PM73488

QSE

5 Gbit/s ATM Switch Fabric Element

DATASHEET

Released
Issue 3: June 1999

AAL1gator, AAL1gator2, Evil Twin Switching, QRT, QSE, and SATURN are trademarks of PMC-Sierra, Inc.

AMCC is a registered trademark of Applied MicroCircuits Corporation.

i960 is a registered trademark of Intel Corporation.

National Semiconductor is a registered trademark of National Semiconductor Corporation.

Vitesse is a trademark of Vitesse Semiconductor Corporation.

All other brand or product names are trademarks
of their respective companies or organizations.

U.S. Patents 5,557,607, 5,570,348, and 5,583,861

Public Revision History

Issue Number	Issue Date	Details of Change
Issue 1	March 1998	Creation of Document
Issue 2	October 1998	Fixed all known typos/errors (*e.g.* wrong pin-out: RAM_ADD(16) and RAM_PARITY swapped).
Issue 3	June 1999	Production Release Version

PMC-Sierra, Inc.
105-8555 Baxter Place
Burnaby BC Canada V5A 4V7
Phone: 604.415.6000 FAX: 604.415.6200

Table of Contents

List of Figures

List of Tables

Product Overview

DESCRIPTION

The PM73488 (QSE) is an advanced communications device that enables the implementation of high performance switching systems. The QSE is a 32 × 32 cell based switch element, with a total *sustainable bandwidth* of 5 Gb/s. (The *peak, or raw, bandwidth* is much more than that: about 8 Gb/s). The QSE is designed to be used with other QSE's as part of a larger switch fabric. Various QSE combinations allow fabrics with theoretical peak capacities ranging from 5 Gb/s (one QSE) to 160 Gb/s. The QSE is not ATM specific; however, should the QSE be used for switching ATM cells, the QSE cell size is large enough to allow efficient direct mapping between QSE Cells and ATM cells.

Multistage QSE Fabrics (Delta-Reverse Delta configuration) have rich connectivity with multiple paths between each source/destination pair. A QSE fabric performs cut-through unicast switching and uses Randomization and Evil-Twin algorithms to fully utilize these multiple paths and avoid the build up of internal hot spots. Randomization, in combination with multiple routing paths allows graceful degradation of QSE Fabric performance if internal links fail. To detect failed links, the QSE maintains and checks liveness patterns on input and output ports in hardware, and automatically routes around ports if they die.

QSE data ports are 6 bits wide including a 4-bit wide 66 MHz data path, a one-bit wide start-of-cell indication, and a one-bit wide acknowledgment signal. Each port contains "Phase Aligners" to recover the clock for that port, thus removing the need to synchronize all data to a single global clock.

When switching unicast traffic in a multistage fabric (one to three stages), the first nibble of the cell will come out of the last QSE stage before the last nibble of that cell enters the first stage. The cell thereby traverses the entire fabric in one cell time. If the cell sucessfully makes it to its destination, the ("egress") device accepting the cell from the last stage QSE has the opportunity to send a four bit "Ack Information Packet" back to source indicating what it did with this cell; at its simplest, the egress device can send back one pattern to indicate that the cell was accepted and another to indicate that the cell was dropped due to, say, buffer overflow.

It is also possible that the cell was dropped inside the QSE fabric due to say a collision with another cell. The QSE classifies lost cells as due to one of three causes (collision, all possible outputs dead, or parity errors) and will generate an "Ack Information Packet" back to the source to communicate this event. In each QSE, the 4 bit pattern in the information packet can be independently software configured for each of the three cases. Note that since each QSE can be separately programmed, the patterns can even be setup so that the source knows where the cell was dropped.

The information provided by the "Ack Information Packets" can be used by the device injecting cells into the first QSE stage to decide how to handle the cells; at its simplest, the device can resend cells that did not get through (a more sophisticated algorithm might also take into account where the cell was lost and the behavior of the evil twin algorithm to decide when to resend the cell; for example if the cell was dropped due to output congestion it might make sense to back off on cells to that output).

For unicast traffic, part of switch bandwidth will be used to resend cells that did not make it through the first time around. This implies that sustained throughput is less than peak switching capacity. The amount of bandwidth required for resending cells and the effect of resending on latency and "Cell Delay Variation (CDV)" has been extensively studied with analytical models of the fabric. These results have then been cross checked with results from simulating software models of the fabric. This data is crucial for designing fabrics that can efficiently support guaranteed "Quality of Service (QOS)" requirements. The recommended QSE fabric configurations for high quality switching takes these results into account; for example the 3 stage 160 Gb/s sustained throughput fabric has a peak capacity of 256 Gb/s (60% margin).

The QSE fabric is store-and-forward for multicast traffic. Cell replication is performed in an optimal tree based manner where replication is done as far downstream as possible and each QSE contains cell buffers to buffer multicast cells. A multipriority backpressure feedback is used to control the flow of multicast cells through the fabric.

FEATURES

Switching Algorithm

- Supports blocking resolution in the switch fabric.
- Guarantees a lower bound on switch performance using a patented randomization algorithm called Evil Twin Switching.
- Determines routes using specified bits in the header (self-routing switch fabric) for unicast traffic.
- Determines output groupings using a lookup table for multicast traffic.
- Allows output ports to be combined in groups of 1, 2, 4, 8, 16, or 32 for unicast traffic.
- Allows output ports to be combined in groups of 1, 2, or 4 for multicast traffic.

Multicast Support

- Supports optimal tree-based multicast replication in the switch fabric.
- Supports 512 internal multicast groups, expandable to 256K with external SRAM.
- Provides 64 internal cell buffers for multicast cells.

Diagnostic/Robustness Features

- Checks the header parity.
- Counts tagged cells.
- Checks for connectivity and stuck-at faults on all switch fabric interconnects.

I/O Features

- Provides 32 switch fabric interfaces with integrated phase aligner clock recovery circuitry.
- Provides a Start-Of-Cell (SOC) output per four switch element interfaces.
- Provides an external 8-bit Synchronous SRAM (SSRAM) interface for multicast group expansion.
- Provides a demultiplexed address/data CPU interface.
- Provides an IEEE 1149.1 (JTAG) boundary scan test bus.

Physical Characteristics

- 3.3 V supply voltage.
- 5 V tolerant inputs.
- 596-pin Enhanced Plastic Ball Grid Array (EPBGA) package.
- Operates from a single 66 MHz clock.

Figure 1 shows a QSE system block diagram.

