unused
30-24	6-0	YEgThFull	Egress FIFO Almost Full threshold (word)
23	7		Unused
22-16	6-0	YEgThEmpt	Egress FIFO almost empty threshold (word)
15-14	7-6		Unused
13-8	5-0	YInThFull	Ingress FIFO almost Full threshold (word)
7-6	7-6		Unused
5-0	5-0	YInThEmpt	Ingress FIFO almost empty threshold (word)

4.1.1.10 Control _Y PATH Registers

YIdlCRCInit	YM3DebEnb1	YM3DebEnb0	YSynchHunt	Unused	YAPA	YDASLEn	YstartSynch	Unused	Yreseteyerror	YstartDlseq	YTX_SYNCH_REQ_IN	YYelXmit	YREgFIFOC	YRInFIFOC	YSnifRCVsigInd	YSnifSigLoss	Unused	Unused	Unused
31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7 6 5	4	3 2 1 0

Reset Output Status (Power-on-Reset, Software Reset Path Y) ‘0000 0C00’

Address in Word Mode x'28’

Address in Byte Mode x'28 to 2B’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	YIdlCRCInit	Ingress: initial FCS register value for Idle Packet CRC Computation LU LengthCRC Init Register in Hex 16 D0 17 DD 18 04 19 FA 20 07
23	7	YM3DebEnb1	1 Disable DASL debug bus 1 0 Enable DASL debug bus 1 (reserved)
22	6	YM3DebEnb0	1 Enable DASL debug bus 0 (see register @8C for displayed signals) 0 Disable the routing to the debug bus
21	5	YSynchHunt	1 Starts receiver synchronization for bit / nible / byte / packet alignment. 0 Stop the receiver synchronization process
20	4		Unused
19	3	YAPA	1 Indicates on APAN_Y I/O that the port is not synchronized 0 Written by the microprocessor to indicate the port is fully synchronized: mapped to APA_Y I/O)
18	2	YDASLEn	1 Enable DASL port operation for transmitter and receiver 0 Disable DASL port operation
17	1	YstartSynch	1 Initiate transmission of ingress synchronization packets 0 Stop transmission of ingress synchronization packets
16	0		Unused
15	7	Yreseteyerror	1 Resets DASL eye error detector. See register @30 bit 6 for the error 0 Normal operation
14	6	YstartDlseq	1 Start DASL Interface Sequencer (ON during normal operation) 0 Stop DASL Interface Sequencer
13	5	YTX_SYNCH_REQ_IN	This bit must be written at ‘1’ for reset and then at ‘0’ to start DASL transmitter (to emulate a pulse that will reset DASL transmitter state machine)

Bits / Word	Bits / Bytes	Name	Description
12	4	YYelXmit	1 Tells IDI to send one yellow packet on the ingress path to the switch (must be reset (set to '0') by software and rewritten afterwards to send the next yellow packet.) 0 Normal operation
11	3	YREgFIFOC	1 Reset Egress FIFO Counter (for debug purpose) 0 Normal operation
10	2	YRInFIFOC	1 Reset ingress FIFO Counter (for debug purpose) 0 Normal operation
9	1	YSnifRCVsigInd	Sniffer in real time for DASL Receive Data Indicator. Read the status (see reg @30 bit 13). Used for debug purpose
8	0	YSnifSigLoss	Sniffer in real time for DASL loss of signal detection. Read the status (see also reg @30 bit 5). Used for debug purpose
7-5	7-5		Unused
4	4		Unused
3-0	3-0		Unused

Figure 31: DI and Data Aligned Serial Link (DASL) Startup Sequence Path Y

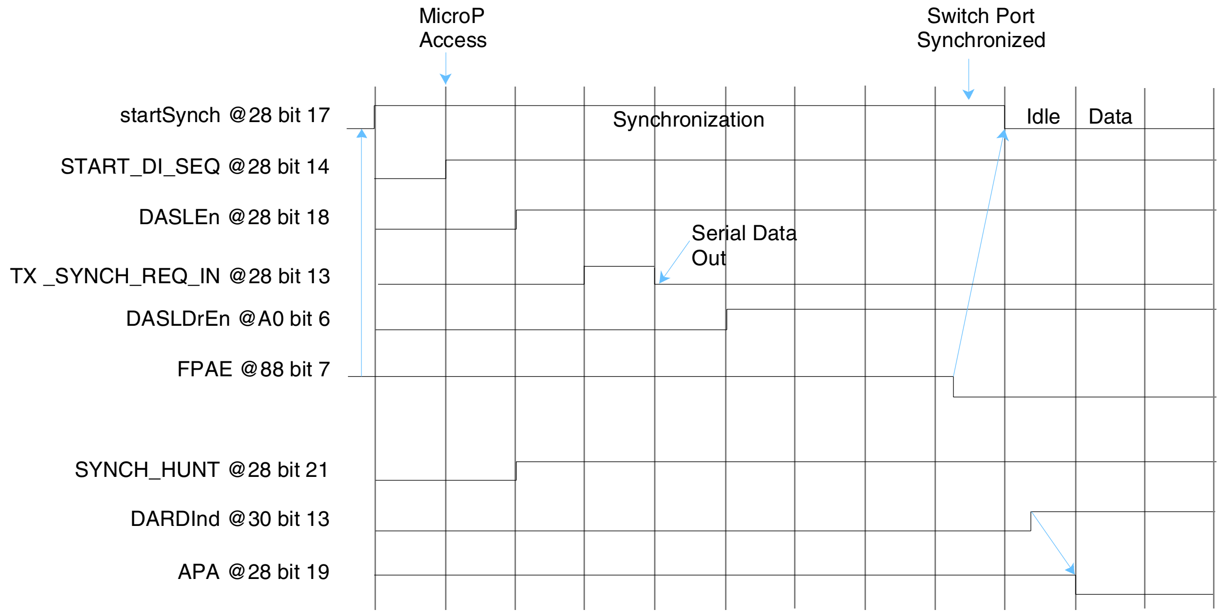
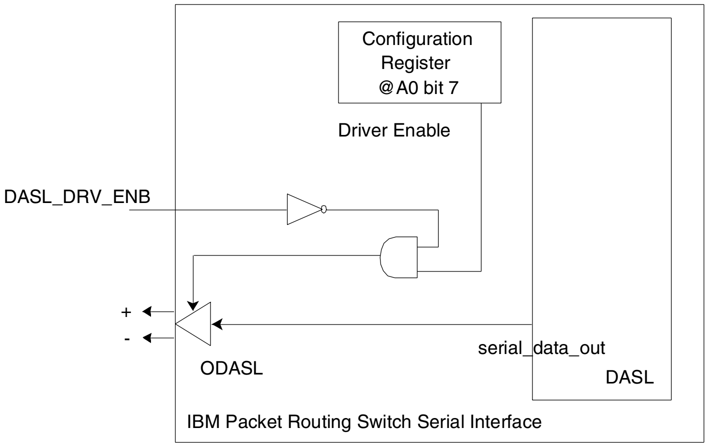


Figure 32: Enabling of Data Aligned Serial Link (DASL) Data Transmission and Reception Path Y



4.1.1.11 Y Plane Parity and CRC_Error_Count_Y Registers

The microprocessor has the highest priority when writing into this register, so it may at any time overwrite the current value even if the counter is being incremented by the source of an error.

Note: All the counters do not wrap around.

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

Reset Output Status (Power-on-Reset, Software Reset Path Y)	All '0's
Address in Word Mode	x'2C'
Address in Byte Mode	x'2C to 2F'
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-12	7-4		Unused
11-8	3-0	YHPRY	IBM 28.4 Routing Packet Switch Header Parity error count (on Egress from switch) 0 Reset
7-4	7-4		
3-0	3-0	YIErrCount	Idle Packet CRC error counter (on Egress from Switch) 0 Reset

4.1.1.12 Y Plane Events 1 Register (Event 1 _Y)

Unused	Unused	Unused	YTX_SYNCH_REQ_OUT	YDARDInd	YUMTX	YUMRX	YM3 IRQ	YESBover	YISBover	YDASLPEr	YDPrtErr	YDsigLoss	YYellRcv	Unused	YGrtPty	YICRCErr	YHPty
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path Y)	All ’0’s
Address in Word Mode	x'30’
Address in Byte Mode	x'30 to 3B’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description	Notes
31-24	7-0		Unused	
23-16	7-0		Unused	
15	7		Unused	
14	6	YTX_SYNCH_REQ_OUT	1 Transmit Synchronization request from the SDC to observe the status of DASL (when the picocode needs to reset DASL transmitter) 0 Normal mode	2
13	5	YDARDInd	1 DASL receiver is synchronized and ready to receive data. It is read by the microprocessor to assert APA @30 bit 19 to inform the switch that the adapter is synchronized and it can stop the transmission of the synchronization packets. Remains ‘ON’ if nothing abnormal occurs. 0 DASL receiver is not synchronized	1
12	4	YUMRX	1 At least one data packet has been sent by IDI to the switch (debug purpose) 0 No data packet has been forwarded by IDI to the switch	2
11	3	YUMTX	1 At least one data packet has been detected at EDI from the switch (debug purpose) 0 No data packet has been detected by EDI	2
10	2	YM3 IRQ	1 SDC Interruption: a parity error occurred during the transfer between the SDC and the instruction or data store. Remains ‘ON’ as long as the condition is present and that the DASL is not reset. 0 SDC operation is error free	1
9	1	YESBover	1 Egress Buffer Overrun, occurs upon abnormal condition and requires chip re-initialization 0 No buffer overrun (normal operation)	1,3

1. I = an interrupt is raised because of error or status change.
2. S = status and is reported through polling.
3. In case of buffer overrun all packets are discarded.

Bits / Word	Bits / Bytes	Name	Description	Notes
8	0	YISBover	1 Ingress Buffer Overrun, occurs upon abnormal condition and requires chip re-initialization. 0 No buffer overrun (normal operation)	1, 3
7	7	YDASLPEr	1 DASL port error detected (returned when a port cannot be synchronized because the minimum eye opening criteria is not met or a data misalignment has been detected). Remains 'ON' as long as condition is present. 0 Port is synchronized (not experiencing any error)	1
6	6	YDPrtErr	1 DASL eye error detected. Remains 'ON' as long as the condition persists and that register @28 bit 15 is not activated. 0 No eye error (receiver is wide open)	1
5	5	YDsigLoss	1 DASL loss of signal detected. Remains 'ON' as long as the condition is present. 0 DASL receiver detects valid signal	1
4	4	YYellRcv	1 EDI detects yellow packet coming from the switch 0 No yellow packet detected by EDI	1
3	3		Unused	
2	2	YGrtPty	1 Output Queue Grant flywheel counter is desynchronized 0 Output Queue Grant flywheel counter is synchronized	1
1	1	YICRCErr	1 LU CRC detected in switch idle packet (EDI) 0 No LU CRC detected in EDI	1
0	0	YHPty	1 Parity error detected on switch header on egress side (EDI) 0 No switch header parity error is detected	1

1. I = an interrupt is raised because of error or status change.
2. S = status and is reported through polling.
3. In case of buffer overrun all packets are discarded.

4.1.1.13 Y Plane Event 1 Checker Enable Register (Event 1 Checker Enable _Y)

Unused								Unused								Unused		YDARDIndEn	Unused			YESBoverEn	YISBoverEn	YDASLPErn	YDPrtErrEn	YDsiglossEn	YYellRcvEn	Unused	YGrtPtyEn	YICRCErrEn	YHPtyEn
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path Y) All ’0’s

Address in Word Mode x'34’

Address in Byte Mode x'34 to 37’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-14	7-6		Unused
13	5	YDARDIndEn	Enable DASL Receive Data Indicator
12-10	4-2		Unused
9	1	YESBoverEn	Enable egress Buffer Overrun Checker
8	0	YISBoverEn	Enable ingress Buffer Overrun Checker
7	7	YDASLPErn	Enable DASL Port in error
6	6	YDPrtErrEn	Enable egress DASL Interface EYE error Flag
5	5	YDsiglossEn	Enable DASL loss of signal detection Flag
4	4	YYellRcvEn	Enable egress yellow packet reception Checker
3	3		Unused
2	2	YGrtPtyEn	Enable egress Grant Priority Error Flag Checker
1	1	YICRCErrEn	Enable egress IBM 28.4 Routing Packet Switch Idle Packet CRC Error Checker
0	0	YHPtyEn	Enable egress IBM 28.4 Routing Packet Switch Header 0 parity error Flag Checker

Note: To enable the checker corresponding to an event, set ’1’ in the corresponding bit position.

4.1.1.14 Interrupt Enable _Y Register

Unused								Unused								Unused		YDARDIndE	Unused		YM3Int	YESBovE	YISBovE	YDASLPEE	YDPrtErrE	YDsigLossE	YYellRcvE	Unused	YGrtPtyE	YICRCErrE	YHPtyE
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, Software Reset Path Y) All ’0’s

Address in Word Mode x'38’

Address in Byte Mode x'38 to 3B’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-14	7-6		Unused
13	5	YDARDIndE	Enable DASL Receive Data Indicator Interrupt
12-11	4-3		Unused
10	2	YM3Int	Enable M3 interrupt
9	1	YESBovE	Enable Egress Buffer Overrun Interrupt
8	0	YISBovE	Enable Ingress Buffer Overrun Interrupt
7	7	YDASLPEE	Enable DASL Port in Error Interrupt
6	6	YDPrtErrE	Enable Egress DASL Interface EYE error Flag Interrupt
5	5	YDsigLossE	Enable DASL Loss of Signal Detection Interrupt
4	4	YYellRcvE	Enable Egress Yellow Packet Reception Interrupt
3	3		Unused
2	2	YGrtPtyE	Enable Egress Grant Priority Error Flag Interrupt
1	1	YICRCErrE	Enable Egress IBM 28.4 Routing Packet Switch Idle Packet CRC Error Interrupt
0	0	YHPtyE	Enable Egress IBM 28.4 Routing Packet Switch Header 0 Parity Error Flag Interrupt

Note: To enable the interrupt a ‘1’ must be set in the corresponding bit position.

4.1.1.15 DASL M3 Picocode X Register

XRead	XWrite	Xcomplete	Unused		Xaddress			Xaddress								XData								XData							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset) All ’0’s

Address in Word Mode x'40’

Address in Byte Mode x'40 to 43’

Access Type Read/Write

Bits/Word	Bits/Bytes	Name	Description
31	7	XRead	Read Enable. Does not reset by itself
30	6	XWrite	Write enable
29	5	Xcomplete	Read instruction complete flag (acknowledge). must be reset after completion
28-27	4-3		Unused
26-24	2-0	Xaddress	The following table gives the mapping between the eleven bits in this register, the addresses in the code and the bits within the DASL hardware Bit ordering for each entity. MSB LSB DASL 0 10 Code 10 0 Register 26 16
23-16	7-0	Xaddress	
15-8	7-0	XData	M3 read/write Data
7-0	7-0	XData	M3 read/write Data

Read and Write Operations

The write operation is performed as follows:

1. Set data byte in @ 40
2. Set data byte in @ 41
3. Set address in @ 42
4. Set the remaining bits of the address and bit 30 in @ 43 to issue the write operation
5. Reset bit 30 in @ 43

Repeat the above operational sequence as long as there is data to be loaded in the DASL. Bit 30 does not need to be reset between two successive write operations. For example, in the sdc.h file provided by IBM, the syntax in cmd file: WRITE (ADDRESS >= x’40’ , DATA >= x’4000900F’); where ‘4000’ means write enable and 000 is the start address) is Start Address and ‘900F’ is Data.

The read operation is performed as follows:

1. Set address byte in @42
2. Set the remaining address byte and bit 31 in @43
3. Poll @43 bit 29 for read completion. Then the data is valid and can be read from @40 and 41
4. Reset read enable bit 31 in @43 and read complete bit by writing '0' in bits 29 and 31

4.1.1.16 SDC Controller X Register (SDC_Debug_CNTL X)

XSDC_D_Ctl3								XSDC_D_Ctl2								XSDC_D_Ctl1								XSDC_D_Ctl0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset)	All ’0’s
Address in Word Mode	x'44’
Address in Byte Mode	x'44 to 47’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	XSDC_D_Ctl3	Debug Bus Control
23-16	7-0	XSDC_D_Ctl2	Debug Bus Control
15-8	7-0	XSDC_D_Ctl1	Debug Bus Control
7-0	7-0	XSDC_D_Ctl0	Debug Bus Control

4.1.1.17 SDC Data X in Bus Register (SDC_Debug_Data_In X)

XSDC_D_DataIn3	XSDC_D_DataIn2	XSDC_D_DataIn1	XSDC_D_DataIn0
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset) All ’0’s

Address in Word Mode x'48’

Address in Byte Mode x'48 to 4B’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	XSDC_D_DataIn3	Debug Bus data In
23-16	7-0	XSDC_D_DataIn2	Debug Bus data In
15-8	7-0	XSDC_D_DataIn1	Debug Bus data In
7-0	7-0	XSDC_D_DataIn0	Debug Bus data In

4.1.1.18 SDC Data X Out Bus Register (SDC_Data X_Out Bus)

XSDC_D_DataOUt3								XSDC_D_DataOut2								XSDC_D_DataOut1								XSDC_D_dataOut0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset)	All ’0’s
Address in Word Mode	x'4C’
Address in Byte Mode	x'4C to 4F’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	XSDC_D_DataOUt3	Debug Bus data Out
23-16	7-0	XSDC_D_DataOut2	Debug Bus data Out
15-8	7-0	XSDC_D_DataOut1	Debug Bus data Out
7-0	7-0	XSDC_D_dataOut0	Debug Bus data Out

4.1.1.19 SDC Address X Bus Register (SDC_Debug_Data_Address X)

Unused								Unused								XSDC_Add1								XSDC_Add0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset)	All ’0’s
Address in Word Mode	x'50’
Address in Byte Mode	x'50 to 53’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-8	7-0	XSDC_Add1	Debug Bus Address
7-0	7-0	XSDC_Add0	Debug Bus Address

4.1.1.20 SDC Status X Register (SDC_Status_Reg X)

<table>
<tr><td colspan="8">XSDC_D_St3</td><td colspan="8">XSDC_D_St2</td><td colspan="8">XSDC_D_St1</td><td colspan="8">XSDC_D_St0</td></tr>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

Reset Output Status (Power-on-Reset, DASL path X reset, SDC path X reset)	All ’0’s
Address in Word Mode	x'54’
Address in Byte Mode	x'54 to 57’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	XSDC_D_St3	Debug Bus Status
23-16	7-0	XSDC_D_St2	Debug Bus Status
15-8	7-0	XSDC_D_St1	Debug Bus Status
7-0	7-0	XSDC_D_St0	Debug Bus Status

4.1.1.21 DASL M3 Picocode Y Register

YRead	YWrite	Ycomplete	Unused	Yaddress	Yaddress	YData	YData
31	30	29	28 27	26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0

Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path yreset)	All ’0’s
Address in Word Mode	x'60’
Address in Byte Mode	x'60 to 63’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31	7	YRead	Read Enable. Does not reset by itself
30	6	YWrite	Write enable
29	5	Ycomplete	Read instruction complete flag. must be reset after completion
28-27	4		Unused
26-24	3-0	Yaddress	The following table gives the mapping between the eleven bits in this register, the addresses in the code and the bits within the DASL hardware Bit ordering for each entity. MSB LSB DASL 0 10 Code 10 0 Register26 16
23-16	7-0	Yaddress	
15-8	7-0	YData	M3 read/write Data.
7-0	7-0	YData	M3 read/write Data.