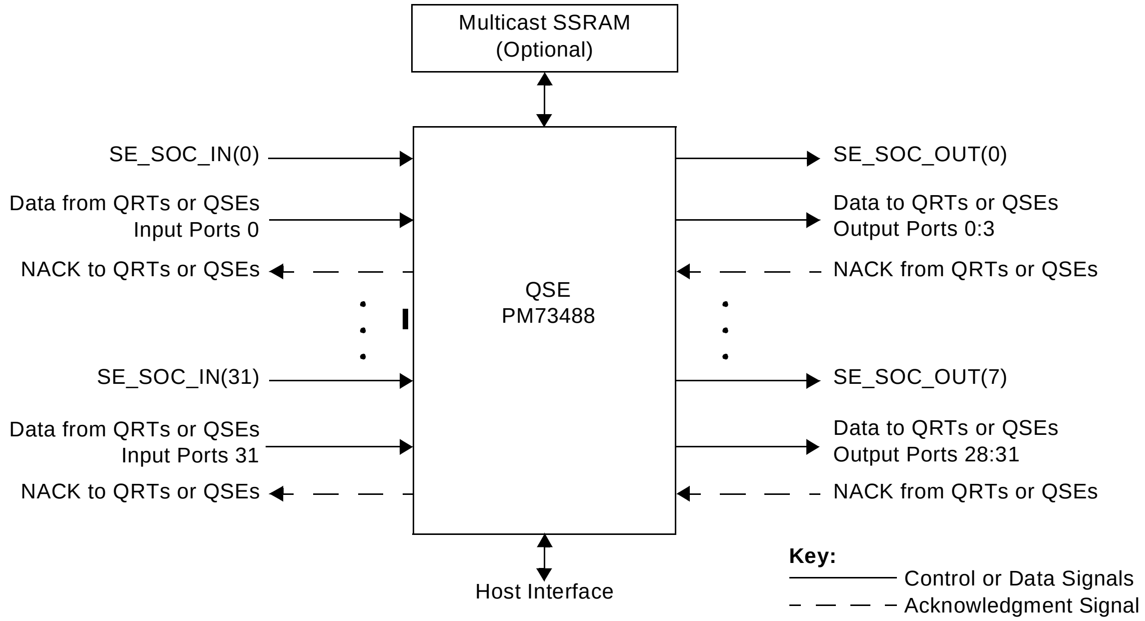


Figure 1. QSE Interface Block Diagram

1 HOW THE QSE FITS INTO YOUR SYSTEM

The QSE, together with the QRT, supports a wide range of high-performance ATM switching systems. These systems range in size from 5 Gbps to 160 Gbps. The systems can be developed to provide scalability with linear cost. Another key feature of the QSE/QRT architecture is that it is exceptionally fault-tolerant, both in the switch fabric and the UTOPIA interface.

This section contains a quick overview of the QSE and several example applications:

- a 5 Gbps switch using PM73487s and a PM73488,
- a 10 Gbps switch using PM73487s and PM73488s,
- a switch architecture using PM73487s and PM73488s that scales from 5 Gbps to 20 Gbps,
- a switch architecture using PM73487s and PM73488s that scales from 5 Gpbs to 160 Gbps

1.1 QSE System Overview

The QSE is switch element, combinations of which allows switch fabric implementations that span from 5 Gbps to 160 Gbps. The bandwidth of a single QSE is 5Gbps of *sustainable* bandwidth; the raw, or peak, bandwidth is 8Gbps. (Thus the QSE has an in-built speed-up factor of 8/5 = 1.6.)
The QSE has 32 input ports and 32 output ports. Each port is a 66 MHz 6-bit interface, out of which 4 are data and 2 are control. Each port can be connected to another QSE or QRT. Figure 2 shows a QSE connected to a QRT.

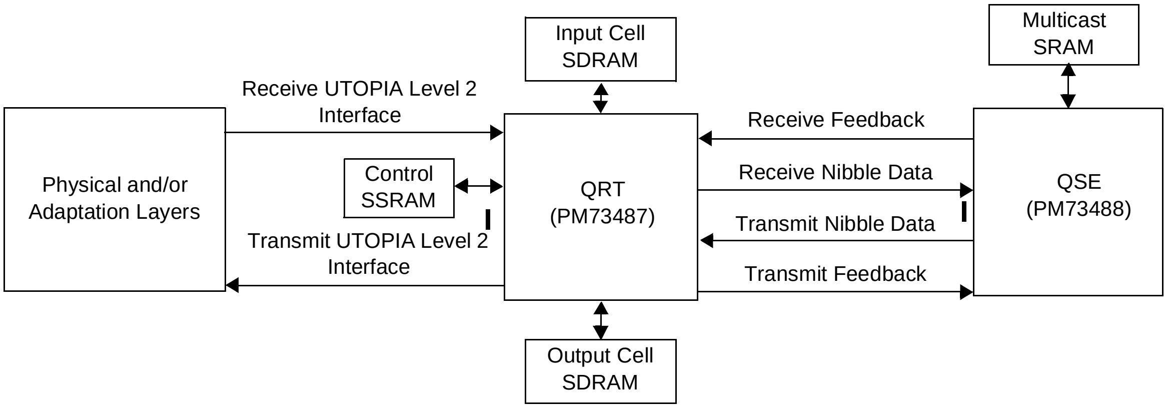


Figure 2. QSE System Overview

1.2 32 x 32 Switch Application (5 Gbps)

Figure 3 shows a basic 32 × 32 switch application (5 Gbps) using one QSE and eight QRTs.

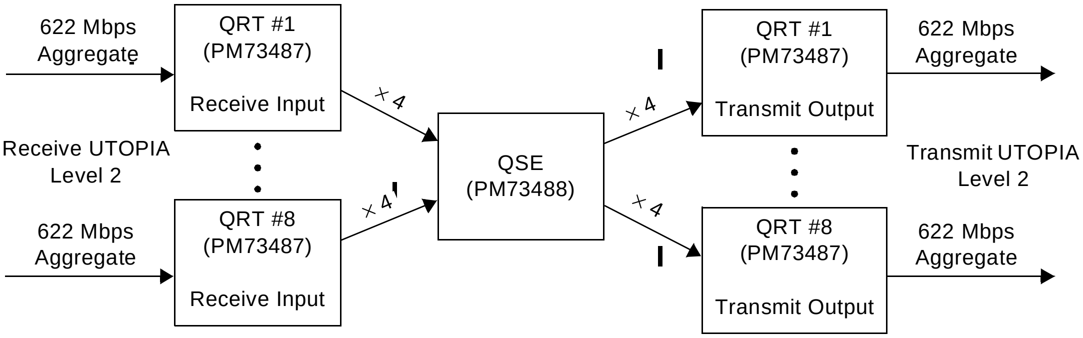


Figure 3. 32 x 32 Switch Application (5 Gbps)

1.3 64 x 64 Switch Application (10 Gbps)

Figure 4 shows a 64 × 64 switch application (10 Gbps) using 6 QSEs and 16 QRTs. This application uses QSEs in a 3-stage fabric. This sized system can be implemented in a single 19 inch rack.

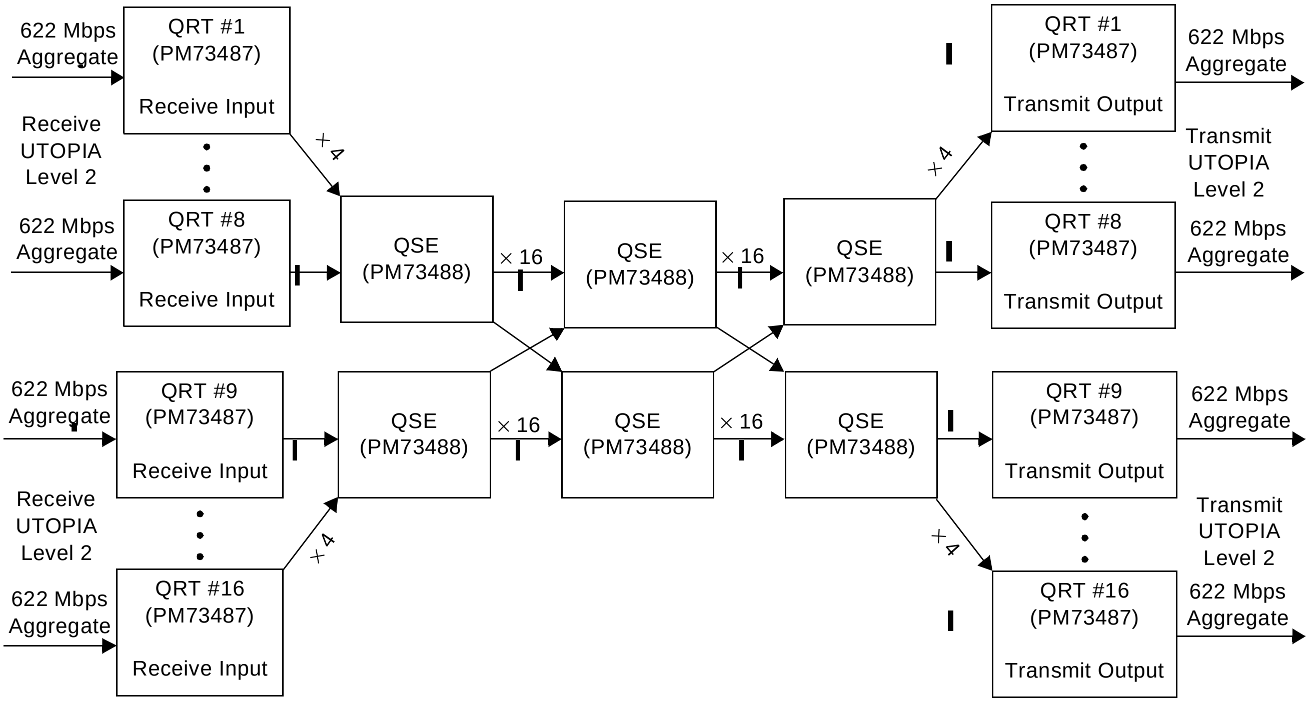


Figure 4. 64 x 64 Switch Application (10 Gbps)

1.4 5 Gbps to 20 Gbps Application Example - Seamless Growth

Figure 5 illustrates the modularity of the QSE and QRT architecture. A 5 Gbps system can immediately be created (as shown in Figure 5), then be upgraded to 10 Gbps (as shown in Figure 6), or 20 Gbps (as shown in Figure 7 on page 19) with the QSE and the QRT. Since systems composed of the QSEs and QRTs are based on a single-stage switch fabric, the per-port cost for each system will remain the same.

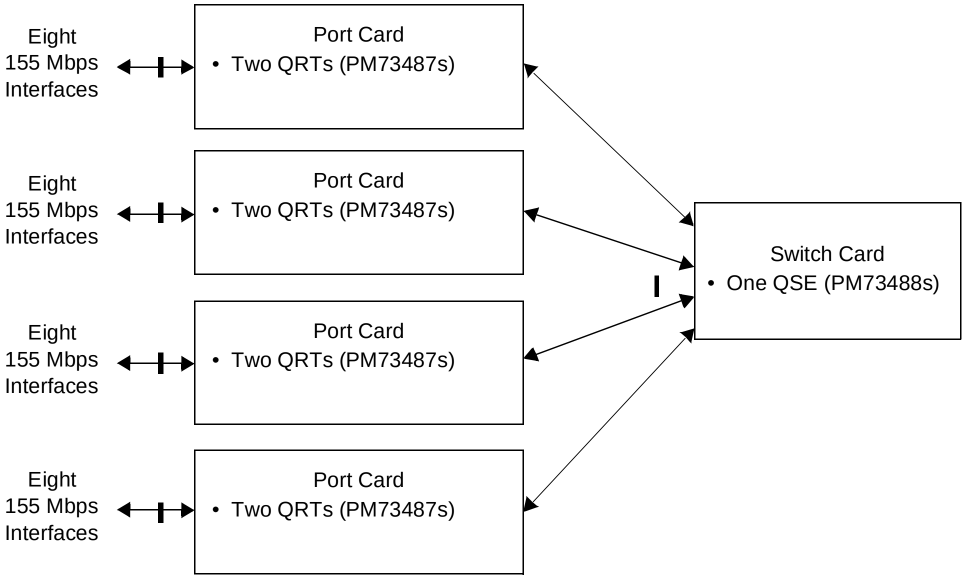


Figure 5. 5 Gbps ATM Switch Using 8 QRTs, and 1 QSE

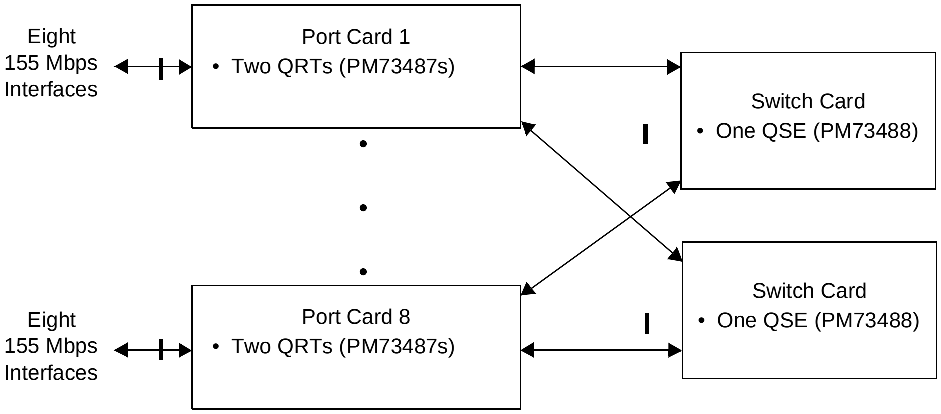


Figure 6. 10 Gbps ATM Switch Using 16 QRTs, and 2 QSEs

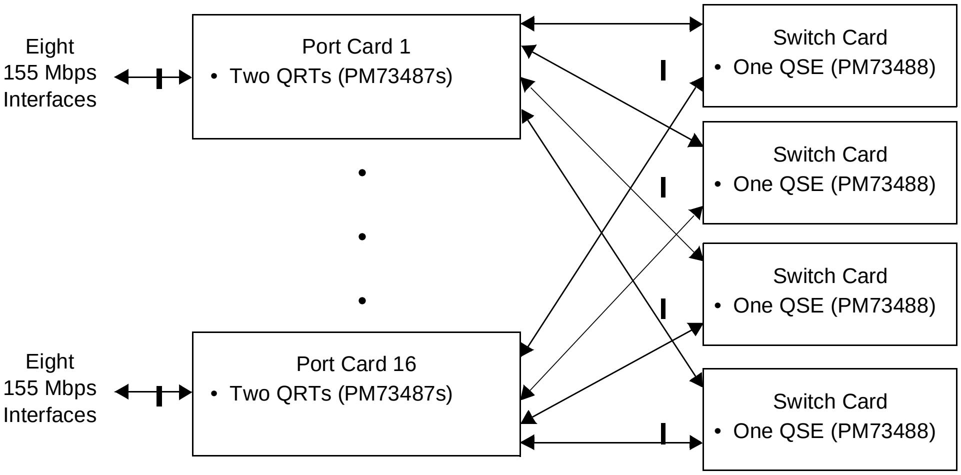


Figure 7. 20 Gbps ATM Switch Using 32 QRTs, and 4 QSEs

1.5 5 Gbps to 160 Gbps Application Example – LAN-to-WAN

A powerful application of the QRT and QSE devices is the creation of modules that can be used in a range of switches with only the interconnection changing between different sizes. ATM switches from 5 Gbps to 160 Gbps can be realized with only two unique cards. A port card has one QRT, and a switch card has two QSEs. The switch fabric consists of three stages, each with 32 QSEs (or 16 switch cards). To plan for future scalability, the middle stage must be built-in upfront. This is a one-time cost. Then, in order to scale in 5 Gbps increments, one switch card and its accompanying eight port cards should be added. Finer bandwidth scaling is possible by populating the additional switch card with port cards as needed (in increments of 622 Mbps). With this switch fabric topology, scaling is possible up to 160 Gbps. Once the initial middle stage cost has been incurred, the per-port cost for 5 Gbps through 160 Gbps systems remains constant