4.1.1.22 SDC Controller Y Register (SDC_Debug_CNTL Y)

YSDC_D_Ctl3								YSDC_D_Ctl2								YSDC_D_Ctl1								YSDC_D_Ctl0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path Y reset) All ’0’s

Address in Word Mode x'64’

Address in Byte Mode x'64 to 67’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	YSDC_D_Ctl3	Debug Bus Control
23-16	7-0	YSDC_D_Ctl2	Debug Bus Control
15-8	7-0	YSDC_D_Ctl1	Debug Bus Control
7-0	7-0	YSDC_D_Ctl0	Debug Bus Control

4.1.1.23 SDC Y Data In Bus Register (SDC_Debug_Data_In Y)

YSDC_D_DataIn3								YSDC_D_DataIn2								YSDC_D_DataIn1								YSDC_D_DataIn0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path Y reset) All ’0’s

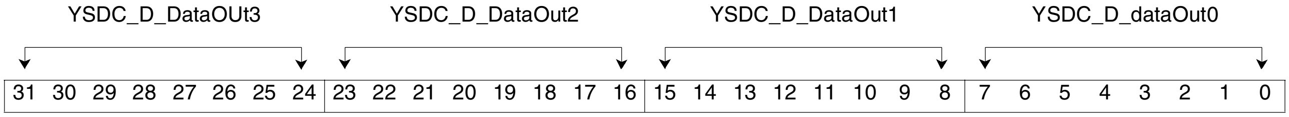
Address in Word Mode x'68’

Address in Byte Mode x'68 to 6B’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	YSDC_D_DataIn3	Debug Bus data In
23-16	7-0	YSDC_D_DataIn2	Debug Bus data In
15-8	7-0	YSDC_D_DataIn1	Debug Bus data In
7-0	7-0	YSDC_D_DataIn0	Debug Bus data In

4.1.1.24 SDC Data Y Out Bus Register (SDC_Data Y_Out Bus)



Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path Y reset) All ’0’s

Address in Word Mode x'6C’

Address in Byte Mode x'6C to 6F’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0	YSDC_D_DataOUt3	Debug Bus data Out
23-16	7-0	YSDC_D_DataOut2	Debug Bus data Out
8-15	7-0	YSDC_D_DataOut1	Debug Bus data Out
7-0	7-0	YSDC_D_dataOut0	Debug Bus data Out

4.1.1.25 SDC Address Y Bus Register (SDC_Debug_Data_Address Y)

Unused	Unused	YSDC_Add1	YSDC_Add0
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0

Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path Y reset) All ’0’s

Address in Word Mode x'70’

Address in Byte Mode x'70 to 73’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
8-15	7-0	YSDC_Add1	Debug Bus Address
7-0	7-0	YSDC_Add0	Debug Bus Address

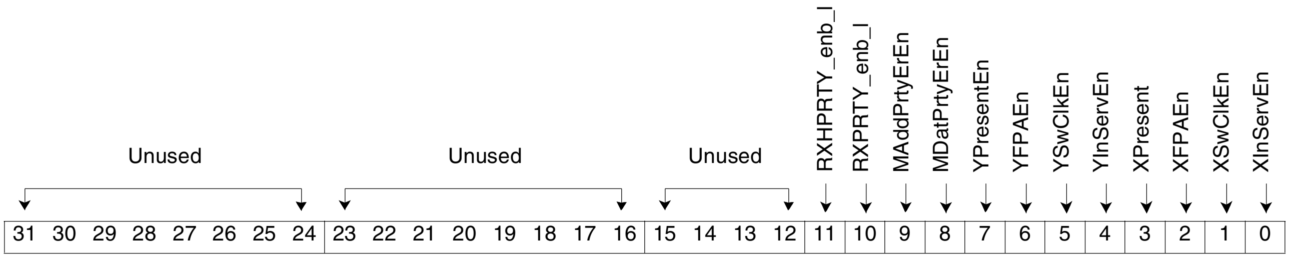
4.1.1.26 SDC Status Y Register (SDC_Status_Reg Y)

YSDC_D_St3								YSDC_D_St2								YSDC_D_St1								YSDC_D_St0							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, DASL path Y reset, SDC path Y reset)	All ’0’s
Address in Word Mode	x'74’
Address in Byte Mode	x'74 to 77’
Access Type	Read/Write

Bits/ Word	Bits/ Bytes	Name	Description
31-24	7-0	YSDC_D_St3	Debug Bus Status
23-16	7-0	YSDC_D_St2	Debug Bus Status
8-15	7-0	YSDC_D_St1	Debug Bus Status
7-0	7-0	YSDC_D_St0	Debug Bus Status

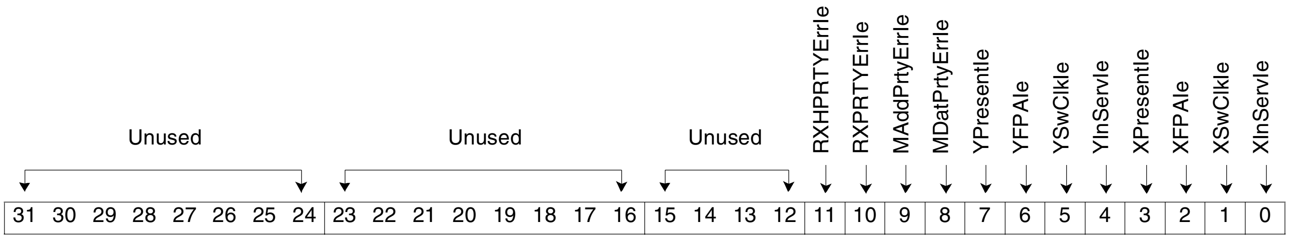
4.1.1.27 Event 2 Checker Enable_X and _Y Registers



Reset Output Status (Power-on-Reset, PE Reset)	All ’0’s
Address in Word Mode	x'80’
Address in Byte Mode	x'80 to 83’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-12	7-4		Unused
11	3	RXHPRTY_enb_l	IBM 28.4 G Packet Routing Switch Header Parity error checker enable
10	2	RXPRTY_enb_l	Enable ingress PE bus parity error checker
9	1	MAddPrtyErEn	Enable processor address parity error detection
8	0	MDatPrtyErEn	Enable processor data parity error detection
7	7	YPresentEn	Enable Y path Fabric present and fully inserted
6	6	YFPAEn	Enable Y path Fabric Port Available checker
5	5	YSwClkEn	Enable Switch Clock Y Missing checker
4	4	YInServEn	Enable Switch Board Y in service (Active/ready).
3	3	XPresent	Enable X path Fabric present and fully inserted
2	2	XFPAEn	Enable X path Fabric Port Available checker (FPA)
1	1	XSwClkEn	Enable Switch Clock X missing checker
0	0	XInServEn	Enable Switch Board X in service (Active/ready).

4.1.1.28 Event 2 µP Interrupt Enable_X and _Y Registers



Reset Output Status (Power-on-Reset, PE Reset)	All ’0’s
Address in Word Mode	x'84’
Address in Byte Mode	x'84 to 87’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
16-23	7-0		Unused
12-15	7-4		Unused
11	3	RXHPRTYErrle	Enable IBM 28.4 G Packet Routing Switch Header Parity Error Interrupt
10	2	RXPRTYErrle	Enable Ingress PE Bus Parity Error Interrupt
9	1	MAddPrtyErrle	Processor Address Parity Error Interrupt
8	0	MDatPrtyErrle	Processor Data Parity Error Interrupt
7	7	YPresentle	Enable Y Path Fabric Not Present or Not Fully Inserted Interrupt
6	6	YFPAle	Enable Y Path Fabric Port Not Available Checker (FPA)
5	5	YSwClkle	Enable Switch Clock Y Missing Interrupt
4	4	YInServle	Enable Switch Board Y in Service (Active/ready) interrupt
3	3	XPresentle	Enable X path Fabric Not Present or Not Fully Inserted Interrupt
2	2	XFPAle	Enable X path Fabric Port Available Interrupt (FPA)
1	0	XSwClkle	Enable Switch Clock X Missing Interrupt
0	0	XInServle	Enable Switch Board X in Service (Active/ready) Interrupt

4.1.1.29 Event 2 _X and _Y Registers

Some of these register bits directly reflect the status of some chip input lines, so their value depends on those lines.

Unused								Unused								Unused				IHPRTY_Err	IPERTY_Err	MAddPrtyErr	MDatPrtyErr	YPresent	YFPAE	YSwClkE	YInServE	XPresent	XFPAE	XSwClkE	XInServE
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, PE Reset)	All ’0’s
Address in Word Mode	x'88’
Address in Byte Mode	x'88 to 8B’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-12	7-4		Unused
11	3	IHPRTY_Err	1 IBM 28.4 G Packet Routing Switch Header Parity error checker (counter in @CC) 0 Normal Operation
10	2	IPERTY_Err	1 Ingress PE bus parity error (see register @CC for counter) 0 Normal Operation
9	1	MAddPrtyErr	1 Processor address parity error 0 Normal Operation
8	0	MDatPrtyErr	1 Processor data parity error 0 Normal Operation
7	7	YPresent	1 Y path Fabric is not present or not fully inserted - information derived from SWITCH_Y_PRESENT I/O pin 0 Normal operation
6	6	YFPAE	1 Y path Fabric Port is not available (the DASL receiver for that switch port is not synchronized - information is derived from FPAN_X I/O pin) 0 Normal operation
5	5	YSwClkE	1 Switch Clock Y missing information is sensed at the output of the multiplexer that allow to select the switch clock source 0 Normal operation
4	4	YInServE	Switch Board Y in service (Active/ready) 1 When switch Y in service is detected, remains active as long as switch Y is in service. 0 Switch Y is in protection mode
3	3	XPresent	1 X path Fabric is not present or not fully inserted (This information is directly derived from SWITCH_X_PRESENT I/O pin 0 Normal operation

Note: Some of the bits in this register are directly forced by the hardware. Even if they are Read/Write, when the microprocessor writes a value different from the current bit content, the hardware will immediately overwrite the microprocessor value.

Bits / Word	Bits / Bytes	Name	Description
2	2	XFPAE	1 X path Fabric Port is not available (the DASL receiver for that switch port is not synchronized - information directly derived from FPAN_X I/O pin) 0 Normal operation
1	1	XSwClkE	1 Switch Clock X missing - sensed at the output of the multiplexer that allows selection of the switch clock source 0 Normal operation
0	0	XInServE	Switch Board X in service (Active/ready) 1 When switch X in service is detected it remains active as long as switch X is in service 0 Switch X is in protection mode

Note: Some of the bits in this register are directly forced by the hardware. Even if they are Read/Write, when the microprocessor writes a value different from the current bit content, the hardware will immediately overwrite the microprocessor value.

4.1.1.30 ABIST Failure Test Status _X_Y Registers

This register is not accessible by the user and the bit should not be set to ‘1’. It is used only during Manufacturing test.

Unused								Unused								DEBUG_select_1					Unused	YABistM3Fail	YABistdone	DEBUG_select_0					DIAGOUT_enb_1	XABistM3Fail	XABistDone
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

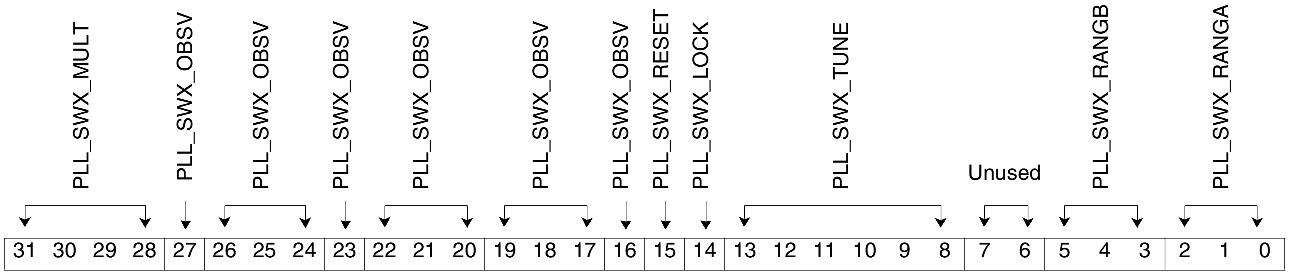
Reset Output Status (Power-on-Reset)	All ’0’s
Address in Word Mode	x'8C’
Address in Byte Mode	x'8C to 8F’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description	Notes
31-24	7-0		Unused	
23-16	7-0		Unused	
15-11	7-3	DEBUG_select_1	Select debug bus 1. Controls debug bus multiplexing.	1
10	2		Unused	
9	1	YABistM3Fail	1 ABIST Test Failure flag on M3 RAM 0 Normal operation	
8	0	YABistDone	1 ABIST Test complete Flag 0 Normal operation	
7-3	7-3	DEBUG_select_0	Select debug bus 0. Controls debug bus multiplexing.	1
2	2	DIAGOUT_enb_l	Select redirect the ABIST result on the ABIST_Diagout I/O 1 Y side 0 X side	
1	1	XABistM3Fail	1 ABIST Test Failure flag on M3 RAM 0 Normal operation	
0	0	XABistDone	1 ABIST Test complete Flag 0 Normal operation	

1. See debug bus definition, page 108, for details.

4.1.1.31 Switch X PLL Setting Register

These settings correspond to a VCO setting of 660 MHz. The PLL observe bits will automatically overwrite what has been written by the microprocessor.

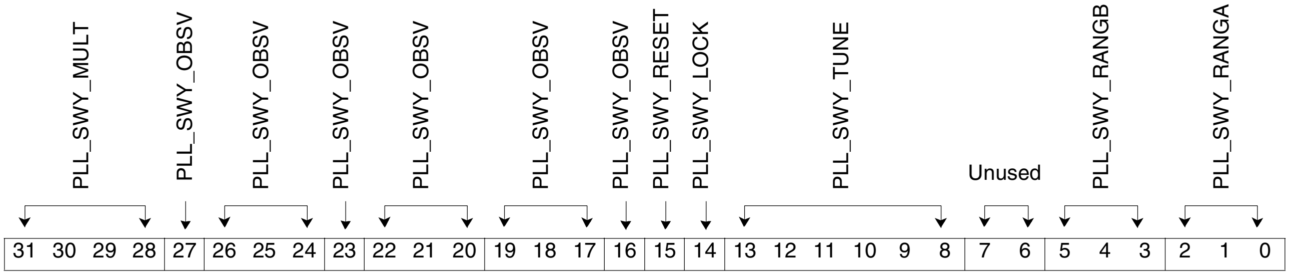


Reset Output Status (Power-on-Reset)	‘0800 C000’
Address in Word Mode	x'90’
Address in Byte Mode	x'90 to 93’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-28	7-4	PLL_SWX_MULT	Programmable Switch X PLL multiplier. 0010 Normal operation, as the reference clock is 55 - 62.5 MHz for a switch byte clock of 110 - 125 MHz.
27	3	PLL_SWX_OBSV	Switch X PLL Observe Reset
26-24	2-0	PLL_SWX_OBSV	Switch X PLL Observe: Multiplier bits 3-1.
23	7	PLL_SWX_OBSV	Switch X PLL Observe: Multiplier bit 0
22-20	6-4	PLL_SWX_OBSV	Switch X PLL Observe: buffered version of range B (2-0)
19-17	3-1	PLL_SWX_OBSV	Switch X PLL Observe: buffered version of range A (2-0)
16	0	PLL_SWX_OBSV	Switch X PLL Observe Reset (delayed by one clock)
15	7	PLL_SWX_RESET	1 Switch X PLL reset: equivalent to bypass mode 0 Normal operation after PLL programming
14	6	PLL_SWX_LOCK	1 Switch X PLL locked 0 PLL is unlocked
13-8	5-0	PLL_SWX_TUNE	Programmable Switch X PLL Tune. 010011 Normal operation as product of forward and feedback dividers is between (6/4)<M<6
7-6	7-6		Unused
5-3	5-3	PLL_SWX_RANGB	Programmable Switch X PLL Range B. 101 Normal operation as PLLOUTB frequency is 133 - 267 MHz
2-0	2-0	PLL_SWX_RANGA	Programmable Switch X PLL Range A. 010 PLLOUTA frequency is 66 - 134 MHz

4.1.1.32 Switch Y PLL Setting Register

These settings correspond to a VCO setting of 660 MHz. The PLL observe bits will automatically overwrite what has been written by the microprocessor.



Reset Output Status (Power-on-Reset)	‘0800 C000’
Address in Word Mode	x'94’
Address in Byte Mode	x'94 to 97’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-28	7-4	PLL_SWY_MULT	Programmable Switch Y PLL multiplier. 0010 Normal operation, as the reference clock is 55 - 62.5 MHz for a switch byte clock of 110 -125 MHz.
27	3	PLL_SWY_OBSV	Switch Y PLL Observe Reset
26-24	2-0	PLL_SWY_OBSV	Switch Y PLL Observe: Multiplier bit 3-1.
23	7	PLL_SWY_OBSV	Switch Y PLL Observe: Multiplier bit 0
22-20	6-4	PLL_SWY_OBSV	Switch Y PLL Observe: buffered version of range B (2-0)
19-17	3-1	PLL_SWY_OBSV	Switch Y PLL Observe: buffered version of range A (2-0)
16	0	PLL_SWY_OBSV	Switch Y PLL Observe Reset (delayed by one clock)
15	7	PLL_SWY_RESET	1 Switch Y PLL reset: equivalent to bypass mode 0 Normal operation after programming
14	6	PLL_SWY_LOCK	1 Switch Y PLL locked 0 Switch Y PLL is unlocked
13-8	5-0	PLL_SWY_TUNE	Programmable Switch Y PLL Tune. 010011 Normal operation as product of forward and feedback dividers is between (6/4)<M<6
7-6	7-6		Unused
5-3	5-3	PLL_SWY_RANGB	Programmable Switch Y PLL Range B. 101 Normal operation as PLLOUTB frequency is 133 - 267 MHz
2-0	2-0	PLL_SWY_RANGA	Programmable Switch Y PLL Range A. 010 Normal operation as PLLOUTA frequency is 66 - 134 MHz

4.1.1.33 Chip ID Register

Chip ID

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

Reset Output Status (not applicable, wired information)

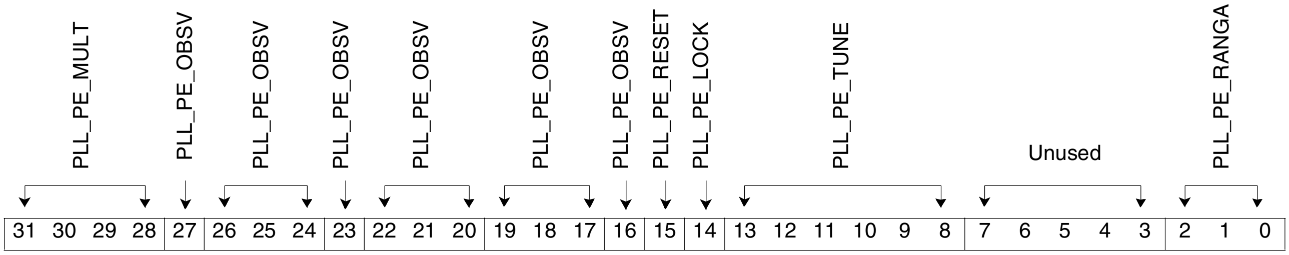
Address in Word Mode x'98'

Address in Byte Mode x'98 to 9B'

Access Type Read

Bits / Word	Bits / Bytes	Name	Description
0-31		ChipID	34423600

4.1.1.34 Protocol Engine PLL Setting Register (PE PLL Register)



Reset Output Status (Power-on-Reset) ‘0800 C000’

Address in Word Mode x'9C’

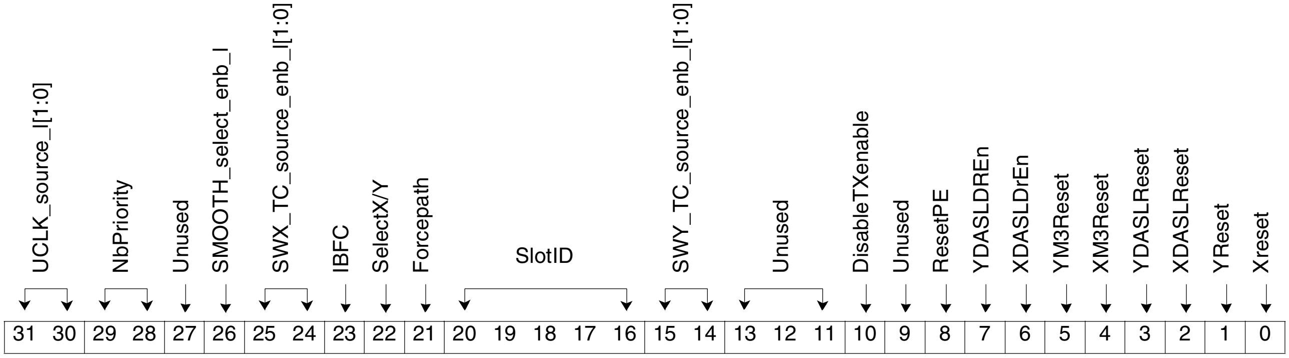
Address in Byte Mode x'9C to 9F’

Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-28	7-4	PLL_PE_MULT	Programmable PE PLL multiplier. 0001 Frequency multiplication is 1
27	3	PLL_PE_OBSV	PE X PLL Observe Reset
26-24	2-0	PLL_PE_OBSV	PE X PLL Observe: Multiplier bit 3-1.
23	7	PLL_PE_OBSV	PE PLL Observe: Multiplier bit 0
22-20	6-4	PLL_PE_OBSV	PE PLL Observe: buffered version of range B (2-0)
19-17	3-1	PLL_PE_OBSV	PE PLL Observe: buffered version of range A (2-0)
16	0	PLL_PE_OBSV	Switch PE PLL Observe Reset (delayed by one clock)
15	7	PLL_PE_RESET	1 PE PLL reset: equivalent to bypass mode 0 Normal operation after PLL programming
14	6	PLL_PE_LOCK	1 PE PLL locked 0 PE PLL is unlocked
13-8	5-0	PLL_PE_TUNE	Programmable Switch PE PLL Tune. 010010 Product of forward and feedback dividers is between (6/4)<M<6
7-3	7-3	Unused	
2-0	2-0	PLL_PE_RANGA	Programmable PE PLL Range A. 010 PLLOUTA is 66 -134 MHz

4.1.1.35 Common Control Register

The PLL observe bits automatically overwrite the microprocessor writes.



Reset Output Status (Power-on-Reset)	‘0500413F’
Address in Word Mode	x'A0’
Address in Byte Mode	x'A0 to A3’
Access Type	Read/Write

Bits/ Word	Bits/ Bytes	Name	Description
31-30	7-6	UCLK_source_I[1:0]	Select Ingress Clock source: 00 Microprocessor 01 PE clock used to time the PE interface 10 External smoothing PLL 11 Microprocessor clock
29-28	5-4	NbPriority	Priority Enable controls the cycling process of Output Queue Grant and Shared Memory Grant information from the switch 00 Enable only Priority 0 01 Enable Priority 0 and 1 enabled 10 Enable Priority 0, 1 and 2 11 Enable Priority 0, 1, 2 and 3 (all four priorities in use)
27	3		Unused
26	2	SMOOTH_select_enb_I	Smooth PLL out selection and driver enable 0 The selected switch clock used by the converter is not the one used to drive an external smoothing PLL. In this case, a clock coming from the PE itself (or an external oscillator) is used to drive the PE interface rather than the From_Smooth_PLL_out. 1 The To_Smooth_PLL_In interface line is used to drive the PE interface to support a smooth switching from one clock plane to the other in case of switch plane switchover. In that case, the switch and the protocol engines are clock synchronous.
25-24	1-0	SWX_TC_source_enb_I[1:0]	Switch_X Clock selector: 11 Normal setting: selects Switch Fabric X Clock for Switch X interface 10 Selects external TEST_CLK oscillator for switch X interface 01 Selects external MP_CLK oscillator for switch X interface 00 Selects Force Clock ‘1’ for Switch interface X (may be used for clock miss detection)

Bits/ Word	Bits/ Bytes	Name	Description
23	7	IBFC	In Band Flow Control (Out of Band Flow Control is default) 1 OQG and SMG information are carried in the packet header of each egress packet. On ingress the in band TxPause is extracted from the packet qualifier and provided to the egress FIFO flow control mechanism to control the IBM 28.4 G Packet Routing Switch SND_GRANT interface line. 0 Out of Band Flow control (also used for switch loopback)
22	6	SelectX/Y	Select the plane in service in case Forcepath bit 21 is enabled. 0 X in service 1 Y in service.
21	5	Forcepath	1 Force plane in use through this register bit 22. Otherwise it is through Inservice lines 0 Normal operation
20-16	4-0	SlotID	Senses in which backplane slot the converter is plugged. It is directly mapped to Slot_IDn[4:0] 0 Corresponding bit is asserted 1 Correspinding bit is deasserted
15-14	7-6	SWY_TC_source_enb_l[1:0]	Switch_Y Clock selector: 11 Normal: selects Switch Fabric Y clock for Switch Y interface 10 Selects external TEST_CLK oscillator for switch Y interface 01 Selects external MP_CLK oscillator for switch Y interface 00 Selects Force Clock ‘1’ for switch interface Y (may be used for clock miss detection)
13-11	5-3		Unused
10	2	DisableTXenable	1 Validates TxData on egress PE interface 0 Normal operation
9	1		Unused
8	0	ResetPE	1 Reset PE interface and all Registers, except this one. Does not reset Microprocessor front end, PLL register, or reset register 0 Normal operation
7	7	YDASLDREn	1 Normal operation 0 Y path DASL Driver Enable. Same result as interface line $\overline{\text{DASL\_DRV\_ENB}}$
6	6	XDASLDrEn	1 Normal operation 0 X path DASL Driver Enable. Same result as interface line DASL_DRV_ENB
5	5	YM3Reset	1 M3 reset SDC Path Y Reset 0 Normal operation
4	4	XM3Reset	1 M3 reset SDC Path X Reset 0 Normal operation
3	3	YDASLReset	1 DASL Path Y Reset 0 Normal operation
2	2	XDASLReset	1 DASL Path X Reset 0 Normal operation
1	1	YReset	1 Reset Path Y 0 Normal operation
0	0	Xreset	1 Reset Path X 0 Normal operation

4.1.1.36 Interrupt Register Indirection

Unused	Unused	Unused	YEvent1B1	YEvent1B0	XEvent1B1	XEvent1B0	Unused	Event2B1	Event2B0
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3 2	1	0

Reset Output Status (Power-on-Reset) All '0's

Address in Word Mode x'A4'

Address in Byte Mode x'A4 to A7'

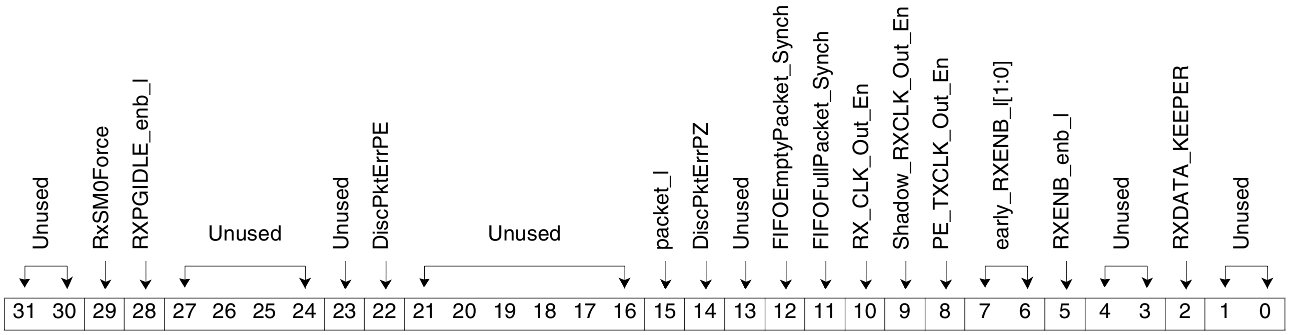
Access Type Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-8	7-0		Unused
7	7	YEvent1B1	Event 1 on plane Y byte 1
6	6	YEvent1B0	Event 1 on plane Y byte 0
5	5	XEvent1B1	Event 1 on plane X byte 1
4	4	XEvent1B0	Event 1 on plane X byte 0
3-2	3-2		Unused
1	1	Event2B1	Event 2 byte1
0	0	Event2B0	Event 2 byte 0

Note: The PLL Clock missing bits are not implemented.

Note: The bits in this register are followers of what is occuring in other triggering registers. To clear a bit in this register, its source must be cleared. For example, if a bit is set in Register 18 (Interrupt Enable _X), Register 38 (Event 1 Interrupt Enable _Y), or Register 84 (Event 2 µP Interrupt Enable_X and _Y), the corresponding bit in this register will be set. When the bits in those registers are cleared, the corresponding bit in this register is cleared.

4.1.1.37 Ingress PE Settings Register (INGRESS_PE_INTERFACE (IPI) - Receive)



Reset Output Status (Power-on-Reset, PE reset)	‘20018000’
Address in Word Mode	x'C0’
Address in Byte Mode	x'C0 to C3’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31-30	7-6		Unused
29	5	RxSM0Force	Force Shared Memory 00 de-Grant 0 (highest priority) for FIFO full 1 In band flow control 0 Out of band flow control
28	4	RXPGIDLE_enb_l	1 Select whether idle packets are to be received 0 Out of band flow control (no idle packet on PE interface)
27-24	3-0		Unused
23	7		Unused
22	6	DiscPktErrPE	0 Do not discard packets with PE bus parity in error 1 Discard packets with bad parity on PE interface
21-16	5-0		Unused
15	7	packet_l	1 Operate in packet mode on PE interface 0 Word mode enabled: not supported
14	6	DiscPktErrPZ	1 Discard packets with bad switch header parity 0 Do not discard packets with bad switch header parity
13	5		Unused
12	4	FIFOEmptyPacket_Synch	1 FIFO Empty is locked high until the next start of packet 0 Normal FIFO operation
11	3	FIFOFullPacket_Synch	1 FIFO full is locked high until the next start of packet 0 Normal FIFO operation
10	2	RX_CLK_Out_En	1 Enable PE RCV clock out 0 Disable PE RCV clock out
9	1	Shadow_RXCLK_Out_En	1 Enable shadow receive clock out driver 0 Disable shadow receive clock out driver
8	0	PE_TXCLK_Out_En	1 Enable the PE TX transmit clock out driver 0 Disable PE TX clock out driver

Bits / Word	Bits / Bytes	Name	Description
7-6	7-6	early_RXENB_l[1:0]	RXDATA vs. RXENB calibration up to four URXCLK clock cycles. 00 0 URXCLK cycle delay 01 1 URXCLK cycle delay 10 2 URXCLK cycle delay - standard operation 11 3 URXCLK cycle delay
5	5	RXENB_enb_l	Receive Enable interface line assertion for FIFO full 1 Enables RXENB to control flow for ingress FIFO full 0 RXENB interface line is not used for ingress link level flow control (typically IBFC mode)
4-3	4-3		Unused
2	2	RXDATA_KEEPER	Ingress PE data delay for correct resampling Bit Selected Clock 0 ShadowRxClockOut - switch loopback 1 ShadowRXclockIn - Normal operation
1-0	1-0		Unused

4.1.1.38 Egress PE Setting Register (EGRESS_PE_INTERFACE - Transmit)

TX_FORCE_PERR	TXPRTY_enb_l	late_TXFULL_l[1:0]		TxFULL_enb_l	TXFULL_PHASE_enb_l	Unused		BM4_Pos_E				BM3_Pos_E				BM2_Pos_E				BM1_Pos_E				PQ_Pos_E				SGrantPriDetEn	RX2TXENB_enb_l	SwitchLoop	TXPGIdleSend
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reset Output Status (Power-on-Reset, PE reset)	‘38432100’
Address in Word Mode	x'C4’
Address in Byte Mode	x'C4 to C7’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31	7	TX_FORCE_PERR	1 Force parity error on egress bus 0 Disable parity error insertion on the egress UTOPIA-3 interface
30	6	TXPRTY_enb_l	Egress PE interface parity enable 1 Enable assertion of the parity bit 0 Disable assertion of the parity bit
29-28	5-4	late_TXFULL_l[1:0]	$\overline{\text{TXFULL}}$ vs. $\overline{\text{TXENB}}$ calibration up to four UTXCLK clock cycles. 00 1 UTXCLK cycle delay 01 2 UTXCLK cycle delay 10 3 UTXCLK cycle delay 11 4 UTXCLK cycle delay (operates as described in the UTOPIA-3 document)
27	3	TxFULL_enb_l	1 Enable the Converter to use $\overline{\text{TXFULL}}$ interface line 0 Disable any action of $\overline{\text{TXFULL}}$ on the converter operation
26	2	TXFULL_PHASE_enb_l	Select the active level of $\overline{\text{TXFULL}}$ 1 Operate with $\overline{\text{TXFULL}}$ condition asserted as active high 0 Operate with the standard UTOPIA-3 $\overline{\text{TXFULL}}$ phase (active low)
25-24	1-0		Unused
23-20	7-4	BM4_Pos_E	Bit Map 4 byte position in packet 0100 Put OQG4 as first byte in second word of packet
19-16	3-0	BM3_Pos_E	Bit Map 3 byte position in packet 0011 Put OQG3 as forth byte in first word of packet
15-12	7-4	BM2_Pos_E	Bit Map 2 byte position in packet 0010 Put OQG2 as third byte in first word of packet
11-8	3-0	BM1_Pos_E	Bit Map 1 byte position in packet 0001 Put OQG1 as second byte in first word of packet
7-4	7-4	PQ_Pos_E	Packet Qualifier byte position in packet 0000 Put PQ as first byte in first word of packet

Bits / Word	Bits / Bytes	Name	Description	
3	3	SGrantPriDetEn	1	Support for IBM Packet Routing Switch Send Grant per Priority Enable. (Set to 0 in out of band flow control)
			0	IBFC: operate with single send grant priority ‘to the IBM 28.4G Packet Routing Switch’.
2	2	RX2TXENB_enb_l	1	Routeinternal $\overline{\text{RXENB}}$ to the internal $\overline{\text{TxFull}}$ to flow control the egress path during switch loop
			0	Internal $\overline{\text{RXENB}}$ is asserted to egress path during switch loop (no data can go through)
1	1	SwitchLoop	1	Send TXDATA[31:0] on PE ingress interface while in switch loop X or Y
			0	Normal Operation - data continues to exit converter
0	0	TXPGIdleSend	1	Enable send idle on egress interface
			0	Operate as described in the UTOPIA-3 document: only data packets are sent

4.1.1.39 Common PE Setting Register (PE (Common))

Grant_from_TxFull_Enable	Grant_from_PQ_Enable	Unused		BM4_Pos_I				BM3_Pos_I				BM2_Pos_I				BM1_Pos_I				PQ_Pos_I				Unused	frame_length_I			Mode32Port	TX2RXPRTYenb_I	PEIE	BackpresLinkD
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

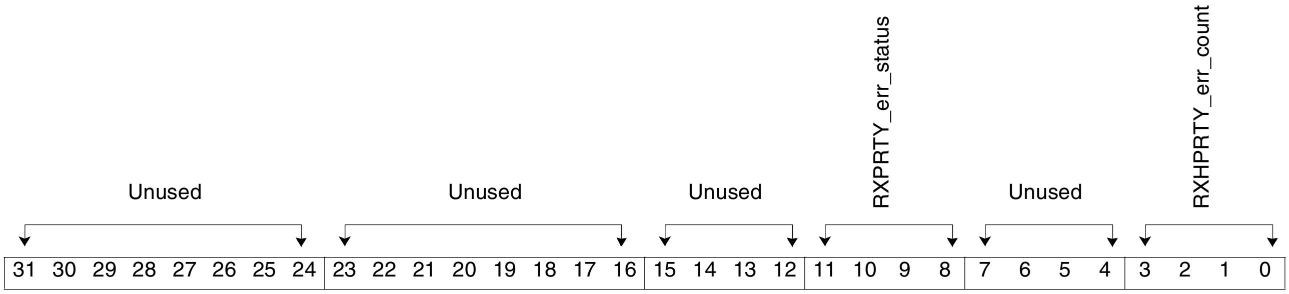
Reset Output Status (Power-on-Reset)	‘01321008’
Address in Word Mode	x'C8’
Address in Byte Mode	x'C8 to CB’
Access Type	Read/Write

Bits / Word	Bits / Bytes	Name	Description
31	7	Grant_from_TxFull_Enable	1 PE sends $\overline{\text{TXFULL}}$ to control the flow of packets from the converter may be used for any flow control mode 0 $\overline{\text{TXFULL}}$ is not used for link level flow control
30	6	Grant_from_PQ_Enable	1 Extract TxPause flow control information from incoming packet qualifier to control the send grant per priority to the switch, or to control the flow of packets from the converter. 0 Out of band flow control
29-28	5-4		Unused
27-24	3-0	BM4_Pos_I	Bit Map 4 byte position in packet 0001 Get BM4 from second byte of first word (during BM1 swap BM4 is moved there from first byte of second word.)
23-20	7-4	BM3_Pos_I	Bit Map 3 byte position in packet 0011 Get BM3 from fourth byte of first word
19-16	3-0	BM2_Pos_I	Bit Map 2 byte position in packet 0010 Get BM2 from third byte of first word
15-12	7-4	BM1_Pos_I	Bit Map 1 byte position in packet 0001 Get BM1 from second byte of first word
11-8	3-0	PQ_Pos_I	Packet Qualifier byte position in packet 0000 Get packet qualifier from first byte of first word
7	7		Unused
6-4	6-4	frame_length_I	Set the length of the LU: 000 16-byte LU (64-byte packet) 001 17-byte LU (68-byte packet 010 18-byte LU (72-byte packet 011 19-byte LU (76-byte packet) 100 20-byte LU (80-byte packet)
3	3	Mode32Port	1 Select 32-bit bit map header mode 0 Select 16-bit bit map header mode

Bits / Word	Bits / Bytes	Name	Description
2	2	TX2RXPRTYenb_l	1 Check PE data parity in switch loopback mode 0 Do not check PE data parity in switch loopback mode
1	1	PEIE	1 Normal operation 0 Disable PE Interface. Default mode is inactive. Inhibit output driver (equivalent to Hi-Z)
0	0	BackpresLinkD	1 operate in out of band flow control 0 Disable back pressure serial link (set to Hi-Z)

4.1.1.40 Parity and CRC Error Count Register (PARITY_Error_Count)

The microprocessor has the highest priority when writing into this register.



Reset Output Status (Power-on-Reset, PE reset) All ’0’s

Address in Word Mode x'CC’

Address in Byte Mode x'CC to CF’

Access Type Read/Write

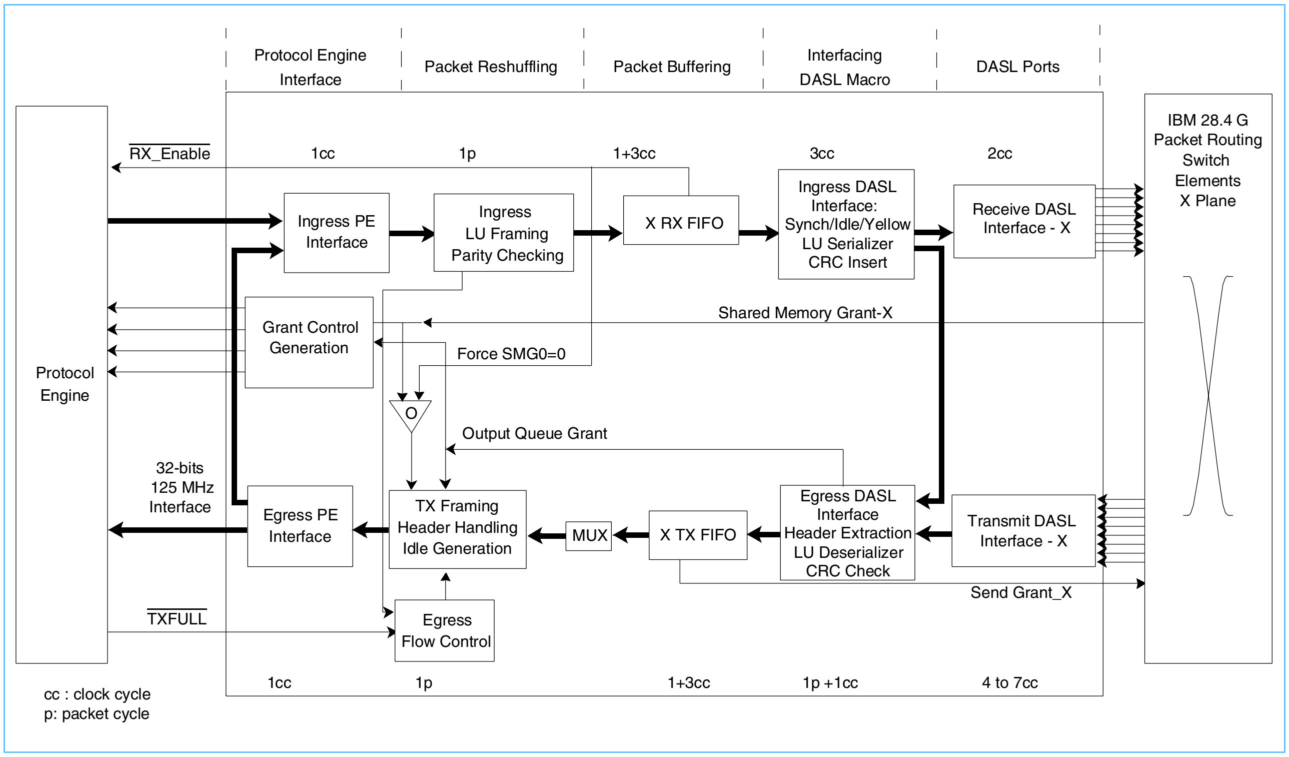
Bits / Word	Bits / Bytes	Name	Description
31-24	7-0		Unused
23-16	7-0		Unused
15-12	7-4		Unused
11-8	3-0	RXPRTY_err_status	Ingress PE bus Parity error count.
7-4	7-4		Unused
3-0	3-0	RXHPRTY_err_count	IBM 28.4 G Packet Routing Switch Header Parity Error Count

5. IBM Packet Routing Switch Serial Interface Converter Latency

There are two kinds of flow control latencies to be considered: packet latency and flow control latency.