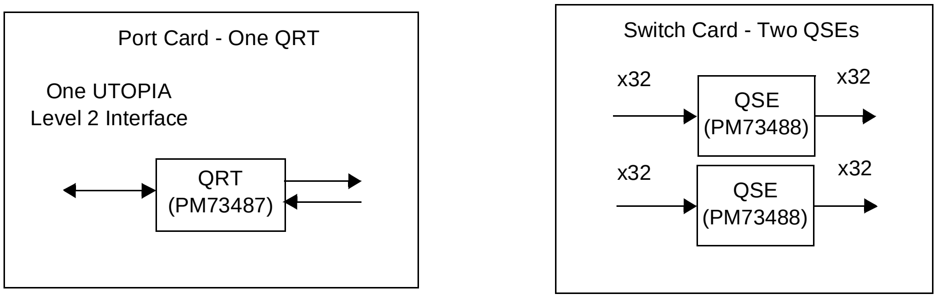


Figure 8. 5 Gbps to 160 Gbps Switches Modeled Using Only Two Cards

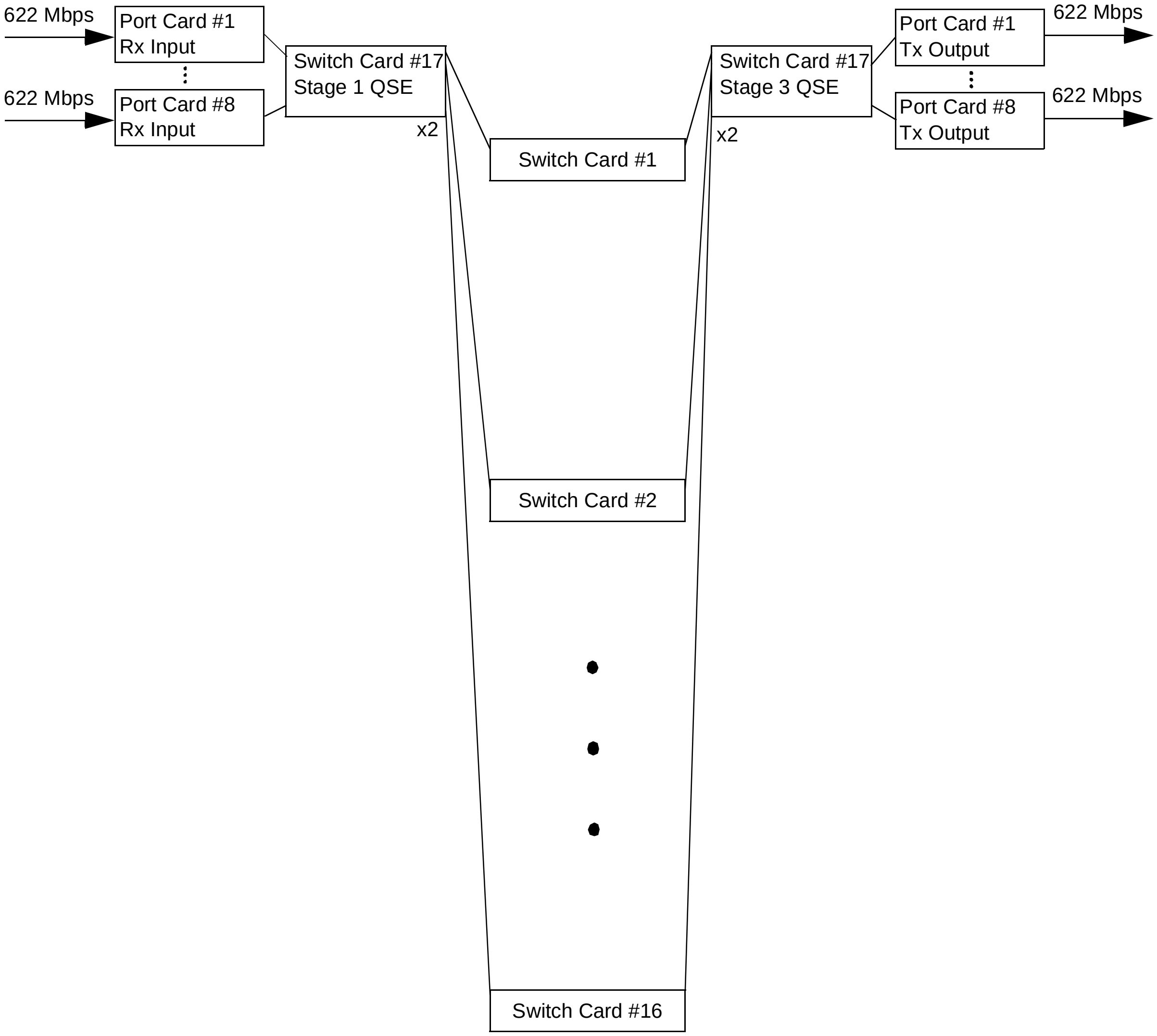


Figure 9. 5 Gbps ATM Switch

Figure 9 shows a 5 Gbps ATM switch using 8 port cards (8 QRTs) and 17 switch cards (34 QSEs). The middle stage is composed of 16 switch cards. The 5 Gbps bandwith is achieved by adding switch card #17 (which is depicted using two boxes: one stage 1 QSE and one stage 3 QSE), and eight port cards (each of which is depicted using two boxes: one for the Rx input side, and one for the Tx output side). Lines between stage 1 and stage 2, and stage 2 and stage 3 switch cards represent two sets of wires, one to each of the QSEs in the middle stage switch cards.

Figure 10 shows a 10 Gbps ATM switch using 16 port cards (16 QRTs) and 18 switch cards (36 QSEs). Here, another switch card and eight port cards have been added to the 5 Gbps switch depicted in Figure 9.