Packet latency is shown below. Packet reshuffling is the main cause because, on ingress, the entire packet must be received before it can be ascertained if the packet must be discarded due to errors. The re-synchronization layer between the PE clock domain and the switch clock domain is the secondary contributor to packet latency.

Figure 33: Converter Latency Diagram


Protocol Engine Interface
Packet Reshuffling
Packet Buffering
Interfacing DASL Macro
DASL Ports
IBM 28.4 G Packet Routing Switch Elements X Plane
Protocol Engine
RX_Enable
1cc
1p
1+3cc
3cc
2cc
Ingress PE Interface
Ingress LU Framing Parity Checking
X RX FIFO
Ingress DASL Interface: Synch/Idle/Yellow LU Serializer CRC Insert
Receive DASL Interface - X
Grant Control Generation
Shared Memory Grant-X
Force SMG0=0
O
Output Queue Grant
32-bits 125 MHz Interface
Egress PE Interface
TX Framing Header Handling Idle Generation
MUX
X TX FIFO
Egress DASL Interface Header Extraction LU Deserializer CRC Check
Transmit DASL Interface - X
Send Grant_X
TXFULL
Egress Flow Control
1cc
1p
1+3cc
1p +1cc
4 to 7cc
cc : clock cycle
p: packet cycle

Flow control information latency has two impacts:

- The shared memory thresholds within the switch have to be set to low sharing (for example: Priority 3 to 16 Packets, Priority 2 to 32 packets, Priority 1 to 48 packets, and Priority 0 to 64 packets).
- The flow control check bit 10 of mode register must be disabled.

6. JTAG Description

The IBM Packet Routing Switch Serial Interface (the converter) is compliant with the IEEE Standard Test Access Port and Boundary-Scan Architecture for testing and debugging components assembled at the board level. (See "IEEE Standard Test Access Port and Boundary-Scan Architecture", doc. IEEE Std 1149.1-1990, and the IBM ASIC Products Application Note "IEEE 1149.1 Boundary-Scan in IBM ASICs", 11/97, version 4.)

JTAG test architecture consists of a Test Access Port (TAP) and associated controller, an Instruction Register, and (on the converter) three Test Data Registers named the Bypass Register, the Idcode Register, and the Boundary Scan Register.

The TAP, a general-purpose port, provides access to the implemented test support functions defined by the JTAG Standard. It consists of the following five ports: the Test Clock Input (TCK), the Test Mode Select Input (TMS), the Test Data Input (TDI), the Test ReSeT input (TRST), and the Test Data Output (TDO). The TAP controller is a synchronous finite state machine that responds to changes at the TMS and TCK signals and controls the sequence of operations of the internal test logic. It generates the clock and control signals required to shift data down the Instruction Register and Test Data Registers. These registers constitute separate shift-register based paths that are connected in parallel between the TDI and TDO.

The Instruction Register is three bits long. It captures the Instruction Opcode desired during a test. The converter supports the following instructions declared in the JTAG Standard: EXTEST, INTEST, SAMPLE, IDCODE, HIGHZ, CLAMP and BYPASS.

Table 17: Supported JTAG Instructions

Instruction Name	Instruction Opcode
Extest	000
Intest	001
Sample	010
IDcode	011
Hi-Z	100
Clamp	110
Bypass	111

The Bypass Register is a one bit long shift register and is scannable during the BYPASS instruction. It provides a minimum length serial path to move test data between TDI and TDO. Use the Bypass Register to speed access to JTAG registers in other components on a board-level test data path.

The Boundary Scan Register is a shift register that allows sampling and/or forcing of signals flowing into and out of the system logic through the system ports.

During the JTAG operation the following converter PIs must be at the specific value indicated in the following table. As these PIs are already internally tied up or tied down, they do not need any user force action except the RI line, which must be forced to 1 externally.

Table 18: Compliance Pattern

Signal Names	Value
CE0_IO	0
CE0_Scan	0
CE0_TEST	0
CE1_A	1
CE1_B	1
TEST_B2	1
CE1_C1	1
CE1-C2	1
TEST_C3	1
LT	1
DI1	1
DI2	1
RI	1

The IDcode register is a 32-bit SRL 32h’34423600’. Physically the IDcode register is implemented this way:

TDI->Version (31:28)->Location (27:24)->ID -UDASL P/N- (23:12)->Manufacturer ID (11:1)->’1’->TDO converter.

Table 19: ID Code Description

ID code	Value
Version (31:28)	0000
Location (27:24)	0000
IBM Packet Routing Switch Serial Interface Converter Part Number (23:12)	011000110010
Manufacturer ID (11:1)	01000100001
Required by the standard	1

7. I/O Definition and Package Pin Assignment

7.1 Signals Description

All functional signals are 3.3 V LVTTL compatible for drivers and receivers except the following:

- Protocol engine interface
- Protocol engine clocks provided by the IBM Packet Routing Switch Serial Interface Converter
- Back pressure serial link
- LSDD and JTAG test signals

These are based on the tri-state driver/receiver (BP2550 type) that interfaces 2.5 V internal functions with 3.3 V-tolerant 2.5 V CMOS drivers and receivers off chip bidirectional data buses. The driver is 50 Ω source-terminated.

The switch fabric clocks are balanced HSTL levels.

Table 20: Tests Signals

Name	Input/ Output	Levels	Description	Notes
$\overline{\text{DI1}}$		LVCMOS	Non-test driver inhibit for all chip boundary outputs. 0 Chip boundary outputs are disabled and in tri-state. 1 Inactive, all boundary outputs are controlled by normal functions. An internal pull-up resistor forces the inactive state. Boundary outputs are chip outputs or common I/O’s that serve as primary outputs of the internal boundary logic. Feed directly by boundary latches or special boundary logic books that make up the boundary logic.	
$\overline{\text{DI2}}$	Input	LVCMOS	The test driver inhibit for all chip non-boundary outputs. 0 Chip non-boundary outputs are disabled and in tri-state. An internal pull-up resistor forces the inactive state. 1 Inactive, all non-boundary outputs are controlled by normal functions. Non-boundary outputs are chip outputs or common I/O’s that bring test function and LSSD scan data directly to and from the internal boundary logic.	
$\overline{\text{RI}}$	Input	LVCMOS	Gates all boundary receivers during internal test to prevent unknown states from entering the internal logic and to reduce switching activities. RI pad must be tied up externally for system mode. When active, all boundary receivers are placed in a known state independent of the receiver input.	
CE1_A	Input	LVCMOS	External source of the internal SRL scan A clock used during LSSD test to enable the tester to independently source the internal SRL clocks from the primary inputs.	1
CE1_B	Input	LVCMOS	External source of the internal SRL scan B clock used during LSSD test to enable the tester to independently source the internal SRL clocks from the primary inputs.	1
TEST_B2	Input	LVTTL	External source of the internal SRL scan B clock used during LSSD test to enable the tester to independently source the internal SRL clocks from the primary inputs.	1
CE1_C1	Input	LVCMOS	External source of the internal SRL scan C clock used during LSSD test to enable the tester to independently source the internal SRL clocks from the primary inputs (used for logic).	1

1. An internal pull-up resistor forces the inactive state.
2. Must be kept LOW during normal PLL operation.

Table 20: Tests Signals

Name	Input/ Output	Levels	Description	Notes
CE1_C2	Input	LVCMOS	External source of the internal SRL scan C clock used during LSSD test to enable the tester to independently source the internal SRL clocks from the primary inputs (used for RAMS).	1
CE0_IO	Input	LVCMOS	Used to force the JTAG EXTEST operation for IBM test methodology purposes.	1
CE0_Scan	Input	LVCMOS	Gates both the A and B LSSD test clocks.	1
CE0_TEST	Input	LVCMOS	Forces "0" in system mode and ‘1’ in LSSD test mode.	1
TEST_C3	Input	LVTTL	External source of the internal GRA scan C clock used during LSSD test to enable the tester to independently source the internal GRA clocks from the primary inputs.	1
PE_TESTIN	Input	LVCMOS	Programs the PLL to perform parametric testing at the wafer and module level.	2
SWITCH_X_TESTIN	Input	LVCMOS	Programs the DASL PLL for fabric X to perform parametric testing at the wafer and module levels.	2
SWITCH_Y_TESTIN	Input	LVCMOS	Programs the DASL PLL for fabric Y to perform parametric testing at the wafer and module levels.	2
PE_TESTOUT	Output	LVCMOS	Monitor TESTOUT output during PLL test to verify the PLL output frequency.	
SWITCH_X_TESTOUT	Output	LVCMOS	Monitor TESTOUT output during PLL test to verify the PLL output frequency.	
SWITCH_Y_TESTOUT	Output	LVCMOS	Monitor TESTOUT output during PLL test to verify the PLL output frequency.	
ABIST_CLK	Input	LVCMOS	Connected to the ABIST controller STCLK input and used only during test operation. Keep this signal inactive (tied to ground) during system mode to reduce power consumption in the BIST logic.	
ABIST_DIAGOUT	Output	LVTTL	Each SRAM DIAGOUT output is multiplexed to this ABIST_DIAGOUT PO to observe the pass/fail flag of individual arrays during the ABIST diagnostic mode. See register @8C bit 2 for X or Y result on that pin.	

1. An internal pull-up resistor forces the inactive state.
2. Must be kept LOW during normal PLL operation.

Table 21: JTAG Interface External Signals

Name	Input/Output	Levels	Description	Notes
TCK	Input	3.3 V LVTTL receiver tri-state CIO	Test Clock Input	1, 2
TMS	Input	3.3 V LVTTL receiver tri-state CIO	Test Mode Select Input	1, 2
TDI	Input	3.3 V LVTTL receiver tri-state CIO	Test Data Input	1, 2
TDO	Output	3.3 V LVTTL 50 Ω tri-state CIO	Test Data Output	1, 2
TRST	Input	3.3 V LVTTL receiver tri-state CIO	Test Reset Input must be asserted during the power-on-reset to reset the JTAG control logic.	1, 2

1. An internal pull-up resistor forces the inactive state
2. . See IEEE 1149.1 specification for details.

Table 22: Processor Interface Signals

Name	Input/ Output	Levels	Description
$\overline{\text{MP\_SEL}}$	Input	LVTTL	IBM Packet Routing Switch Serial Interface converter select for processor access
$\overline{\text{MP\_WR}}$	Input	LVTTL	Write/Read indicator on processor bus
$\overline{\text{MP\_PRDY}}$	Output	LVTTL	Ready signal for transfer complete indication
$\overline{\text{MP\_INT}}$	Output	LVTTL	Interrupt to processor
MP_ADDR_7... MP_ADDR_0	Inputs	LVTTL	Processor bus address
MP_ADD_PRTY	Inputs	LVTTL	Processor bus parity (optional and can be set through an interface pin)
MP_DATA_7... MP_DATA_0	Input/ Output	LVTTL	Processor Data bus
MP_DATA_PRTY	Input/ Output	LVTTL	Processor Data bus parity (optional and can be set through an interface pin)
$\overline{\text{MP\_PRTY\_ENB}}$	Input	LVTTL	Enable generation of and checking of the microprocessor data and address parity
MP_BURST_MODE	Input	LVTTL	Enable the operation in either single byte mode access or in byte burst mode compatible with I960. When tied to ground, the chip operates in byte mode access, when tied to 3.3 V it operates in word mode access.

Table 23: IBM Packet Routing Switch Serial Interface Converter (the converter) Signals

Name	Input/Output	Levels	Description
DASL_X_TX_0 / DASL_X_TX_0N... DASL_X_TX_7 / DASL_X_TX_7N	Input	HSTL	Differential (400 to 500 Mbps) signals for input port from switch X
DASL_X_RX_0 / DASL_X_RX_0N... DASL_X_RX_7 / DASL_X_RX_7N	Output	HSTL	Differential (400 to 500 Mbps) signals for output port to switch X
DASL_Y_TX_0 / DASL_Y_TX_0N... DASL_Y_TX_7 / DASL_Y_TX_7N	Input	HSTL	Differential (400 to 500 Mbps) signals for input port from switch Y
DASL_Y_RX_0 / DASL_Y_RX_0N... DASL_Y_RX_7 / DASL_Y_RX_7N	Output	HSTL	Differential (400 to 500 Mbps) signals for output port to switch Y
SEND_GNT_X	Ouput	LVTTL	0 The switch on switch board X is not allowed to send data to the converter 1 The switch is allowed to send data to the converter. In reset mode this signal is forced to Hi-Z.
SEND_GNT_Y	Output	LVTTL	0 The switch on switch board Y is not allowed to send data to the converter 1 The switch is allowed to send data to the converter. In reset mode this signal is forced to Hi-Z.
MEM_GNT_X_3... MEM_GNT_X_0	Input	LVTTL	Indicates that the shared memory is not full for a given priority. 0 The switch on switch board X cannot accept anymore data from the converter 1 The switch can accept more data from the converter
MEM_GNT_Y_3... MEM_GNT_Y_0	Input	LVTTL	Indicates that the shared memory is not full for a given priority. 0 The switch on switch board Y cannot accept anymore data from the converter 1 The switch can accept more data from the converter
$\overline{\text{DASL\_DRV\_ENB}}$	Input	LVTTL	Sets DASL drivers in Hi-Z until the board housing the converter is fully inserted. Insures that both ends of the adapter board to which the converter is soldered are fully inserted. An internal pull-up resistor forces the inactive state.

Note: The electrical level supported by the converter lines is called High Speed Transceiver Logic (HSTL). It is specified in JEDEC's JESD8-6 standard.

Figure 34: Detection of Line Card Fully Inserted

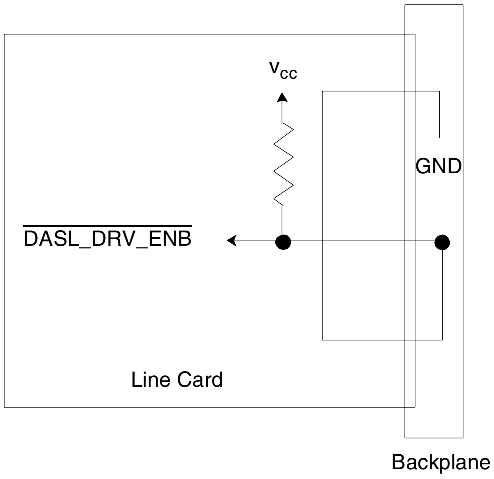


Table 24: Receive PE Interface Signals See PE interface description (ingress and egress descriptions)for functional description

Name	Input/Output	Levels	Description
RXDATA_31... RXDATA_0	Input	LVCMOS	Input Data Bus from PE to the converter
RXPRTY	Input	LVCMOS	Provide optional odd parity checking on input Data Bus from PE to the converter
RXSOP	Input	LVCMOS	Start of packet from PE to converter
RXPAV	Input	LVCMOS	Indicates that at least one packet is available in the PE
$\overline{\text{RXENB}}$	Output	LVCMOS	Asserted by converter during packet cycles when the PE is allowed to send data

Table 25: Transmit PE Interface Signals

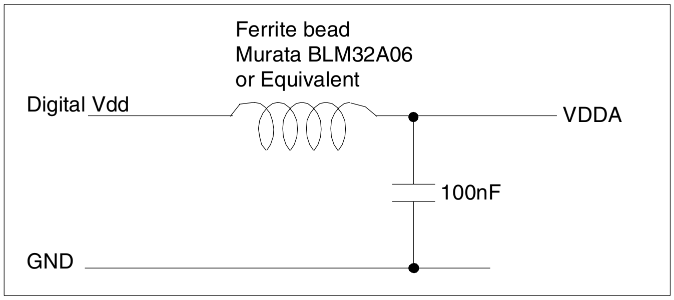
Name	Input/Output	Levels	Description
TXDATA_31... TXDATA_0	Output	LVCMOS	Output Data Bus from the converter to the PE. Signals are Hi-Z when the converter is not selected
TXPRTY	Output	LVCMOS	Provide optional odd parity bit on output Data Bus from the converter to PE. Signals are Hi-Z when the interface is not selected
TXSOP	Output	LVCMOS	Asserted by the IBM Packet Routing Switch Serial Interface Converter to the PE to indicate the position of the start of packet. Signal is Hi-Z when the interface is not selected.
$\overline{\text{TXFULL}}$	Input	LVCMOS	The PE has no space available for any additional packets
$\overline{\text{TXENB}}$	Output	LVCMOS	Asserted by converter during packet cycles when the PE is allowed to receive data

Table 26: Clocking/PLL External Signals

Name	Input/ Output	Levels	Description
+SWITCH_X_CLK / -SWITCH_X_CLK	Input	HSTL (balanced)	Switch Core X system clock used for the internal clock generation in the switch X interface side (clock frequency is 55 - 62.5 MHz).
+SWITCH_Y_CLK / -SWITCH_Y_CLK	Input	HSTL (balanced)	Switch Core Y system clock used for the internal clock generation in the switch Y interface side (clock frequency is 55 - 62.5 MHz).
MP_CLK	Inputs	LVTTL	Processor Bus clock input: 25 - 66 MHz. This clock also drives the PE interface when no other clock has been selected by setting register @A0 bit 30-31 to b'00' or '11' (default settings)
PE_CLK	Input	LVCMOS	Clock used to drive all converter PE side actions. Clock source can drive the PE interface by setting register @A0 bit 31-30 to b'01'
FROM_SMOOTH_PLL_OUT	Input	LVTTL	Clock derived from the in service clock and smoothed by an external PLL in case of clock switch-over. Used to drive all PE side actions of the converter. Clock source can drive the PE interface by setting register @A0 bit 31-30 to b'10'
TO_SMOOTH_PLL_IN	Output	LVTTL	Raw clock derived from the in service clock source of either switch fabric derived from either the switch X or the switch Y clock source depending on the status of the Switch X or Y InService lines from the switch core. It can be forced to be either switch X or Y clock according to the plane which has been forced in the configuration register @A4 bits 21 and 22. Enable bit 26 at configuration @A4 enables this signal
PE_TXCLK_OUT	Ouput	LVCMOS	Clock delivered by the converter that allows the transfer/synchronization of the TXDATA[31:0] and their associated controls from the converter to the PE. Clock source is determined by configuration register @A0 bit 31-30. Enabling this clock's driver depends on register @C0 bit 8.
PE_RXCLK_out	Output	LVCMOS	Clock delivered by the converter in case PE_CLK is not available on the PE layer. Allows the transfer/synchronization of the RXDATA[31:0] and their associated controls from the PE to the DASL chip. Clock source is determined by configuration register @A0 bit 31-30. Enabling this clock's driver depends on register @C0 bit 10
SHADOW_RXCLK_OUT	Output	LVCMOS	Clock delivered by the converter to allow a matched sampling with the Receive data coming from the PE. It can be connected to either an on board delay line or the attached PE to provide (along with SHADOW_RXCLOCK_IN) a clock phase incurring the same delay as the data. Enabling this clock's driver depends on register @C0 bit 10.
SHADOW_RXCLK_IN	Input	LVCMOS	Clock derived from the SHADOW_RXCLK_OUT after having passed through either an onboard delay line or the PE. It can be used to sample the ingress receive data.
SWITCH_X_VDDA	Input	PLL Analog VDD	Required to get a dedicated filtered voltage to switch X the PLL
SWITCH_Y_VDDA	Input	PLL Analog VDD	Required to get a dedicated filtered voltage to switch Y the PLL
PE_VDDA	Input	PLL Analog VDD	Required to get a dedicated filtered voltage to PE the PLL.

To provide isolation from the noisy internal digital V_{dd} signal, VDDA is brought to a package pin. If little noise is expected at the board level, then VDDA can be connected directly to the digital V_{dd} plane. In most circumstances, it is prudent to palce a filter circuit on VDDA as shown below. All wire lengths should be kept as short as possible to minimize coupling from other signals. The impedance of the ferrite bead should be much greater than that of the capacitor at frequencies where noise is expected. Many applications have found that a resistor, instead of a ferrite bead, does a better job of reducing jitter. The resistor should be kept to a value lowr than 2 Ω. Experimentation is the best way to determine the optimal fileter design for a specific application.

Figure 35: VDDA Filtering



- VDDA filtering for each PLL

  The impedance of the ferrite bead should be much greater than that of the capacitor at frequencies where noise is expected. Many applications have found that a resistor instead of a ferrite bead does a better job of reducing jitter. The resistor should be kept to a value lower than 2 Ω . Experimentation is the best way to determine the optimal filter design for a specific application.

Table 27: Back Pressure Serial Link Signals

Name	Input/Output	Level	Description
START_GCXFR	Output	LVCMOS	Start of transfer of the switch grant information asserted by the converter to the PE to indicate the position of the start of the overall priority cycle in a superframe. The superframe is made of as many frames as there are priorities in use. This signal is Hi-Z when the interface is not selected
ODD_OQG	Output	LVCMOS	Contains the information per priority related to the switch output queues 0-7 and 16-23. This signal is Hi-Z when the interface is not selected
EVEN_OQG	Output	LVCMOS	Contains the information per priority related to the switch output queues 8-15 and 24-31. This signal is Hi-Z when the interface is not selected.
SHARED_GNT	Output	LVCMOS	Indicate in a frame the relevant priority on the ODD/EVEN OQG serial link and carry the global shared memory grant. The last bit of each frame carries the odd parity bit computed on all data bit in the frame. This signal is Hi-Z when the interface is not selected.

Table 28: Miscellaneous External Signals

Name	Input/ Output	Level	Description	Notes
$\overline{\text{POR}}$	Input	LVTTL	Creates a hardware reset of the converter (flush LSSD).	1
$\overline{\text{Slot\_IDN\_4}}$... Slot_IDN_0	Input	LVTTL	Hardware Slot ID. These bits are directly mapped to the SlotID Control Register @A0 bits 20-16.	1
$\overline{\text{APAN\_X}}$	Output	LVTTL	Adapter Port Available (APA) signal sent to the switch board X. This bit is directly mapped, with inversion, from the X Adapter Port Available Register @08 bit 19	
$\overline{\text{APAN\_Y}}$	Output	LVTTL	Adapter Port Available (APA) signal sent to the switch board Y. This bit is directly mapped from the Y Adapter Port Available Register @28 bit 19	
$\overline{\text{FPAN\_X}}$	Input	LVTTL	Fabric Port Available (FPA). This bit is directly mapped to the XFPA Control Register @88 bit 2.	1
$\overline{\text{FPAN\_Y}}$	Input	LVTTL	Fabric Port Available (FPA). This bit is directly mapped to the YFPA Control Register @88 bit 6.	1
$\overline{\text{SWITCH\_X\_IN}}$ $\overline{\text{SERVICE}}$	Input	LVTTL	Fabric X in service selects which plane/clock is currently in use. This bit is directly mapped, with inversion, to the XInService Control Register @88 bit 0.	1
$\overline{\text{SWITCH\_Y\_IN}}$ $\overline{\text{SERVICE}}$	Input	LVTTL	Fabric Y in service selects which plane/clock is currently in use. This bit is directly mapped, with inversion, to the YInService Control Register @88 bit 4.	1
$\overline{\text{SWITCH\_X\_PR}}$ $\overline{\text{ESENT}}$	Input	LVTTL	Fabric X detects that the switch board has been inserted into the backplane and that a switch clock should be present. This bit is directly mapped to the switch present Control Register @88 bit 3.	1
$\overline{\text{SWITCH\_Y\_PR}}$ $\overline{\text{ESENT}}$	Input	LVTTL	Fabric Y detects when a switch board is inserted into the backplane and that a switch clock should be present. This bit is directly mapped to the switch present Control Register @88 bit 7.	1

1. An internal pull-up resistor forces the inactive state.

Figure 36: Switch Present Detection

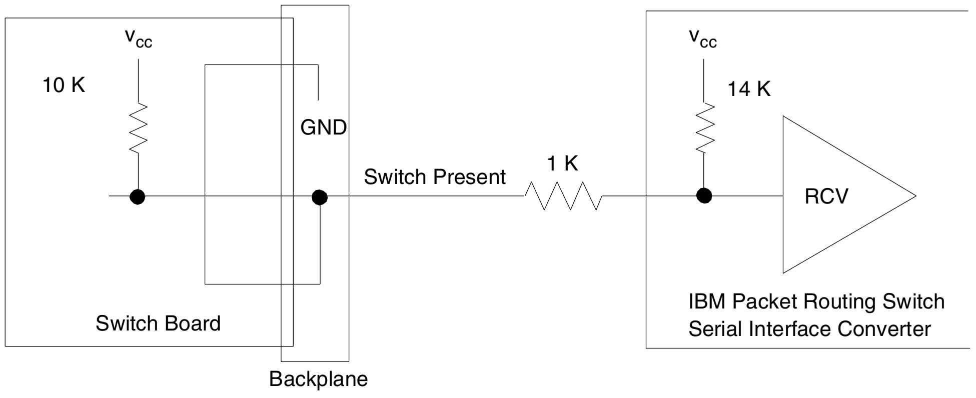


Table 29: Spares Signals Used to Carry Additional DC Voltages

Name	Input/Output	Level	Description
DDV1	Input	VDD1	Voltage I/O connected to VDD1 / 2.5 V
DDV2	Input	VDD1	Voltage I/O connected to VDD2 / 2.5 V
DNG1	Input	GND	Voltage I/O connected to GND
DNG2	Input	GND	Voltage I/O connected to GND

Table 30: Debug Purpose External Signals

Name	Input/Output	Level	Description
AC_TEST_in	Input	LVCMOS	Test I/O used to determine the DASL macro propagation delay
AC_TEST_out	Output	LVCMOS	Test I/O used to determine the DASL macro propagation delay (based on 12 DASL delay lines)
PE_PLLLOCK	Output	LVCMOS	Test I/O used to determine the lock state of the PE PLL
SWITCH_X_PLLLOCK	Output	LVCMOS	Test I/O used to determine the lock state of the switch X PLL
SWITCH_Y_PLLLOCK	Output	LVCMOS	Test I/O used to determine the lock state of the switch Y PLL
TEST_CLK	Output	LVCMOS	External oscillator clock input: 50 - 62.5 MHz
DBG_Bus SELECT_0[15:0] and DBG_Bus SELECT_1[15:0]	Output	LVCMOS	2x 16 bit-buses provide direct I/O access (logic analyzer) to the debug bus specified by the DBG_Select register @8C

Table 31: DBG_SELECT Bus Definition

Bits Reg @8C 7-3	Bits Reg @8C 15-10	Description
00000	00000	Debug bus not used - DBG_DATA bus is tri-stated.
00001	00001	Data movement monitoring Bits Description 15 Switch _Y_FREQOUT bit 1_padout (clockgen island) 14 Switch_X_FREQOUT bit 0_padout (clockgen island) 13-8 Unused 7 Egress Packet clock EDI Y (sop_o_TEBUF) 6 Egress Packet clock EDI X (sop_o_TEBUF) 5 Ingress Packet clock ISEQ Y (pkt_clk_O_TIBUF) 4 Ingress Packet clock ISEQ X (pkt_clk_O_TIBUF) 3 YUMTX: Data packet transmitted indicator at ingress Y switch interface 2 YUMRX: Data packet received indicator at ingress Y switch interface 1 XUMTX: Data packet transmitted indicator at ingress X switch interface 0 XUMRX: Data packet received indicator at ingress X switch interface
00010	00010	M3_debug 0 (16 bits) X Bits Description 15 M3 Oscillator (OSC_125 --->OSC_DIV4 with C_Clock) 14 Reserved 13-11 ALU Status Bits (M3_DEBUG0[2:4]) 10- 0 Program Counter (M3_DEBUG0[5:15])
00011	00011	M3_debug 0 (16 bits) Y Bits Description 15 M3 Oscillator (OSC_125 --->OSC_DIV4 with C_Clock) 14 Reserved 13-11 ALU Status Bits (M3_DEBUG0[2:4]) 0-10 Program Counter (M3_DEBUG0[5:15])
00100	00100	IDI to DASL X Bits Description 15 C_Clock 125 14 Packet clock ISEQ (Pkt_clk_0_TIBUF) 13-8 Unused 0-7 Parallel_Data_in master LU (DASL_DATA_IN_DEBUG)
00101	00101	IDI to DASL Y Bits Description 15 C_Clock 125 14 Packet clock ISEQ (SOP_O_TEBUF) 13-8 Unused 0-7 Parallel_Data_in master LU (DASL_DATA_IN_DEBUG)
00110	00110	DASL to EDI X Bits Description 15 C_Clock 125 14 Packet clock ISEQ (Synchro_i_fdasl) 13-8 Unused 0-7 Parallel_Data_Out master LU (DASL_DATA_OUT_DEBUG)
00111	00111	DASL to EDI Y Bits Description 15 C_Clock 125 14 Packet clock ISEQ (SOP_0_TEBUF) 13-8 Unused 0-7 Parallel_Data_Out master LU (DASL_DATA_OUT_DEBUG)

Table 31: DBG_SELECT Bus Definition

Bits Reg @8C 7-3	Bits Reg @8C 15-10	Description
01000	01000	Output Queue Grant priority 0 X Bits Description 15-0 GRANTP0_O [15:0]
01001	01001	Output Queue Grant priority 1 X Bits Description 15-0 GRANTP1_O [15:0]
01010	01010	Output Queue Grant priority 2 X Bits Description 15-0 GRANTP2_O [15:0]
01011	01011	Output Queue Grant priority 3 X Bits Description 15-0 GRANTP3_O [15:0]
01100	01100	Output Queue Grant Priority 0 Y Bits Description 15-0 GRANTP0_O [15:0]
01101	01101	Output Queue Grant Priority 1 Y Bits Description 15-0 GRANTP_1O[15:0]
01110	01110	Output Queue Grant Priority 2 Y Bits Description 15-0 GRANTP2_O [15:0]
01111	01111	Output Queue Grant Priority 3 Y Bits Description 15-0 GRANTP3_O [15:0]
10000	10000	Receive Framing in Bits Description 15 URXCLK 14 Start_RXxfr 13 RxPRTY_error 12 Rxdata_valid 11-8 Unused 7-0 RXDATA_L_Out [31:24] (8 bits of master LU)
10001	10001	Receive Framing Out Bits Description 15 URXCLK 14 SOP_to_RXfifo 13 VALID_to_RXfifo 12-8 Unused 7-0 RXframed_data_L [31:24] (8 bits of master LU)
10010	10010	Receive FIFO in Bits Description 15 URXCLK 14 SOP_to_RXfifo 13 VALID_to_RXfifo 12-8 Unused 7-0 RXframed_data_L [31:24] (8 bits of master LU)

Table 31: DBG_SELECT Bus Definition

Bits Reg @8C 7-3	Bits Reg @8C 15-10	Description
10011	10011	RXFIFO out X Bits Description 15 DASLCLK_X 14 fifo_empty 13 fifo_almost_empty 12 fifo_full 11 fifo_almost_full 10 ACK_to_RXdasl 9 SOP_from_RXdasl 8 REQ_from_RXdasl 7-0 RXfifoed_data_L [31:24] (8 bits of master LU)
10100	10100	RXFIFO out Y Bits Description 15 DASLCLK_Y 14 fifo_empty 13 fifo_almost_empty 12 fifo_full 11 fifo_almost_full 10 ACK_to_RXdasl 9 SOP_from_RXdasl 8 REQ_from_RXdasl 7-0 RXfifoed_data_L [31:24] (8 bits of master LU)
10101	10101	TXFIFO path X Bits Description 15 DASLCLK_X 14 SOP_from_TXdasl 13 fifo_empty 12 fifo_almost_empty 11 fifo_full 10 fifo_almost_full 9 VALID_from_TXdasl 8 Unused 7-0 DATA_from_Filter [31:24] (8 bits master LU)
10110	10110	TXFIFO path Y Bits Description 15 DASLCLK_Y 14 SOP_from_TXdasl 13 fifo_empty 12 fifo_almost_empty 11 fifo_full 10 fifo_almost_full 9 VALID_from_TXdasl 8 Unused 7-0 DATA_from_Filter [31:24] (8 bits master LU)

Table 31: DBG_SELECT Bus Definition

Bits Reg @8C 7-3	Bits Reg @8C 15-10	Description
10111	10111	TxFraming in (from FIFO X) Bits Description 15 UTXCLK 14 SOP_to_TXfifo_x 13 Hold_toTXfifo_x 12 FAF_from_TXfifo-x 11 VALID_from_TXfifo_x 10 REQ_to_RXfifo_x 9-8 Unused 7-0 DATA_from_TXfifo_x [31:24] (8 bits master LU)
11000’	11000’	TxFraming in (from FIFO Y) Bits Description 15 UTXCLK 14 SOP_to_TXfifo_y 13 Hold_toTXfifo_y 12 FAF_from_TXfifo-y 11 VALID_from_TXfifo_y 10 REQ_to_RXfifo_y 9-8 Unused 7-0 DATA_from_TXfifo_y [31:24] (8 bits master LU)
11001’	11001’	TxFraming towards PE interface Bits Description 15 UTXCLK 14 Unused 13 HOLD_xfr_from_TXint 12 PENDING_from_TXframing 10-8 Unused?? 7-0 DATA_from_TXframing [31:24] (8 bits master LU)
11010’	11010’	UTOPIA-3 like ingress interface Bits Description 15 URXCLK 14 RXSOP 13 $\overline{\text{RXFULL}}$ 12 $\overline{\text{RXENB}}$ 10-8 Unused 7-0 RXDATA [31:24] (8 bits master LU)
11011’	11011’	UTOPIA-3 like Egress interface Bits Description 15 UTXCLK 14 TXSOP 13 $\overline{\text{TXFULL}}$ 12 $\overline{\text{TXENB}}$ 10-8 Unused 7-0 TX DATA [31:24] (8 bits master LU)

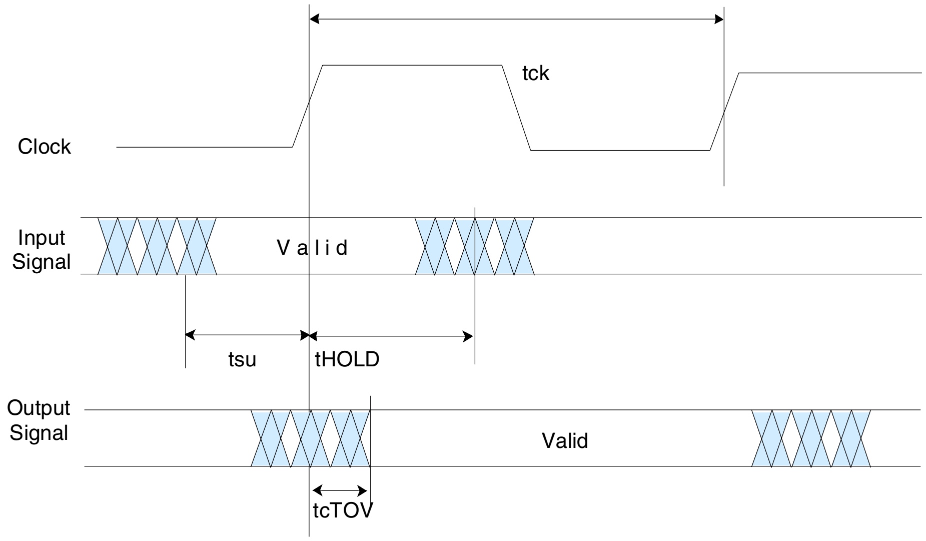
7.2 I/O Timing

7.2.1 IBM Packet Routing Switch Serial Interface Converter A.C. Characteristics

7.2.1.1 AC parameters characteristics

The converter's external interfaces (UTOPIA-3 like, Processor, and DASL) specify that all inputs and outputs are registered. Only Set-up times and Hold times for inputs and Clock-to-outputs times for Outputs are needed.

Figure 37: AC parameters timing diagram

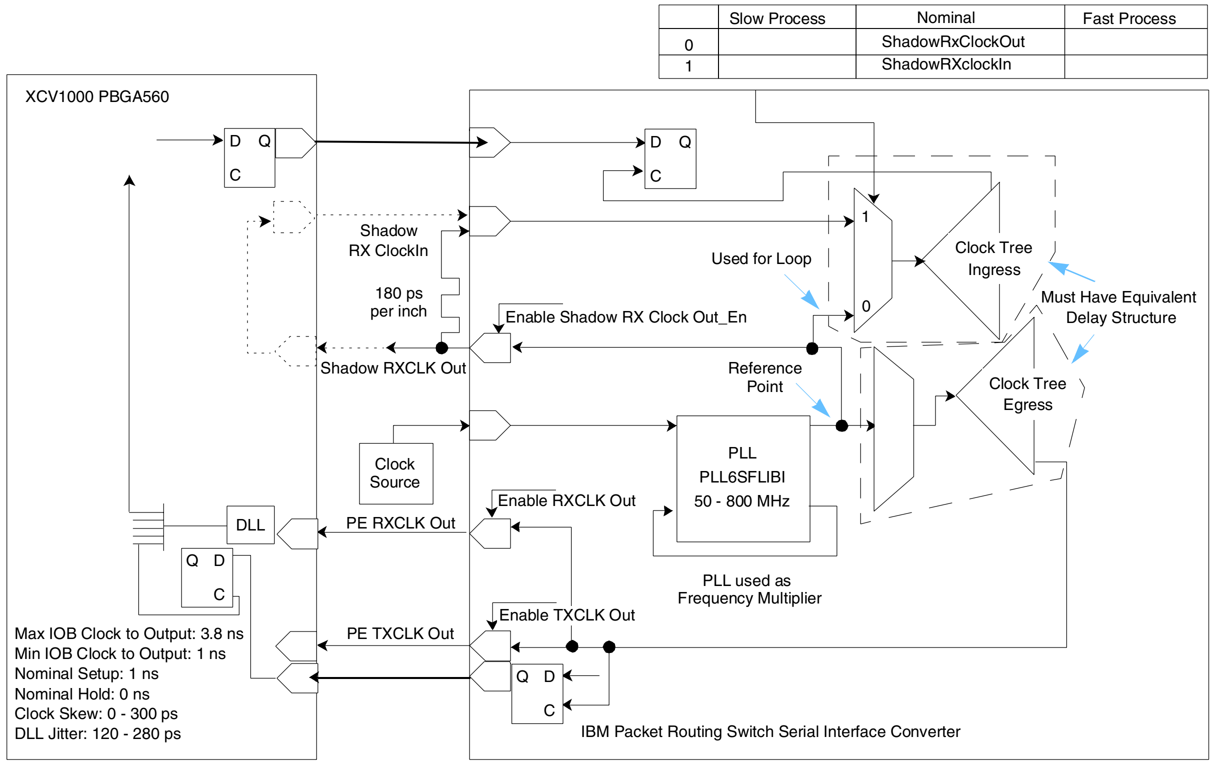


7.2.1.2 Protocol Engine (UTOPIA-3 like) Interface A.C. Specifications

Symbol	Parameter	Test condition	Min	Typical	Max	Unit
f	clock frequency		50	110	125	MHz
tck	clock period		8	9	20	ns
tpw	Clock Pulse Width	High or Low	3.3			ns
tcTOV	Clock-to-Output valid	xx pF, all outputs	0.8		5	ns
tsu	Input Set-up Time	All inputs	2			ns
tHOLD	Input Hold Time	All inputs	0			ns

The following drawing shows the flexibility in implementing the UTOPIA-3 like interface timing.