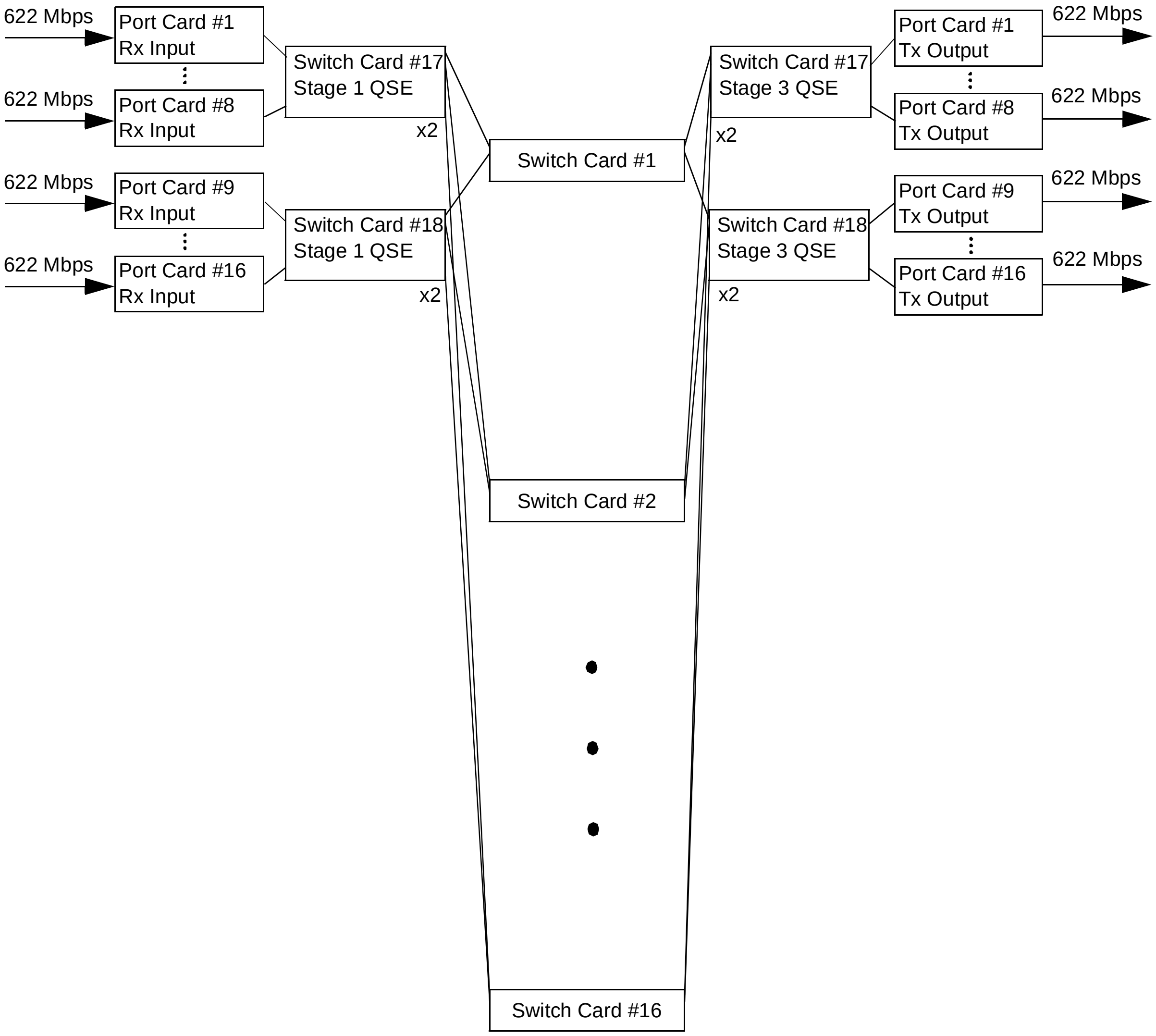


Figure 10. 10 Gbps ATM Switch

Figure 11 shows a 15 Gbps ATM switch using 24 port cards (24 QRTs) and 19 switch cards (38 QSEs).Here, once again, another switch card and eight port cards have been added

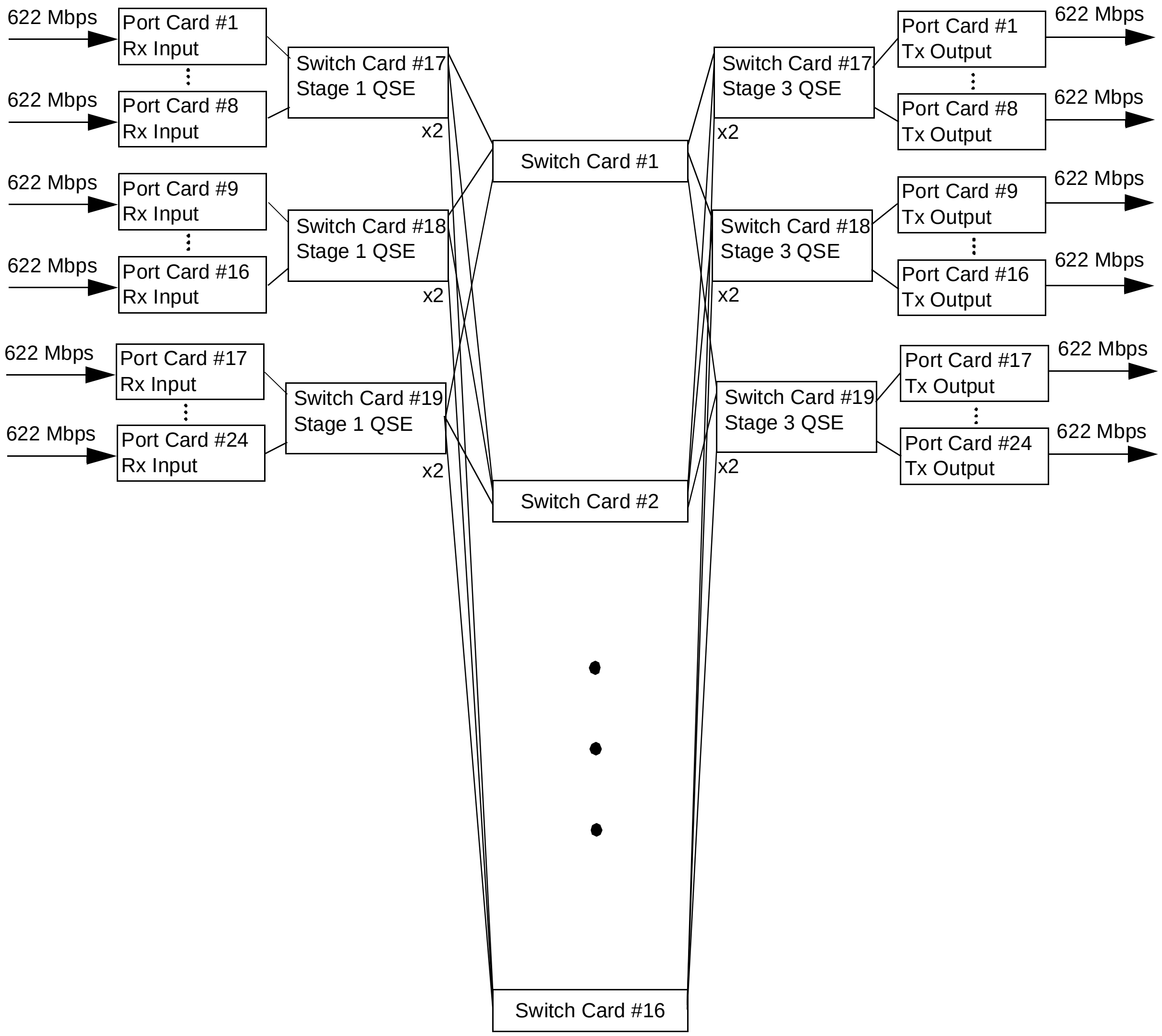


Figure 11. 15 Gbps ATM Switch

Figure 12 shows a 20 Gbps ATM switch composed of 32 port cards (32 QRTs) and 20 switch cards (40 QSEs). By adding additional sets of a switch card and eight port cards in the same manner, this system can scale up to 160 Gbps.