Figure 38: Options for Ingress UTOPIA-3 Like Interface Clocking



	Slow Process	Nominal	Fast Process
0		ShadowRxClockOut	
1		ShadowRXclockIn	

7.2.1.3 Microprocessor Interface A.C. Specifications

Symbol	Parameter	Test Condition	Min	Typical	Max	Unit
f	Clock Frequency		25	50	66	MHz
tck	Clock Period		15	20	40	ns
tpw	Clock Pulse Width	High or Low				ns
TcTOV	Clock to Output valid	xx pF, all outputs			10	ns
tsu	Input Set-up Time	All inputs	2			ns
tHOLD	Input Hold Time	All inputs	0			ns

8. Electrical Specifications

Table 32: Absolute Maximum Ratings

Parameter	Description	Min	Typ	Max	Units
V_{DD}	Supply voltage VDD	-0.5	2.5	2.7	V
V_{IN}	Input voltage	-0.5		$V_{DD+0.6}$	V
V_{OUT}	Output voltage	-0.5		$V_{DD+0.6}$	V
	Thermal impedance junction to ambient package Airflow=0		14.7		°C/W
	Thermal impedance junction to ambient package Airflow=100FPM		13.3		°C/W
	Thermal impedance junction to ambient package Airflow=200FPM		11.9		°C/W
	Thermal impedance junction to case package		1.9		°C/W
T_S	Storage temperature	-65		150	°C
T_A	Operating junction temperature range	0		125	°C
	Electrostatic discharge	-3.000	6,000	3,000	V

Note: Permanent device damage may occur if the above absolute maximum ratings are exceeded. Functional operation should be restricted to the conditions as detailed in the operational section of this data sheet. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

8.1 Power Sequencing

In order to prevent latchup (and destruction) of the chip, the power supplies must be sequenced up and down in a manner that any supply is greater than or equal to the voltage of another supply of lesser value both during "turn on" and "turn off" (including a quick toff - ton sequence).

For example, 3.3 V >/= 2.5 >/= 1.5 V during power up and power down. Actually up to 400 mV of negative voltage can be tolerated during the sequence between any two supplies. There is no time delay requirement, only a negative voltage restriction. Ideally, all supplies would rise together until they reach their operating level and all would fall together until they reach zero.

If one could design a 400 mV (max) diode and place it between 3.3 V (cathode) and 2.5 V (anode), and another between 2.5 V (cathode) and 1.5 V (anode), that would ensure that the 400mV diodes internal to the chip did not get forward biased more than the 400 mV, which would keep the chip out of latchup conditions.

Also, the 3.3 V supply should not exceed the 2.5 V supply by more than 2.7 V during power up/down, and 1.3V in constant use.

8.2 Recommended Operating Conditions

Table 33: LVCMOS Compatible I/Os

Parameters	Symbol	Value		
		Min	Typ	Max
Supply voltage	V_{DD}	2.3 V	2.5 V	2.7 V
Input Up Level	V_{IH}		1.4 V	1.7 V
Input Down Level	V_{IL}	0.7 V	1.4 V	
High Level Output Voltage (V_{DD}=Min, I_{OH}=-4mA)	V_{OH}	2 V	2.2 V	
Low Level Output Voltage (V_{DD}=Min, I_{OL}=6mA I_{OL}=8mA	V_{OL}	Gnd	0.2 V	0.4 V
2.5 V CMOS Driver Minimum DC Currents at Rated Voltage (Vdd at 2.3V, temperature at 100°C) Vlow=2.0 V	I_{High}	7/18 mA		
2.5 V CMOS Driver Minimum DC Currents at Rated Voltage (Vdd at 2.3 V, temperature at 100°C) Vlow=0.4 V	I_{Low}	9/18 mA		
Receiver maximum input leakage Low Level Input Current at LPDL	I_{IL}	No pull up /pull down with pull-down with pull-up	-	0µA 0µA -250µA
Receiver maximum input leakage High Level Input Current at MPUL	I_{IH}	No pull up /pull down with pull-down with pull-up		0µA 400µA 0µA
Input capacitance V_{DD}=Nom	C_I			5 pF

Table 34: LVTTL Compatible I/Os

Parameters	Symbol	Value		
		Min	Typ	Max
Supply voltage	V_{DD4}	3 V	3.3 V	3.6 V
Input Up Level	V_{IH}		1.4 V	2 V
Input Down Level	V_{IL}	0.8 V	1.4 V	
High Level Output Voltage (V_{DD}=Min, I_{OH}=-4mA)	V_{OH}	2.4 V	2.7 V	V_{DD4}
Low Level Output Voltage (V_{DD}=Min, I_{OL}=6mA I_{OL}=8mA	V_{OL}	Gnd	0.25 V	0.4 V
3.3 V LVTTL Driver Minimum DC Currents at Rated Voltage (Vdd at 3 V, temperature at 100°C) Vlow=2.0 V	I_{High}	12/19 mA		
3.3 V LVTTL Driver Minimum DC Currents at Rated Voltage (Vdd at 3 V, temperature at 100°C) Vlow=0.4 V	I_{Low}	8/19 mA		
Receiver maximum input leakage Low Level Input Current at LPDL	I_{IL}	No pull-up /pull-down with pull-down with pull-up	-	0µA 0µA -250µA
Receiver maximum input leakage High Level Input Current at MPUL	I_{IH}	No pull up /pull down with pull-down with pull-up		0µA 400µA 0µA
Input capacitance V_{DD}=Nom	C_I			5 pF

Table 35: Recommended Operating Conditions for all I/Os

Parameter	Ref Signal	Value			Unit
		Min	Typ	Max	
Rising transition rate of the output	DASL Driver	0.14	0.21	.038	ns
Falling transition rate of the output	DASL Driver	0.19	0.28	0.33	ns
Max Input Pin Cap	DASL Receiver	2.5			pF

Table 36: Power Dissipation

Supply	Power	Current
Total 3.3 V (nominal)	0.04 W	12 mA
Total 3.3 V (maximum)	0.05 W	14 mA
Total 2.5 V (nominal)	2.05 W	820 mA
Total 2.5 V (maximum)	2.39 W	880 mA
Total 1.5 V (nominal)	0.34 W	226 mA
Total 1.5 V (maximum)	0.4 W	261 mA
Total Chip (nominal)	2.43 W	
Total Chip (maximum)	2.84 W	

8.3 Signal Pin Assignments

Table 37: Signal Pins Sorted by Grid Location (Page 1 of 4)

Grid Location	Pin Name	Grid Location	Pin Name
A02	TXDATA_2	C09	TXDATA_18
A03	DI2	C10	DASL_Y_TX3
A04	TXDATA_9	C11	SHARED_GNT
A05	DI1	C12	TXSOP
A06	TXDATA_13	C13	DASL_Y_TX6
A07	TXDATA_16	C14	TXDATA_31
A08	DASL_Y_TX0	C15	DASL_Y_TX7
A09	DASL_Y_TX1	C16	DDV_2
A10	$\overline{\text{DASL\_Y\_TX3}}$	C17	DASL_Y_RX1
A11	$\overline{\text{DASL\_Y\_TX5}}$	C18	DBG_BusSELECT_0_1
A12	TXDATA_27	C19	SWITCH_Y_TESTIN
A13	$\overline{\text{DASL\_Y\_TX6}}$	D01	Shadow_RXclock_Out
A14	TXDATA_29	D03	MEM_GNT_Y_3
A15	$\overline{\text{DASL\_Y\_TX7}}$	D05	TXDATA_5
A16	DASL_Y_RX0	D07	START_GCXFR
A17	$\overline{\text{DASL\_Y\_RX1}}$	D09	TXDATA_14
A18	$\overline{\text{DASL\_Y\_RX2}}$	D11	TXDATA_24
A19	DASL_Y_RX3	D13	TXDATA_30
B01	$\overline{\text{SWITCH\_X\_PRESENT}}$	D15	DBG_BusSELECT_0_8
B03	TXDATA_3	D17	DASL_Y_RX4
B05	TXDATA_10	D19	$\overline{\text{DASL\_Y\_RX4}}$
B07	TXDATA_17	E01	PE_PLLLOCK
B09	$\overline{\text{DASL\_Y\_TX1}}$	E02	MP_ADDR_0
B11	DASL_Y_TX5	E03	PE_RXCLK_out
B13	TXDATA_25	E04	MEM_GNT_Y_1
B15	$\overline{\text{DASL\_Y\_RX0}}$	E05	TXDATA_0
B17	DASL_Y_RX2	E06	TXDATA_12
B19	$\overline{\text{DASL\_Y\_RX3}}$	E07	ODD_OQG
C01	PE_TESTOUT	E08	$\overline{\text{TXFULL}}$
C02	MEM_GNT_Y_2	E09	TXDATA_7
C03	PE_TESTIN	E10	DASL_Y_TX2
C04	TXDATA_4	E11	DASL_Y_TX4
C05	RI	E12	$\overline{\text{TXENB}}$
C06	TXDATA_11	E13	TXDATA_19
C07	TXDATA_1	E14	TXDATA_21
C08	$\overline{\text{DASL\_Y\_TX0}}$	E15	ABIST_CLK

Table 37: Signal Pins Sorted by Grid Location (Page 2 of 4)

Grid Location	Pin Name	Grid Location	Pin Name
E16	DBG_BusSELECT_0_0	H09	TXDATA_15
E17	SWITCH_Y_TESTOUT	H11	TXDATA_23
E18	DBG_BusSELECT_0_2	H13	DBG_BusSELECT_0_9
E19	SWITCH_Y_PLLLOCK	H15	DBG_BusSELECT_0_15
F01	MP_DATA_2	H17	$\overline{\text{DASL\_Y\_RX5}}$
F03	Shadow_RXclock_In	H19	DASL_Y_RX5
F05	MP_DATA_1	J01	Slot_IDN_2
F07	TXDATA_8	J02	Slot_IDN_1
F09	TXPRTY	J03	Slot_IDN_3
F11	TXDATA_28	J04	Slot_IDN_4
F13	TXDATA_22	J05	ABIST_DIAGOUT
F15	DBG_BusSELECT_0_6	J06	MEM_GNT_Y_0
F17	DBG_BusSELECT_0_3	J07	TDI
F19	DBG_BusSELECT_0_11	J08	MP_ADDR_1
G01	PE_CLK	J12	DBG_BusSELECT_0_14
G02	MP_ADDR_2	J13	CE1_A
G03	TMS	J14	DBG_BusSELECT_0_5
G04	MP_DATA_PRTY	J15	CE1_C2
G05	MP_DATA_4	J16	DBG_BusSELECT_0_12
G06	$\overline{\text{DASL\_DRV\_ENB}}$	J17	$\overline{\text{DASL\_Y\_RX6}}$
G07	Slot_IDN_0	J18	SWITCH_Y_VDDA
G08	EVEN_OQG	J19	DASL_Y_RX6
G09	TXDATA_6	K01	PE_VDDA
G10	$\overline{\text{DASL\_Y\_TX2}}$	K03	MP_DATA_5
G11	$\overline{\text{DASL\_Y\_TX4}}$	K05	MP_ADDR_4
G12	TXDATA_20	K07	MP_ADDR_5
G13	TXDATA_26	K13	DBG_BusSELECT_1_2
G14	DBG_BusSELECT_0_10	K15	DNG_1
G15	DBG_BusSELECT_0_13	K17	$\overline{\text{DASL\_Y\_RX7}}$
G16	DBG_BusSELECT_0_4	K19	DASL_Y_RX7
G17	$\overline{\text{SWITCH\_Y\_CLK}}$	L01	MP_DATA_7
G18	DBG_BusSELECT_0_7	L02	MP_DATA_6
G19	SWITCH_Y_CLK	L03	MP_ADDR_6
H01	MP_DATA_3	L04	$\overline{\text{MP\_INT}}$
H03	MP_PRDY	L05	TCK
H05	MP_ADDR_3	L06	TDO
H07	MP_DATA_0	L07	TRST

Table 37: Signal Pins Sorted by Grid Location (Page 3 of 4)

Grid Location	Pin Name	Grid Location	Pin Name
L08	MEM_GNT_X_3	N19	$\overline{\text{SWITCH\_X\_CLK}}$
L12	DBG_BusSELECT_1_1	P01	APAN_X
L13	TEST_CLK	P03	TO_SMOOTH_PLL_IN
L14	DBG_BusSELECT_1_9	P05	$\overline{\text{SWITCH\_Y\_PRESENT}}$
L15	CE1_C1	P07	RXDATA_31
L16	DBG_BusSELECT_1_7	P09	RXDATA_28
L17	DASL_X_RX0	P11	RXDATA_5
L18	SWITCH_X_VDDA	P13	DDV_1
L19	$\overline{\text{DASL\_X\_RX0}}$	P15	DBG_BusSELECT_1_11
M01	MEM_GNT_X_0	P17	DASL_X_RX2
M03	FPAN_X	P19	$\overline{\text{DASL\_X\_RX2}}$
M05	MEM_GNT_X_1	R01	TEST_B2
M07	MP_BURST_MODE	R02	MP_ADDR_7
M09	DNG_2	R03	TEST_C3
M11	RXDATA_13	R04	FROM_SMOOTH_PLL_OUT
M13	DBG_BusSELECT_1_6	R05	PE_TXCLK_out
M15	DBG_BusSELECT_1_0	R06	RXDATA_26
M17	DASL_X_RX1	R07	RXDATA_27
M19	$\overline{\text{DASL\_X\_RX1}}$	R08	RXDATA_16
N01	$\overline{\text{MP\_SEL}}$	R09	RXDATA_8
N02	MP_PRTY_ENB	R10	DASL_X_TX7
N03	$\overline{\text{MP\_WR}}$	R11	$\overline{\text{DASL\_X\_TX5}}$
N04	MEM_GNT_X_2	R12	$\overline{\text{DASL\_X\_TX3}}$
N05	SEND_GNT_X	R13	DASL_X_TX2
N06	FPAN_Y	R14	DBG_BusSELECT_1_3
N07	MP_ADD_PRTY	R15	CE1_B
N08	RXDATA_30	R16	DBG_BusSELECT_1_12
N09	RXDATA_9	R17	SWITCH_X_TESTIN
N10	$\overline{\text{DASL\_X\_TX7}}$	R18	$\overline{\text{DASL\_X\_RX3}}$
N11	DASL_X_TX5	R19	SWITCH_X_PLLLOCK
N12	$\overline{\text{DASL\_X\_TX2}}$	T01	$\overline{\text{POR}}$
N13	RXDATA_6	T03	$\overline{\text{SWITCH\_X\_INSERVICE}}$
N14	DBG_BusSELECT_1_5	T05	$\overline{\text{RXENB}}$
N15	DBG_BusSELECT_1_10	T07	RXDATA_21
N16	DBG_BusSELECT_1_8	T09	RXDATA_24
N17	SWITCH_X_CLK	T11	RXDATA_12
N18	DBG_BusSELECT_1_4	T13	RXDATA_2

Table 37: Signal Pins Sorted by Grid Location (Page 4 of 4)

Grid Location	Pin Name	Grid Location	Pin Name
T15	RXDATA_0	W06	RXDATA_25
T17	DBG_BusSELECT_1_14	W07	CE0_Scan
T19	DASL_X_RX3	W08	RXDATA_23
U01	MP_CLK	W09	RXDATA_18
U02	SEND_GNT_Y	W10	DASL_X_TX6
U03	RXDATA_14	W11	DASL_X_TX4
U04	RXPAV	W12	RXDATA_7
U05	CE0_TEST	W13	DASL_X_TX1
U06	AC_TEST_in	W14	RXDATA_4
U07	RXDATA_3	W15	DASL_X_TX0
U08	RXDATA_22	W16	$\overline{\text{DASL\_X\_RX7}}$
U09	RXDATA_19	W17	DASL_X_RX6
U10	$\overline{\text{DASL\_X\_TX6}}$	W18	DASL_X_RX5
U11	RXDATA_10	W19	DASL_X_RX4
U12	DASL_X_TX3		
U13	$\overline{\text{DASL\_X\_TX1}}$		
U14	RXDATA_1		
U15	$\overline{\text{DASL\_X\_TX0}}$		
U16	DBG_BusSELECT_1_15		
U17	$\overline{\text{DASL\_X\_RX6}}$		
U18	DBG_BusSELECT_1_13		
U19	SWITCH_X_TESTOUT		
V01	$\overline{\text{SWITCH\_Y\_INSERVICE}}$		
V03	RXPRTY		
V05	AC_TEST_out		
V07	RXDATA_20		
V09	RXDATA_17		
V11	$\overline{\text{DASL\_X\_TX4}}$		
V13	RXDATA_11		
V15	DASL_X_RX7		
V17	$\overline{\text{DASL\_X\_RX5}}$		
V19	$\overline{\text{DASL\_X\_RX4}}$		
W01	APAN_Y		
W02	RXSOP		
W03	RXDATA_15		
W04	RXDATA_29		
W05	CE0_IO		

Table 38: Signal Pins Sorted By Signal Name (Page 1 of 4)

Pin Name	Grid Location	Pin Name	Grid Location
ABIST_CLK	E15	DASL_X_TX3	U12
ABIST_DIAGOUT	J05	$\overline{\text{DASL\_X\_TX3}}$	R12
AC_TEST_in	U06	DASL_X_TX4	W11
AC_TEST_out	V05	$\overline{\text{DASL\_X\_TX4}}$	V11
APAN_X	P01	DASL_X_TX5	N11
APAN_Y	W01	$\overline{\text{DASL\_X\_TX5}}$	R11
CE0_IO	W05	DASL_X_TX6	W10
CE0_Scan	W07	$\overline{\text{DASL\_X\_TX6}}$	U10
CE0_TEST	U05	DASL_X_TX7	R10
CE1_A	J13	$\overline{\text{DASL\_X\_TX7}}$	N10
CE1_B	R15	DASL_Y_RX0	A16
CE1_C1	L15	$\overline{\text{DASL\_Y\_RX0}}$	B15
CE1_C2	J15	DASL_Y_RX1	C17
$\overline{\text{DASL\_DRV\_ENB}}$	G06	$\overline{\text{DASL\_Y\_RX1}}$	A17
DASL_X_RX0	L17	DASL_Y_RX2	B17
$\overline{\text{DASL\_X\_RX0}}$	L19	$\overline{\text{DASL\_Y\_RX2}}$	A18
DASL_X_RX1	M17	DASL_Y_RX3	A19
$\overline{\text{DASL\_X\_RX1}}$	M19	$\overline{\text{DASL\_Y\_RX3}}$	B19
DASL_X_RX2	P17	DASL_Y_RX4	D17
$\overline{\text{DASL\_X\_RX2}}$	P19	$\overline{\text{DASL\_Y\_RX4}}$	D19
DASL_X_RX3	T19	DASL_Y_RX5	H19
$\overline{\text{DASL\_X\_RX3}}$	R18	$\overline{\text{DASL\_Y\_RX5}}$	H17
DASL_X_RX4	W19	DASL_Y_RX6	J19
$\overline{\text{DASL\_X\_RX4}}$	V19	$\overline{\text{DASL\_Y\_RX6}}$	J17
DASL_X_RX5	W18	DASL_Y_RX7	K19
$\overline{\text{DASL\_X\_RX5}}$	V17	$\overline{\text{DASL\_Y\_RX7}}$	K17
DASL_X_RX6	W17	DASL_Y_TX0	A08
$\overline{\text{DASL\_X\_RX6}}$	U17	$\overline{\text{DASL\_Y\_TX0}}$	C08
DASL_X_RX7	V15	DASL_Y_TX1	A09
$\overline{\text{DASL\_X\_RX7}}$	W16	$\overline{\text{DASL\_Y\_TX1}}$	B09
DASL_X_TX0	W15	DASL_Y_TX2	E10
$\overline{\text{DASL\_X\_TX0}}$	U15	$\overline{\text{DASL\_Y\_TX2}}$	G10
DASL_X_TX1	W13	DASL_Y_TX3	C10
$\overline{\text{DASL\_X\_TX1}}$	U13	$\overline{\text{DASL\_Y\_TX3}}$	A10
DASL_X_TX2	R13	DASL_Y_TX4	E11
$\overline{\text{DASL\_X\_TX2}}$	N12	$\overline{\text{DASL\_Y\_TX4}}$	G11

Table 38: Signal Pins Sorted By Signal Name (Page 2 of 4)