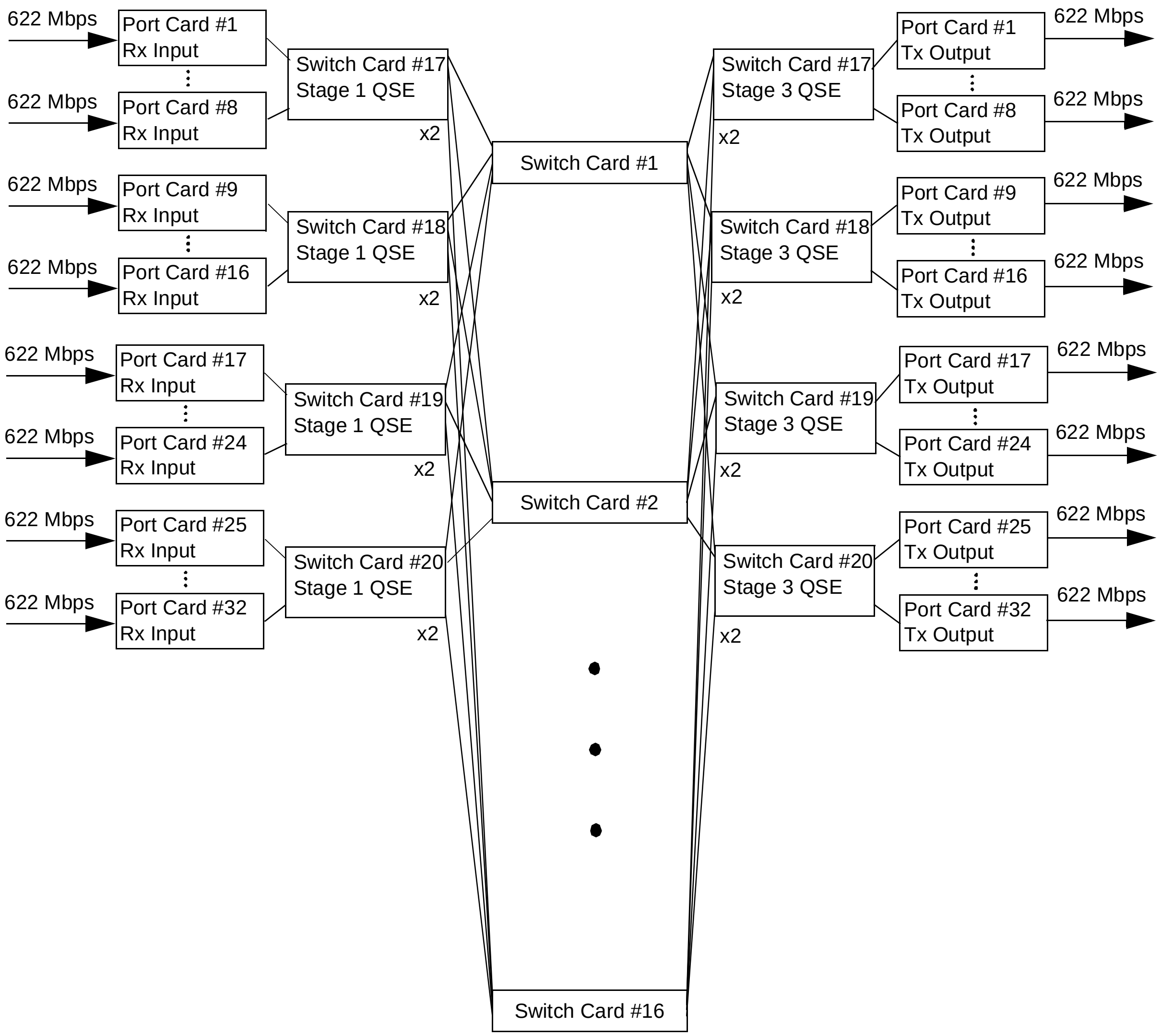


Figure 12. 20 Gbps ATM Switch

2 THEORY OF OPERATION

Multiple PM73488 QSEs can be combined to build a scalable switch fabric. The QSE switches data in the form of 118 nibble cells. The QSE has 32-input ports and 32 output ports, each containing a nibble-wide data interface, an SOC signal, and a backpressure/data-acknowledge signal.

Groups of 1, 2, 4, 8, 16, or 32 ports can be internally configured to act as a single aggregate port (also called gang) for unicast traffic. For multicast traffic, inputs and outputs can be grouped together in groups of 1, 2, or 4 ports. The input multicast grouping mode, output multicast grouping mode, and the unicast grouping modes do not need to be the same. Also, the QSE can be configured as a single 32 input × 32 output switch

The cell flow through the QSE has two separate data paths; one path for unicast cells and another path for multicast cells. Unicast cells are routed from one end of the switch fabric to the other end in a single cell time. In other words, no unicast cells are ever stored in the switch fabric. Unicast cells are stored only at the ingress and egress of the switch fabric. Multicast cells are routed in a store-and-forward manner. Each QSE can store up to 64 multicast cells. The QRT used as an interface to a switch fabric constructed with QSEs allows the construction of an ATM switch up to 160 Gbps.

A diagram of the QSE cell flow is shown in Figure 13. The unicast cell flow contains a routing stage that uses routing information from the cell header to determine the output group. The multicast cell flow contains an interface to an external SSRAM that contains the Multicast Port Vector (MPV) information for routing cells to multiple output groups.

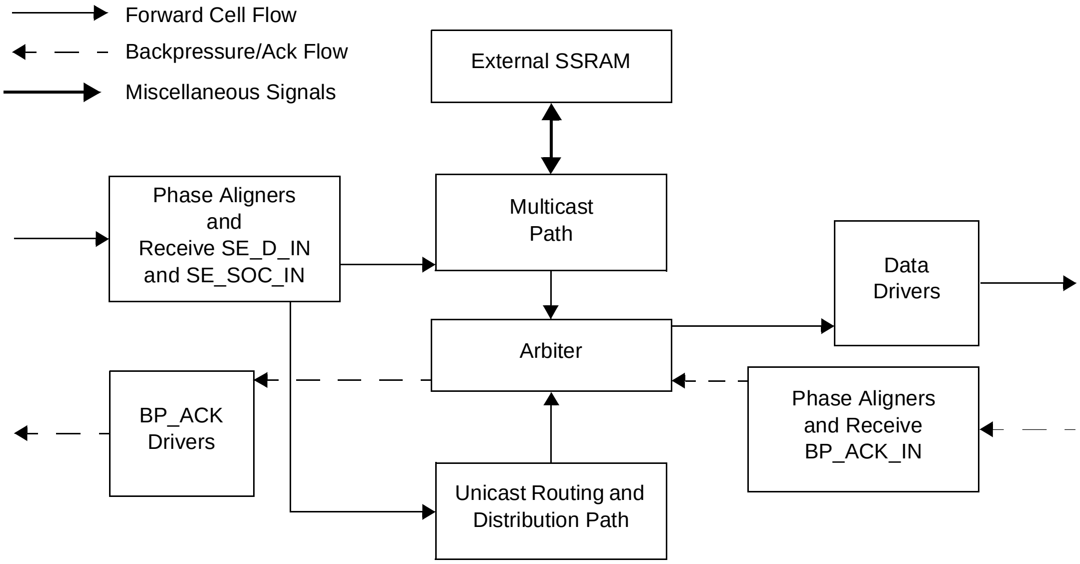

Forward Cell Flow
Backpressure/Ack Flow
Miscellaneous Signals
External SSRAM
Phase Aligners and Receive SE_D_IN and SE_SOC_IN
Multicast Path
Data Drivers
Arbiter
BP_ACK Drivers
Phase Aligners and Receive BP_ACK_IN
Unicast Routing and Distribution Path

Figure 13. Basic QSE Flow

2.1 Phase Aligners

Phase aligners aid in constructing large systems. Clock information is recovered from the data sent to each QSE switch fabric port. Phase aligners are used on the BP_ACK_IN(31:0), SE_SOC_IN(31:0), and SE_D_IN(31:0, 3:0) signal lines. Since there is no setup or hold time requirements on these signals, the overall clock distribution scheme within the system can be simplified. However, overall system jitter and skew between signals on the same switch fabric data port must still be managed.

2.2 Data Drivers

Another aid to constructing large systems is an elastic store at each QSE input data port. The data elastic store allows data arriving from different ports to be offset by up to a maximum of eight clock cycles. The internally generated and software programmable local CELL_START signal marks the end of an 8-clock-period window within which the SOC marker on each of the SE_SOC_IN(31:0) lines must arrive.

2.3 Unicast Routing and Distribution

Each of the 32 nibble-wide inputs is connected to an output by a crossbar. This crossbar is transparently controlled by the cell's routing tag, which specifies the input-to-output connection. In the event of a conflict for an output port, higher priority cells are given preference over lower priority cells. There are three unicast cell priorities: high, medium, and low.

The gang of 32, also known as distribution mode, is a special unicast routing mode in which incoming unicast cells are routed to outputs using PMC's patented congestion-minimization (Evil Twin Switching) algorithm. In this mode, no routing information is used from the cell's routing tag.

Depending on the gang mode, the QSE will need a number of routing bits to determine the output gang of a unicast cell. For example, in gang mode of four, there are eight output gangs, thus three routing bits are required for selecting the QSE. However, in distribution mode no routing bits are needed. The routing bits are taken from the head of the routing tag and are then shifted back in at the tail (which preserves header parity). This allows the next set of routing bits to be always accessible at the same spot in the tag, namely the head. The cell routing tag is broken into eight nibbles, namely TAG_0 through TAG_7.