Pin Name	Grid Location	Pin Name	Grid Location
DASL_Y_TX5	B11	DBG_BusSELECT_1_9	L14
$\overline{\text{DASL\_Y\_TX5}}$	A11	DDV_1	P13
DASL_Y_TX6	C13	DDV_2	C16
$\overline{\text{DASL\_Y\_TX6}}$	A13	DI1	A05
DASL_Y_TX7	C15	DI2	A03
$\overline{\text{DASL\_Y\_TX7}}$	A15	DNG_1	K15
DBG_BusSELECT_0_0	E16	DNG_2	M09
DBG_BusSELECT_0_10	G14	EVEN_OQG	G08
DBG_BusSELECT_0_11	F19	FPAN_X	M03
DBG_BusSELECT_0_12	J16	FPAN_Y	N06
DBG_BusSELECT_0_13	G15	FROM_SMOOTH_PLL_OUT	R04
DBG_BusSELECT_0_14	J12	MEM_GNT_X_0	M01
DBG_BusSELECT_0_15	H15	MEM_GNT_X_1	M05
DBG_BusSELECT_0_1	C18	MEM_GNT_X_2	N04
DBG_BusSELECT_0_2	E18	MEM_GNT_X_3	L08
DBG_BusSELECT_0_3	F17	MEM_GNT_Y_0	J06
DBG_BusSELECT_0_4	G16	MEM_GNT_Y_1	E04
DBG_BusSELECT_0_5	J14	MEM_GNT_Y_2	C02
DBG_BusSELECT_0_6	F15	MEM_GNT_Y_3	D03
DBG_BusSELECT_0_7	G18	MP_ADD_PRTY	N07
DBG_BusSELECT_0_8	D15	MP_ADDR_0	E02
DBG_BusSELECT_0_9	H13	MP_ADDR_1	J08
DBG_BusSELECT_1_0	M15	MP_ADDR_2	G02
DBG_BusSELECT_1_10	N15	MP_ADDR_3	H05
DBG_BusSELECT_1_11	P15	MP_ADDR_4	K05
DBG_BusSELECT_1_12	R16	MP_ADDR_5	K07
DBG_BusSELECT_1_13	U18	MP_ADDR_6	L03
DBG_BusSELECT_1_14	T17	MP_ADDR_7	R02
DBG_BusSELECT_1_15	U16	MP_BURST_MODE	M07
DBG_BusSELECT_1_1	L12	MP_CLK	U01
DBG_BusSELECT_1_2	K13	MP_DATA_0	H07
DBG_BusSELECT_1_3	R14	MP_DATA_1	F05
DBG_BusSELECT_1_4	N18	MP_DATA_2	F01
DBG_BusSELECT_1_5	N14	MP_DATA_3	H01
DBG_BusSELECT_1_6	M13	MP_DATA_4	G05
DBG_BusSELECT_1_7	L16	MP_DATA_5	K03
DBG_BusSELECT_1_8	N16	MP_DATA_6	L02

Table 38: Signal Pins Sorted By Signal Name (Page 3 of 4)

Pin Name	Grid Location	Pin Name	Grid Location
MP_DATA_7	L01	RXDATA_28	P09
MP_DATA_PRTY	G04	RXDATA_29	W04
$\overline{\text{MP\_INT}}$	L04	RXDATA_2	T13
MP_PRDY	H03	RXDATA_30	N08
MP_PRTY_ENB	N02	RXDATA_31	P07
$\overline{\text{MP\_SEL}}$	N01	RXDATA_3	U07
$\overline{\text{MP\_WR}}$	N03	RXDATA_4	W14
ODD_OQG	E07	RXDATA_5	P11
PE_CLK	G01	RXDATA_6	N13
PE_PLLLOCK	E01	RXDATA_7	W12
PE_RXCLK_out	E03	RXDATA_8	R09
PE_TESTIN	C03	RXDATA_9	N09
PE_TESTOUT	C01	$\overline{\text{RXENB}}$	T05
PE_TXCLK_out	R05	RXPAV	U04
PE_VDDA	K01	RXPRTY	V03
$\overline{\text{POR}}$	T01	RXSOP	W02
RI	C05	SEND_GNT_X	N05
RXDATA_0	T15	SEND_GNT_Y	U02
RXDATA_10	U11	Shadow_RXclock_In	F03
RXDATA_11	V13	Shadow_RXclock_Out	D01
RXDATA_12	T11	SHARED_GNT	C11
RXDATA_13	M11	Slot_IDN_0	G07
RXDATA_14	U03	Slot_IDN_1	J02
RXDATA_15	W03	Slot_IDN_2	J01
RXDATA_16	R08	Slot_IDN_3	J03
RXDATA_17	V09	Slot_IDN_4	J04
RXDATA_18	W09	START_GCXFR	D07
RXDATA_19	U09	SWITCH_X_CLK	N17
RXDATA_1	U14	$\overline{\text{SWITCH\_X\_CLK}}$	N19
RXDATA_20	V07	$\overline{\text{SWITCH\_X\_INSERVICE}}$	T03
RXDATA_21	T07	SWITCH_X_PLLLOCK	R19
RXDATA_22	U08	$\overline{\text{SWITCH\_X\_PRESENT}}$	B01
RXDATA_23	W08	SWITCH_X_TESTIN	R17
RXDATA_24	T09	SWITCH_X_TESTOUT	U19
RXDATA_25	W06	SWITCH_X_VDDA	L18
RXDATA_26	R06	SWITCH_Y_CLK	G19
RXDATA_27	R07	$\overline{\text{SWITCH\_Y\_CLK}}$	G17

Table 38: Signal Pins Sorted By Signal Name (Page 4 of 4)

Pin Name	Grid Location	Pin Name	Grid Location
SWITCH_Y_INSERVICE	V01	TXDATA_2	A02
SWITCH_Y_PLLLOCK	E19	TXDATA_30	D13
SWITCH_Y_PRESENT	P05	TXDATA_31	C14
SWITCH_Y_TESTIN	C19	TXDATA_3	B03
SWITCH_Y_TESTOUT	E17	TXDATA_4	C04
SWITCH_Y_VDDA	J18	TXDATA_5	D05
TCK	L05	TXDATA_6	G09
TDI	J07	TXDATA_7	E09
TDO	L06	TXDATA_8	F07
TEST_B2	R01	TXDATA_9	A04
TEST_C3	R03	TXENB	E12
TEST_CLK	L13	TXFULL	E08
TMS	G03	TXPRTY	F09
TO_SMOOTH_PLL_IN	P03	TXSOP	C12
TRST	L07		
TXDATA_0	E05		
TXDATA_10	B05		
TXDATA_11	C06		
TXDATA_12	E06		
TXDATA_13	A06		
TXDATA_14	D09		
TXDATA_15	H09		
TXDATA_16	A07		
TXDATA_17	B07		
TXDATA_18	C09		
TXDATA_19	E13		
TXDATA_1	C07		
TXDATA_20	G12		
TXDATA_21	E14		
TXDATA_22	F13		
TXDATA_23	H11		
TXDATA_24	D11		
TXDATA_25	B13		
TXDATA_26	G13		
TXDATA_27	A12		
TXDATA_28	F11		
TXDATA_29	A14		

8.4 Power Signals

Table 39: Ground Signals

Sequence Number	Grid Location	Sequence Number	Grid Location
1	B02	37	T04
2	B06	38	T08
3	B10	39	T12
4	B14	40	T16
5	B18	41	V02
6	D04	42	V06
7	D08	43	V10
8	D12	44	V14
9	D16	45	V18
10	F02	36	P18
11	F06		
12	F10		
13	F14		
14	F18		
15	H04		
16	H08		
17	H12		
18	H16		
19	J10		
20	K02		
21	K06		
22	K09		
23	K10		
24	K11		
25	K14		
26	K18		
27	L10		
28	M04		
29	M08		
30	M12		
31	M16		
32	P02		
33	P06		
34	P10		
35	P14		

Table 40: 2.5 V V_{DD} Signals

Sequence Number	Grid Location
1	B04
2	B12
3	D10
4	D18
5	F08
6	H02
7	H10
8	H14
9	J09
10	J11
11	K04
12	K08
13	K12
14	K16
15	L09
16	L11
17	M06
18	M10
19	M18
20	P12
21	T02
22	T10
23	V08
24	V16

Table 41: 1.5 V South VDD 2 Signals

Sequence Number	I/O Location
1	F16
2	H18
3	M14
4	P16
5	T18

Table 42: 1.5 V East VDD 3 Signals

Sequence Number	I/O Location
1	P08
2	T06
3	T14
4	V04
5	V12

Table 43: 3.3 V North VDD 4 Signals

Sequence Number	I/O Location
1	D02
2	F04
3	H06
4	M02
5	P04

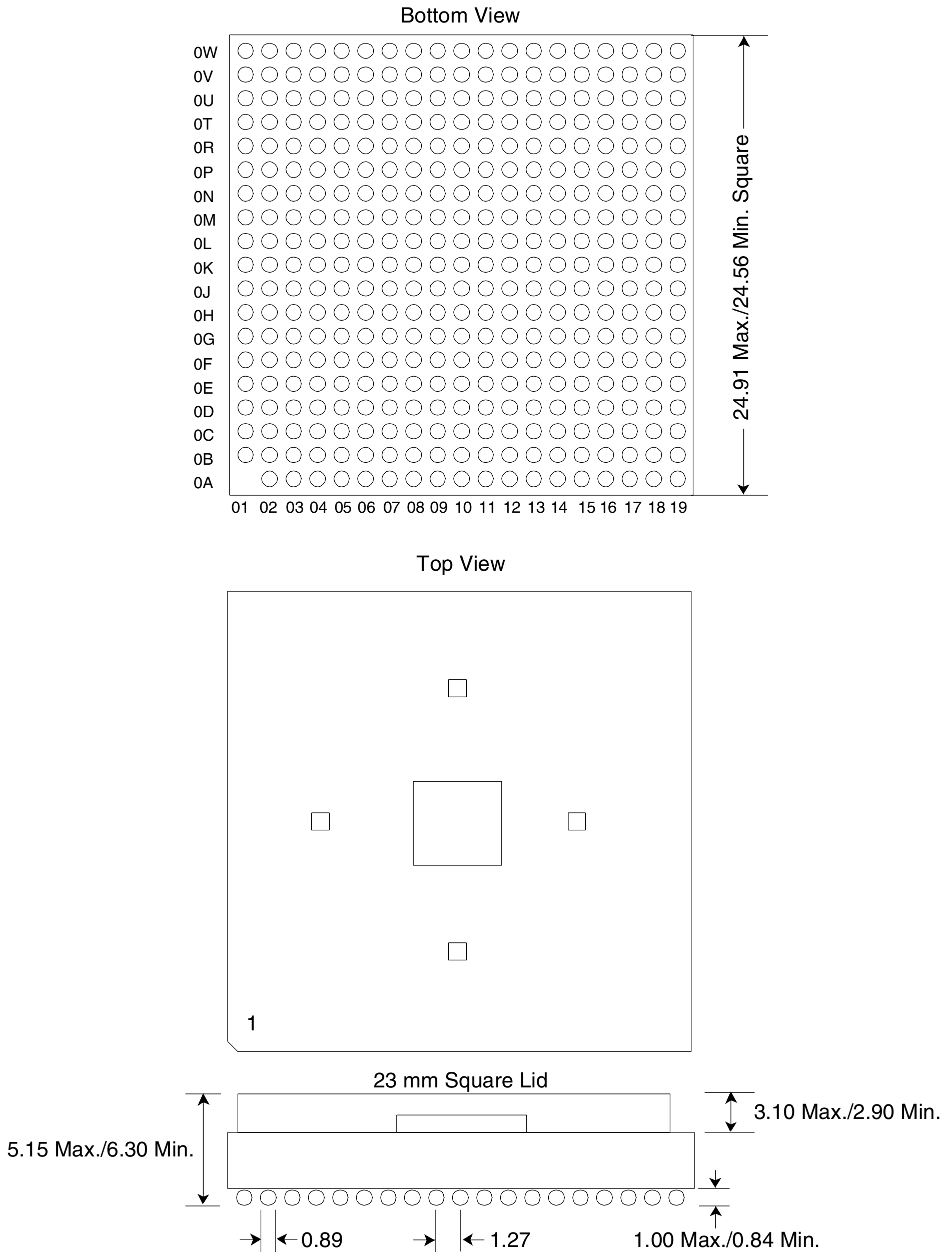
Table 44: 1.5 V West VDD 5 Signals

Sequence Number	I/O Location
1	B08
2	B16
3	D06
4	D14
5	F12

9. Packaging Information

CBGA 360 with standard aluminum lid and grease.

Figure 39: CBGA 25 mm x 25 mm 360-Lead Schematic



10. Appendix A: Data Aligned Serial Link (DASL)

10.1 General Description

The Data Aligned Serial Link interface is intended for high speed point to point inter-chip communication. The interface macro performs multi-bit serialization and de-serialization to reduce the I/O pin count. It is a synchronous device, removing the need for asynchronous interfaces that introduce additional interface latency. DASLs are intended to operate over a back-plane without any additional components. The DASL macro is designed for high levels of integration and uses reduced voltage differential transceivers to reduce power consumption.

Figure 40: Data Aligned Serial Interface Lines

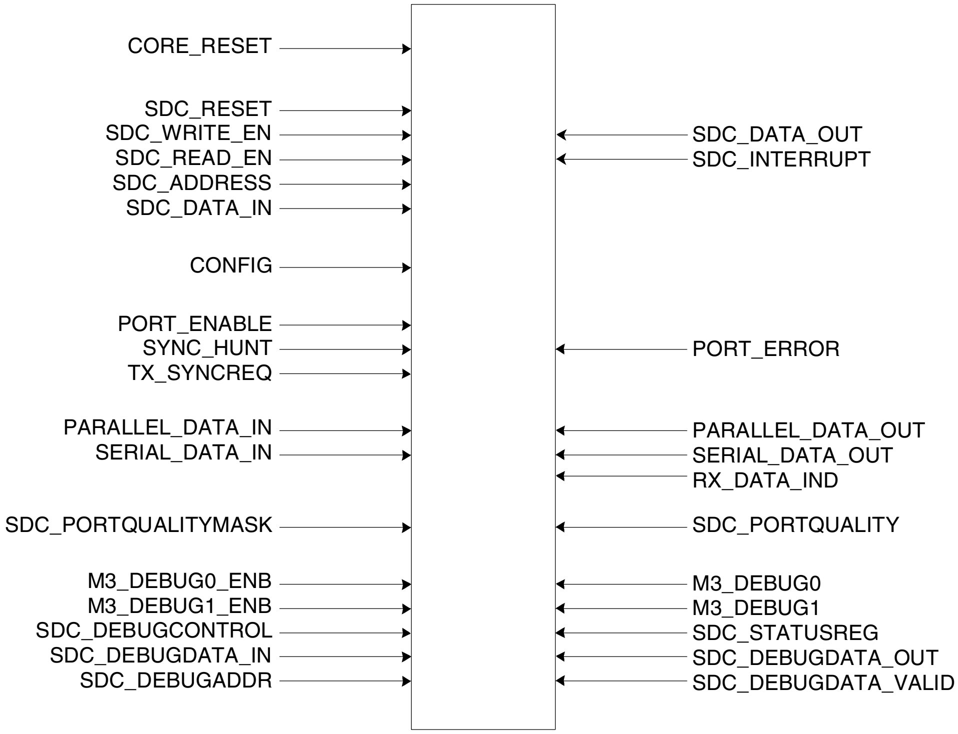


Table 45: Interface Description for DASL Internal Signals (Page 1 of 3)

Signals	Direction	Description
Clocking		
Reset		
CORE_RESET (DASL reset)	In	Core Logic Discrete Reset (active reset signal)
SDC_RESET (SDC Reset)	In	Shared DASL Controller Processor Reset Active reset signal used in conjunction with picocode download.
Test		
SDC Internal Resource Interface		
SDC_WRITE_EN	In	SDC Write Enable triggers a write operation during picocode download.
SDC_READ_EN	In	SDC Read Enable triggers a picocode read operation.
SDC_ADDRESS(0:10)	In	SDC Address Bus
SDC_DATA_IN(0:15)	In	SDC Write Data Bus
SDC_DATA_OUT(0:15)	Out	SDC Read Data Bus
SDC_INTERRUPT	Out	SDC Interrupt.
Application Control Interface		
CONFIG(0:15)	In	Configuration Register (0:4) Number Physical DASL Ports - 1 (n-1). (MSB) = Bit 4. (5:7) Number of Physical Sub-Ports -1 (m-1). MSB = Bit 7. Only two values are supported: (5:7)=110' for a 16-bit port (5:7)=111 for a 32-bit port (8:12) Unique Hardware Version Number (obtain from provider of the source picocode). MSB = Bit 12. (13) External Speed Expansion Mode. This bit should be tied high if external speed expansion is enabled. **This is a reserved function and should not be enabled. This bit must be held low.** (14) Internal Speed Expansion Mode. This bit should be tied high if internal speed expansion is enabled. **This is a reserved function and should not be enabled. This bit must be held low**. (15) DASL Synchronization Mode: same value as the signal DASLSYNC-MODE.
Application Port Interface		
PORT_ENABLE	In	DASL Port Enable
SYNC_HUNT	In	DASL Synchronization Request A low to high transition will force the synchronization sequence of the receive DASL port
PARALLEL_DATA_IN(31:0)	In	Parallel Data In: Data to be transmitted.
TX_SYNCREQ	In	Transmit Link Synchronization Request Input: When active the DASL TXPORT enters an idle state and will begin the synchronization sequence after the falling edge.
RX_DATA_IND	Out	Receive Data Indication: indicates, when active, the DATA_OUT bus now holds valid data
PARALLEL_DATA_OUT(31:0)	Out	Parallel Data Out: data received from the DASL
PORT_ERROR	Out	Port Error: indicates, when active, that an error has been detected by the Shared DASL Controller.

Table 45: Interface Description for DASL Internal Signals (Page 2 of 3)

Signals	Direction	Description
SDC_PORTQUALITYMASK	In	SDC Port Quality Mask: indicates, when active, that the SDC Port Quality Bit should be reset.
SDC_PORTQUALITY	Out	SDC Port Quality: indicates, when active, that the Shared DASL Controller has detected a minimum eye violation on a port. This violation indicates that one or more of the DASLs that make up the port is not operating within specification.
Application DASL Off Chip Interface		
SERIAL_DATA_IN(0:7)	In	Serial Data Input: Serialized data stream received at the serial link speed
SERIAL_DATA_OUT(0:7)	Out	Serial Data Out: Serialized data stream generated at the serial link speed
Diagnostic and Debug		
M3_DEBUG0_ENB	In	Processor Debug 0 Bus Enable: indicates, when active, debug information is available on the M3_DEBUG0 bus. When not active, the M3_DEBUG0 bus will be all zeros.
M3_DEBUG0(0:15)	Out	Debug 0 Bus (0:1) Reserved. All bits are set to zero. (2:4) ALU Status Bits. MSB = Bit 2 (5:15) PC (Program Counter). MSB = Bit 5
M3_DEBUG1_ENB	In	Processor Debug1 Bus Enable: indicates, when active, debug information is available on the M3_DEBUG1 bus. When not active, the M3_DEBUG1 bus will be all zeros.
M3_DEBUG1(0:15)	Out	Debug1 Bus (Instruction Decoder) (0) PX_AUTO_INCREMENT(14:15) From IDCD_CC unit. MSB = Bit 14 (1) ACCESS_PX (2) PY_AUTO_INCREMENT (3) ACCESS_PY (4:5) ALU_A_SELECT. MSB = Bit 4 (6:7) ALU_B_SELECT. MSB = Bit 6 (8:9) DATA_WIDTH. MSB = Bit 8 (10) IMMEDIATE_DATA_FROM_ INSTRUCTION (11) MUX1_CNTL (12) MUXQ_CNTL (13) MUXR_CNTL (14:15) From IDCD_CC unit. MSB = Bit 14
SDC_DEBUGCONTROL(0:31)	In	SDC Debug Control Input (0) Debug Register Enable (1) Port Processing Disable (2) Temperature Compensation Disable (3) DASL Adjustment Disable (4:7) Encoded Debug Select - (8:11) Port Select - See “Debug Control Input Definition” (12:15) DASL Select - See “Debug Control Input Definition” (16:23) Reserved. All bits are set to zero. (24:31) Status Count Input - Used in conjunction with the Status Count Output field of the SDC Status Register to create a watchdog mechanism. The DASL Controller will increment this field by one and place the result in the Status Count Output field of the SDC Status Register.
SDC_DEBUGDATA_IN(0:31)	In	Contains the SDC Debug write data. The data is validated by an active Debug Register Enable with a write command.
SDC_DEBUGADDR(0:15)	In	Contains the SDC Debug address for read and write commands. The address is validated by an active Debug Register Enable with a read/write command. See “SDC Debug Interface Address Map”.

Table 45: Interface Description for DASL Internal Signals (Page 3 of 3)

Signals	Direction	Description
SDC_STATUSREG(0:31)	Out	SDC Status Register (0:7) DASL Control Code Version (8:15) DASL Control Code Revision (16:17) Reserved. All bits are set to zero. (18) Parity Error - A parity error was detected on a Local Store or Instruction RAM read. Active high. (19) Port Quality Error - A logical "or" of the SDC_PORTQUALITY output. Indicates, when active, that the Shared DASL Controller has detected a minimum eye violation on a port. This violation indicates that one or more of the DASLs that make up the port is not operating within specification. (20:23) Current Port Identification - Displays the number of the port currently being processed by the Shared DASL Controller. (24:31) Status Count Output - Written by the Shared DASL Controller to be one greater than the value present in the Status Count Input field of the SDC Debug Control Input bus. Serves as a watchdog mechanism to allow the application to monitor the Shared DASL Controller.
SDC_DEBUGDATA_OUT(0:31)	Out	Contains the SDC Debug read data which is validated by an active SDC Debug Operation Complete. On write commands, this bus will contain the SDC Debug write data.
SDC_DEBUGDATA_VALID	Out	SDC Debug Operation Complete is active high for one OSC_125_DIV4 clock cycle to validate the data on the SDC_DEBUGDATA_OUT bus.

10.2 Resets

- SDC_RESET: (M3 Reset)
  The DASL processor reset must be active during picocode download. It is recommended to activate SDC_RESET during DASL reset (CORE_RESET).
- CORE_RESET: (DASL Reset)
  CORE_RESET signal resets the DASL. To insure proper reset, it is recommended to maintain the CORE_RESET signal during at least 500 ns.

Both SDC_RESET and CORE_RESET signals are synchronous resets, and therefore need the presence of a clock to be performed.

10.3 Picocode Download

The Instruction Memory for the DASL Shared Controller is accessible through the SDC Internal Resource Interface (SDCIRI). The processor is given a separate reset so that it can be disabled until the Instruction Memory is loaded. The SDC_RESET signal must be held active until Instruction Memory is completely loaded. Once the picocode is completely loaded, release the SDC_RESET to start the processor.

10.3.1 Picocode Write

A picocode write operation is performed using the M3 picocode access register @40 for X plane and @60 for Y plane. Data Instruction, Instruction Memory Address, and Write Enable bits must be written in the SDC access register. The hardware handles the transfer in the Instruction Memory. Contiguous operations can be performed until the picocode is completely downloaded.

10.3.2 Picocode Read

A picocode read operation is performed using the M3 picocode access register @40 for X plane and @60 for Y plane. Instruction Memory Address and Read Enable bits must be written in the SDC access register. The hardware handles the read operation.

The SDC access register shall be polled until the Read Completed bit is detected to be ON (active high). When this bit is ON, the data contained in the SDC access register is valid and correspond to the selected Instruction Memory Address.

10.4 SDC_INTERRUPT Signal

A high level on SDC_INTERRUPT (M3 interruption in registers 10 and 30) denotes an interrupt or a parity error condition. If a parity error is detected on the instruction memory or the local data store, an interrupt will be generated and remain active until the Shared DASL Controller is reset.

10.5 SDC Debug Interface

The SDC debug interface allows the application read and write access to any of the resources available to the processor. Access includes local store, sample memory, and any hardware-assist registers addressable by the processor (access to instruction memory is provided through SDC Internal Resource Interface). The interface includes a 32-bit debug control input, a 32-bit debug data output, a 32-bit debug data input, a 16-bit debug address input, and a one bit debug data valid output. The application requests a service offered by the processor via the debug control input. The results are returned by the processor on the debug data output. The debug control input definition is shown below.

Bit #	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	28	29	30	31
Function	Debug Register Enable	Halt Processor	Halt Port Processing	Halt Temperature Compensation	Restrict Temperature Compensation	Halt on Initialization Error	Halt on Temperature Compensation Error	Force DASL Refresh	Reserved				Encoded Debug Select				Reserved	DASL Select			Port Select				Port Status Count							

The definition of each field is shown in the following table.

Table 46: Debug Control Field Definitions

Field Name	Function
Debug Register Enable	High on this signal enables debug mode. When this bit is low, the rest of the bits in the Debug Control Input will be ignored with exception of the Port Status Control field.
Halt Processing	
Halt Port Processing	
Halt Temperature Compensation	
Restrict Temperature Compensation	
Halt on Initialization Error	
Halt on Temperature Compensation Error	
Force DASL Refresh	
Encoded Debug Select	The value of this field indicates which service the picocode should perform. Valid values: ‘0000’ No operation ‘0001’ Request a read to the address given by the Debug Address ‘0010’ Request a write to the address given by the Debug Address ‘0100’ Dump Local Store Memory ‘0101’ Load Sample Memory ‘0110’ Dump Sample Memory ‘1000’ Request a delay line sample from the DASL receiver specified by the Port Select and DASL Select fields. ‘1001’ Update the DASL Data Structure with the Debug Data Input for the DASL receiver specified by the Port Select and DASL Select fields. OTHERS Reserved

Table 46: Debug Control Field Definitions

Field Name	Function
Port Select	An encoded value that selects the port for a given action. A request that utilizes the Port Select field is 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.
DASL Select	An encoded value that selects the DASL receiver for a given action. If the supplied DASL value is out of range, the request will be processed for DASL 0.
Reserved	Reserved. Should be low.
Port Status Count	This 8-bit field will be read by the picoprocessor, incremented, and written back out of the sdc_statusreg_out output.

The status register contain various information about the processor status. The output of this register is connected to the SDC_STATUSREG_OUT output.

Table 47: Status Register Definition

Bit #	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	28	29	30	31
	LSB							MSB	LSB							MSB					LSB			MSB	LSB							MSB
Function	Code Version								Code Revision								Reserved				Current Port ID				Status Count Output							

Table 48: Data Aligned Serial Link (DASL) Signals for Synchronization

Signal Name	Description
DRIVER_ENABLE	Turns ON the DASL Drivers
PORT_ENABLE	Starts the picocode for synchronization operations. When active high, the picocode starts looking at TX_SYNCREQ and SYNC_HUNT signals for DASL transmit and DASL receive synchronization respectively.
TX_SYNCREQ	Triggers the DASL serial data out. When active high, the data delivered to the DASL driver is forced to a steady state high level. On falling edge, the data serialization starts.
SYNC_HUNT	Starts DASL receiver synchronization based on incoming synchronization pattern.

10.6 Data Aligned Serial Link (DASL) Initialization and Operation

Once the chip has been fully configured, but before actual data traffic can take place between an IBM Packet Routing Switch Serial Interface Converter (the converter) and an IBM 28.4G Packet Routing Switch (switch), the DASL interfaces must be initialized to provide bit phase alignment and packet alignment at the data receivers in both directions.

DASL initialization means communication between two chips: a converter and a switch core port.

The port synchronization is under the overall control of the system Control Processor which coordinates the operation between the switch core and the adapters, but synchronization between the switch control and the port adapter can also be performed directly through the interface lines, Fabric Port Available (FPA), and Adapter Port Available (APA).

The registers of interest at both ends of the link are:

- Port enable register

- No signal register
- Sync Status and Sync Hunt register
- Sync Packet Transmit register
- Signal detect Interrupt in Status register

The Sync Status register reports the status of the input receiver, and the Sync Hunt register forces the input ports to start the synchronization sequence.

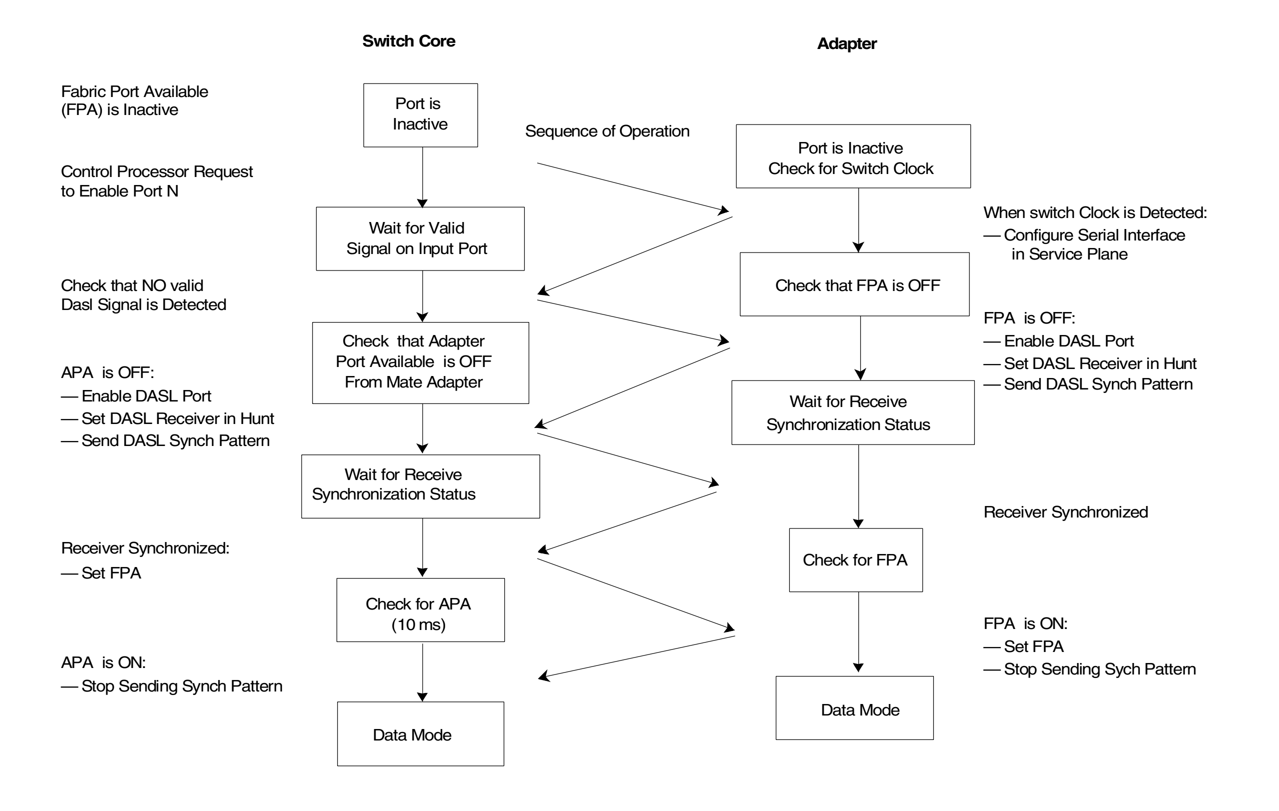
The Sync Packet Transmit register specifies that synchronization packets have to be transmitted in order for the remote chip input port to synchronize. When not transmitting synchronization packets, the output ports transmit normal traffic packets: data packets or idle packets.

The following steps must be taken in order to synchronize input ports, either after reset and initialization of the chip or when the control processor decides to re-synchronize a link due to data errors on the incoming links. Even if the same steps have to be taken on both the switch port and the IBM Packet Routing Switch Serial Interface Converter (the converter) chip, they don't have to be synchronous, but the global sequence of operation must be followed:

1. Disable the switch and converter ports.
2. Enable the switch and converter ports by writing binary '1' Port Enable register.
3. Disable DASL transmission by disabling 'transmit synch enable'.
4. Check for valid connectivity of the receiver to a differential transmitter through the No Signal Register. This ensures the integrity of the serial links.
5. Enable 'Transmit Sync Packets'.
6. Enable DASL transmission by writing a 1 in 'transmit synch enable'.
7. Write a 1 into the Sync Hunt register for the enabled ports to start synchronization.
8. Poll the Sync Status register to verify completion of synchronization after a Sync Time-Out period. If the port fails to synchronize, its Sync Status bits will be `0'.
9. When synchronization has been achieved the local processor at each end reports to the control processor that it is ready for data transfer.
10. Clear any CRC error indication that might have been set up during the synchronization period.
11. Upon reception of both reports the control processor indicates to both ends of the full duplex link (switch port and converter) to stop transmitting Sync packets. Normal packet transfer (idle or data) on the input ports can then be initiated.
12. As the link is now in data mode both the switch control and the adapter controller have to poll the CRC error registers and the No Signal Register to check that the receiver is synchronized and for error free operation.

This operational sequence is mapped in the following flow chart:

Figure 41: Switch Fabric DASL Port Synchronization Sequence



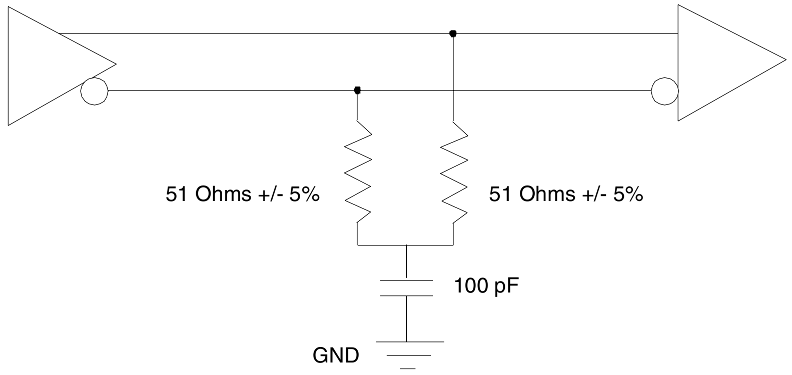
10.7 Line Termination

10.8 DASL and SYS_CLK

The DASL data inputs (DASL_X_TX, DASL_Y_TX) and the clock reference (SWITCH_X_CLK and SWITCH_Y_CLK) use differential HSTL EIA/JEDEC (EIA/JESD8-6) standard compliant I/O books. The following figure gives the recommended termination for those receiver lines.

10.9

Figure 42: DASL Termination



The termination network must be placed within 2.5 cm of the receiver chip.

Note: For switch clock X/Y differential pair, both termination resistors must be directly connected to GND, without the use of the 100 pF capacitor.

11. Appendix B: PLL

11.1 PLL Configuration

During Chip POR, the PLL is held reset and stays in reset mode until a software access releases the PLL RESET bit in the register configuration table. Before starting the PLL (by releasing the PLL RESET bit), the PLL RANGE, PLL MULT and PLL TUNE bits must be written in the PLL register. Once the PLL is started, a poll the PLL register for PLLLOCK bit detection. When the PLLLOCK bit is detected high, the logic clocked by the PLL is ready to work.

11.2 PLL Reset

PLL_RESET bit [15] is provided in the register configuration table @ 90, 94, 9C. PLL RANGE [bits 5-0] and MULT [bits 31-28] bits are provided in the register configuration table @ 90, 94, 9C. PLL TUNE bits [13-8] are provided in the register configuration table @ 90, 94, 9C. PLLLOCK bit [14] is provided in the register configuration table @ 90, 94, 9C.

The RESET function puts the PLL in bypass mode so the output clock is identical to the input reference clock. RESET should be held active (high) during power-on until all of the following condition are met: All PLL inputs are stable and at their final values. The reference clock is stable. PLL VDDA and Vdd are at their final values. A RESET is also required should the PLL inputs change after power-on.

11.3 PLL Range and MULT

PLL RANGE [bits 5-0] and MULT [bits 31-28] are provided in the register configuration table @ 90, 94, 9C. Use these programmable inputs to choose the output frequency of the PLL. The logic state of these inputs must be stable before the PLL can begin to lock.

11.4 PLL Tune

PLL TUNE bits [13-8] are provided in the register configuration table @ 90, 94, 9C. These programmable inputs are used to optimize the PLL stability and jitter. The logic state of these inputs must be stable before the PLL can begin to lock.

11.5 PLL LOCK

PLLLOCK bit [14] is provided in the register configuration table @ 90, 94, 9C. This signal indicates when the feedback clock is in phase with the reference clock. While RESET is high, PLLLOCK will be low. Following reset, PLLLOCK will stay low until lock is achieved.

The PLLLOCK signal will go high less than 300 µs after:

- The reference clock is stabilized at a constant frequency
- The RANGE and MULT inputs are at their final states
- The VDDA input is at the rail

PLLLOCK will return to a low state if the PLL loses lock with the reference clock.

11.6 PLL Settings

The following table gives the recommended values for the IBM 28.4G Packet Routing Switch PLL settings:

Settings (PLL registers)	IBM 28.4G Packet Routing Switch	Switch Operating at 125 MHz Byte Clock
Reference clock	55 MHz	62.5 MHz
PLL RANGE A (bits [2-0])	010	010
PLL RANGE B (bits [5-3])	101	101
PLL MULT (bits [31-28])	0010	0010
PLL TUNE (bits [13-8])	010011	010011
VCO frequency	660 MHz	750 MHz

Note: If the PE PLL is used with a 55 MHz reference clock, the same settings are used except for the unused PLL RANGE B bits.

11.6.0.1 Example of PE PLL Programming

Frequency In	PLL Out Frequency	Range	Mult	VCO	Tune
50	100	001	0010	700[1]	010011
		010		600	010011
		011		500	010011
100	100	001	0001	700[1]	010011
		010		600	010010
		011		500	010010
55	110	001	0010	770[1]	010011
		010		660	010011
		011		550	010011
		100		440	010011
110	110	001	0001	770[1]	010011
		010		660	010010
		011		550	010010
		100		440	010001
62.5	125	010	0010	750[1]	010011
		011		625	010011
		100		500	010011
125	125	010	0001	750[1]	010010
		011		625	010010
		100		500	010001

[1] This is the recommended setting as it is the highest VCO frequency which corresponds to the minimum clock jitter

12. Glossary

12.1 Definitions

Cell	Small portion of fixed length data transported as an entity and not partitioned further. IBM Packet Routing Switch Serial Interface Converter Cell is 64 - 80 bytes long in steps of four bytes
Packet	Used interchangeably with cell
Frame	Stream of packaged data which is multiple of packet length divided by four (so equal to LU length) for instance in the back pressure link if there are only two priorities in use then the frame length of the framing signal will be two equivalent to two logical Unit of the IBM 28.4G Packet Routing Switch or two packet of data where the length of data packet is expressed in words (4 bytes/32 bits)
Ingress	Towards the switch core, corresponds to the data RECEIVED by the switch.
Egress	Away from the switch core, corresponds to the data TRANSMITTED by the switch.

12.2 Terms, Abbreviations, and Acronyms

BM	Bit Map addressing: if there is a '1' in this field, the data which is within the same packet is for the corresponding output port of the IBM 28.4G Packet Routing Switch
Control	Control packet communicates with the switch control. The bit map of such a packet is set to all zero. Upon detecting such value the switch automatically route the packet to the switch control. In response the switch control can also insert control packet in the output queues of the switch.
CRC	Cyclic Redundancy Check. A code used to validate a block of data. A CRC8 is used in the IBM Packet Routing Switch Serial Interface Converter
IBFC	In Band Flow Control corresponds to the case where the switch flow control information (output queue grant and memory grant in the egress direction and send grant in the ingress direction) are carried within any contiguous packets (idle or data) flowing between the protocol engine and the converter.
Idle	Idle packets are used as time fill packet to maintain the integrity of the bus when there is no data
LU	Logical Unit. Defines the length of an 8-bits row of data processed by a single shared memory within the switch. The IBM Packet Routing Switch Serial Interface Converter handles LUs of 16 to 20 bytes corresponding to packet sizes of 64 to 80 bytes.
OBFC	Out of Band Flow Control corresponds to the case where the switch flow control information (output queue grant and memory grant in the egress direction and send grant in the ingress direction) is carried through $\overline{\text{RXENB}}$ or $\overline{\text{TXFULL}}$.
PE	Protocol Engine (protocol processor, ATM layer...) is the device that attaches on the line side to the IBM Packet Routing Switch Serial Interface Converter which runs the higher layer protocols.
PQ	Packet Qualifier is a byte of information used by the switch to interpret packet priority, packet idle or data, or packet color.
Res	Reserved for future use
SDC/M3	Shared DASL Controller (also called M3 picoprocessor)controls the DASL transceiver for the delay line management for temperature tracking, clock drift....
Service	Service packet is any fill up packet (idle resulting from a flow control situation or an empty link) or any physical layer interface packet to check the continuity of a link packet (yellow packet)
Synch	Synchronization packets are service packets used to train the receiver of the DASL in order to acquire bit/Byte/Packet synchronization.
IBM Packet Routing Switch Serial Interface Converter	Chip performing UTOPIA-3 like parallel interface to switch DASL interface conversion
Yellow	Service packet used to check the continuity of the physical link between the IBM 28.4G Packet Routing Switch and the IBM Packet Routing Switch Serial Interface Converter

Revision Log

Revision	Description of Modification
10/15/99	Initial release (00).
07/12/00	First revision (01). Added Grant Control Generation to Figure 2. Added numbering to all headings and tables. Added Table of Contents, List of Figures, List of Tables.