Figure 14 on page 27 shows the tag rotation for gang mode of four (three routing bits are used by the QSE from TAG_0 and then shifted back in at the tail of TAG_7). TAG_0 is broken up and part of it ended up at the end of TAG_7 (shown by the white area in Figure 14 on page 27). As a result, all the other tags (TAG_1 through TAG_7) also get broken up and shifted (as shown by the light and dark gray areas of Figure 14 on page 27).

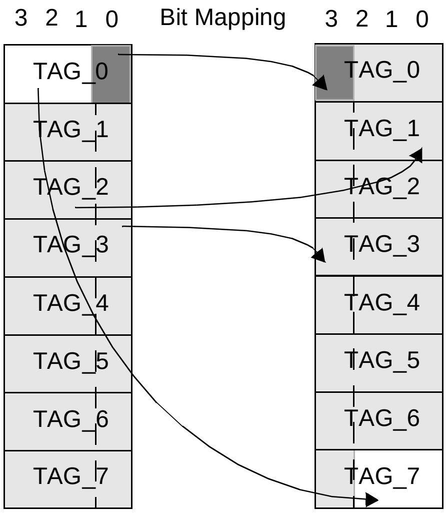


Figure 14. Routing Bits Rotation for Unicast Traffic, Gang Mode of Four

2.4 Multicast Cell Flow

There are 64 internal cell buffers for multicast traffic. These buffers are shared among three multicast priorities: high, medium, and low. These 64 buffers are grouped into two sets of 32-cell buffers each. One set is dedicated to ports 0 to 15, the other set to ports 16 to 32.

A multicast queue engine dynamically allocates the cell buffers to incoming multicast cells. Each cell is buffered until it can be sent out on all output ports to which it should be routed. These output ports are designated by a Multicast Group Vector (MGV) that is associated with a Multicast Group Index (MGI) carried by each multicast cell. Each QSE holds multicast MGVs in an MGV RAM. The QSE has internal RAM to support up to 128 MGVs. This support can be extended up to 256K MGVs by using an external MGV RAM.

Each multicast cell contains the RAM address of the MGV it is supposed to use. When a multicast cell is received, its MGV is fetched from RAM and copied to the MULTICAST_QUEUE_COMPLETION register. The MULTICAST_QUEUE_COMPLETION register tracks to which QSE ports the cell needs to be sent before its cell buffer can be cleared. In a multistage QSE fabric, each multicast cell will look up MGVs at each QSE. The MGV's sequence determines which output ports will finally receive the cell. The MGV structure allows software to create an optimal distribution tree for each multicast cell.

Multicast operation can be best understood by considering the QSE multicast path as two separate engines; the multicast queue engine and the multicast dequeue engine. The multicast queue engine queues cells into the multicast cell buffers (of which there are 64), and issues backpressure on the BP_ACK_OUT(31:0) lines. The multicast dequeue engine selects and dequeues cells from the buffers for output ports as guided by the backpressure received on the BP_ACK_IN(31:0) lines.

2.4.1 Multicast Queue Engine

The multicast queue engine associates input ports with cell buffers, computes backpressure for the input ports, and stores incoming cells into the buffers. In doing so, it guarantees:

- No input port will have more than three cells pending in the QSE — this can be changed to allow four pending cells by setting the “Allow 4 Bits Per Port” bit (bit 1) in the BP CONTROL register.
- No input port will have more than two high-priority cells pending.
- The sum of low- and medium-priority cells pending from any single input port will be less than 2.

In addition, the queue engine allows buffers to be reserved for high-priority cells or high/medium-priority cells. This is controlled by bits 2 and 3 of the BP_CONTROL_REGISTER (refer to section 9.3.29 “BP_CONTROL_REGISTER” on page 109). The four possible combinations for these two bits are listed in Table 1. The multicast queue engine will compute backpressure for the preceding QSE/QRT to ensure the constraints listed in Table 1 are satisfied. The same reservation policy applies to *both* sets of 32 buffers.

Table 1. BP_CONTROL_REGISTER; Threshold Control Bits for Each Set of 32 Buffers

Bit 3	Bit 2	Description
0	0	• Four buffers are reserved for high-priority cells. • Four buffers are reserved for high- or medium-priority cells. • All other buffers can be used by any cell.
0	1	• Four buffers are reserved for high-priority cells. • All other buffers can be used by any cell.
1	0	• Eight buffers are reserved for high- or medium-priority cells. • All other buffers can be used by any cell.
1	1	• All buffers can be used by any cell.

After the MGV address for the cell enters the QSE, the MGV associated with that cell is fetched and loaded into the QUEUE_COMPLETION_REGISTER (an internal register) as soon as possible.

2.4.2 Multicast Dequeue Engine

In each cell time, the multicast dequeue engine selects one multicast cell for each of the 32 output ports. In effect, all multicast cells wanting to go to a particular output port arbitrate among themselves to select the most appropriate port. Arbitration occurs independently for all 32 ports. The cells winning the internal multicast arbitration then compete with the incoming unicast cells for access to the output ports. Multicast arbitration winners are chosen to satisfy the following conditions in this sequence:

- Obey backpressure from the down stream QSE or QRT. Only cells with the allowed priorities will take part in arbitration.
- Higher priority cells win over lower priority cells.
- Cells that came in earlier win over cells that came in later (if they have the same priority).
- If multiple cells have the same priority and came in simultaneously, cells from a random input gang group will be selected.
- If multiple cells have the same priority, came in simultaneously, and belong to the same input gang group, the cell with the lowest port number will be selected.
- Ties are broken randomly.

This arbitration occurs among all cells in the cell buffers and occurs for all 32 ports. In effect, arbitration occurs for output ports in sequence, starting with cells arbitrating for port 0, then for port 1, and continuing on until port 31 (even though the actual implementation uses a parallel algorithm).

Multicast cells that have won this arbitration then compete with unicast cells for access to the output ports. In this contention, the cell with the highest priority wins and ties are broken randomly according to the programmable ratio set in the UC/MC_FAIRNESS_REGISTER (refer to section 9.3.6 “UC/MC_FAIRNESS_REGISTER” on page 97).

All these operations are optimized so that, in the absence of congestion, it is possible for a multicast cell to leave the QSE in the cell time immediately after it arrived.

As mentioned before, the queue completion register (32-bit vector) indicates the outputs to which each multicast cell needs to go. As a cell goes out on its desired outputs, the appropriate bits in the queue completion register are cleared. When all bits in the queue completion register have been cleared, the cell is deleted from the internal buffers and the buffer is reused for new incoming traffic.

Figure 15 shows an example of a high-priority cell preempting a cell in the multicast queue, and the resulting bit settings in the MULTICAST_QUEUE_COMPLETION_REGISTER (an internal register). (For the sake of simplicity, only 8 of the 32 outputs, and eight bits of the MGV_REGISTER (refer to section 9.3.4 “MULTICAST_GROUP_VECTOR_REGISTER” on page 96) and MULTICAST_GROUP_COMPLETION_REGISTER (an internal register) are shown.)

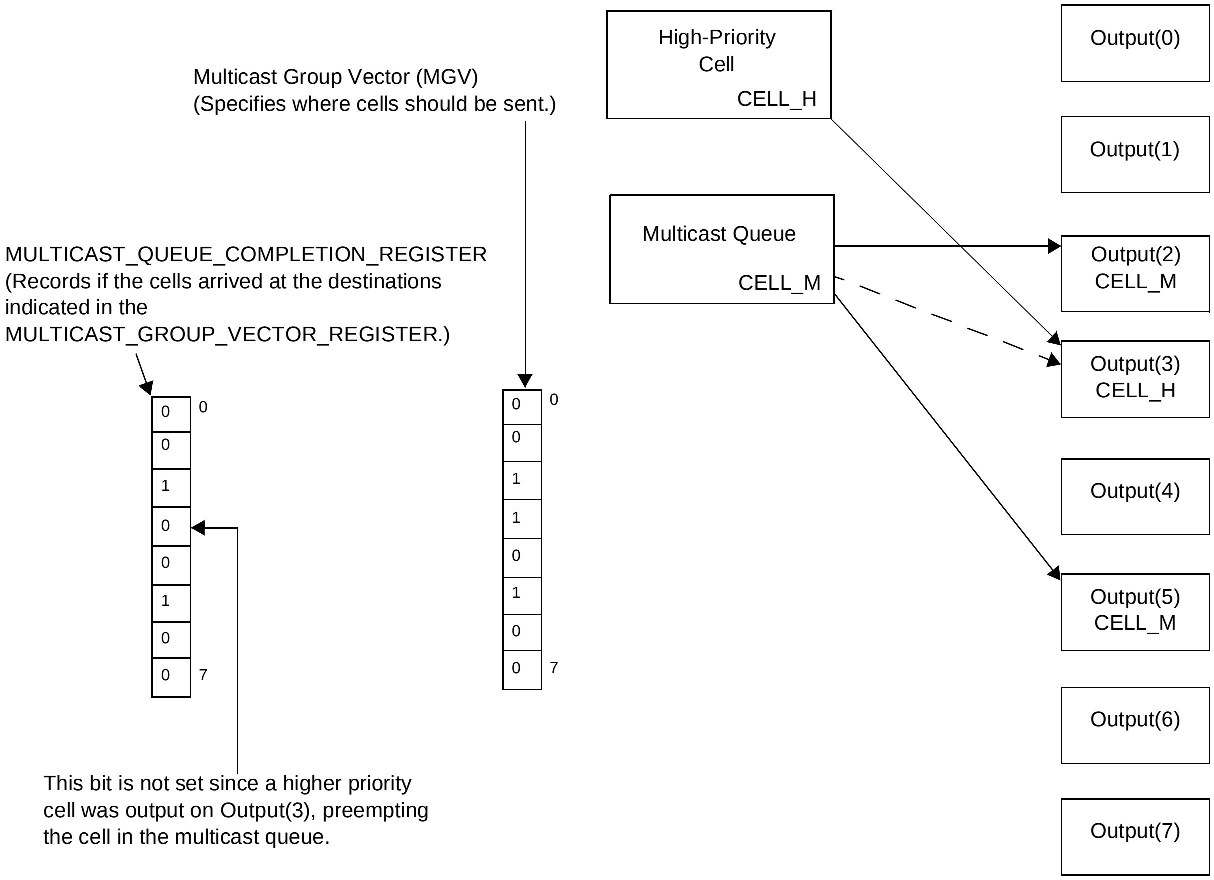

High-Priority Cell
CELL_H
Multicast Group Vector (MGV)
(Specifies where cells should be sent.)
Multicast Queue
CELL_M
MULTICAST_QUEUE_COMPLETION_REGISTER
(Records if the cells arrived at the destinations indicated in the MULTICAST_GROUP_VECTOR_REGISTER.)
0
0
1
0
0
1
0
0
0
7
0
0
1
1
0
1
0
0
0
7
Output(0)
Output(1)
Output(2)
CELL_M
Output(3)
CELL_H
Output(4)
Output(5)
CELL_M
Output(6)
Output(7)
This bit is not set since a higher priority cell was output on Output(3), preempting the cell in the multicast queue.

Figure 15. Example of Multicast Cell Handling in the QSE

2.5 Arbiter

The arbiter selects between unicast cells and multicast cells contending for the same output port. Higher priority cells are given preference over lower priority cells. If a multicast cell and unicast cell have the same priority, one cell is randomly chosen. The random choice can be biased in favor of either unicast cells or multicast cells at different points in the switch fabric by using the UC/MC_FAIRNESS_REGISTER (refer to section 9.3.6 “UC/MC_FAIRNESS_REGISTER” on page 97). In general, unicast cells should be favored at later stages in the switch fabric. Favoring unicast cells is necessary in multiple-stage switch fabrics since unicast cells are routed in a cut-through fashion and multicast cells are routed in a store-and-forward fashion. As such, a unicast cell becomes more “valuable” as it proceeds further in the switch fabric, since it did so at the expense of other cells.

For example, consider a congested 3-stage switch fabric where unicast cells and multicast cells of equal priorities collide at each stage in the fabric, without any biasing. A unicast cell must make it from ingress to egress in one cell time and the chances of doing so would be a little more than $(1/2)^3$ = 12.5%. However, each multicast cell would have a 50% chance of advancing to the next stage in the switch fabric.

2.6 BP_ACK Drivers

The BP_ACK_OUT(31:0) lines are used to send information from a QSE to upstream QSEs or QRTs. These lines are used to send two types of information:

- Backpressure information (for unicast cells).
- Transmit acknowledge information (for multicast cells).

Backpressure information is sent for multicast cells. This information indicates to an upstream QRT or QSE if the QSE can accept another multicast cell in the next cell time. Backpressure information also indicates what multicast cell priorities the QSE can accept.

Cell transmit acknowledge information is sent for unicast cells. This information signals whether or not the unicast cell transmitted in the current cell time made it to its destination QRT. If the cell makes it to the destination QRT, an Acknowledgment (ACK) is sent. If the cell has been dropped in the switch fabric, information is sent back indicating if the cell was dropped internally Mid Switch Negative Acknowledgment (MNACK) or at the output of the switch fabric Output Negative Acknowledgment (ONACK). The MNACK and ONACK is used by the QRT to determine when to retry sending the given cell.

2.7 Interdevice Interconnectability

All input and output ports can be configured in groups of four to directly connect to either QRT devices or other QSE devices. This allows considerable flexibility in the switch fabric types and sizes that can be constructed using the entire PMC chip set.

2.8 Network Topologies and the Speedup Factor (SF)

For many switch fabric architectures using the QSE, a single metric called the Speedup Factor (SF) allows comparison of different network topologies, which is independent of traffic load and type. The SF also allows for predictions about the network performance.

Before describing the SF metric, we will briefly discuss the network philosophy and the different network topologies.

2.8.1 Network Philosophy

Given current technology, to scale through 160 Gbps, a network must be distributed and use buffers at the network inputs and outputs. In an ideal world, crossbars of any arbitrary size could be built to provide connectivity for the network inputs and outputs. Additionally, there would be a central “brain”, or global arbiter, to control the input buffers and schedule cells optimally for routing in the network, as shown in Figure 16.

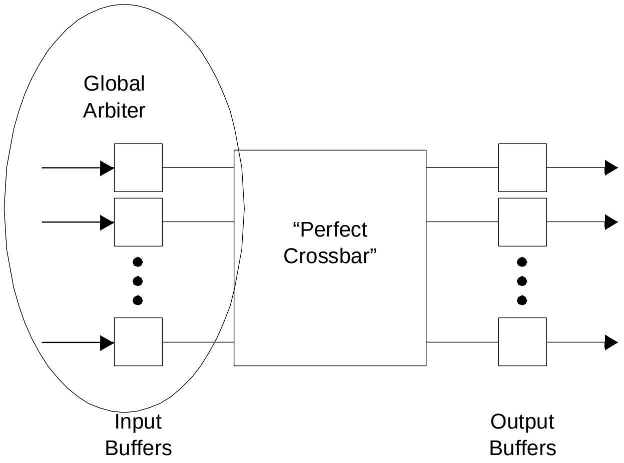


Figure 16. Ideal Distributed Network

Unfortunately, given real constraints, it is not possible to have a global arbiter wired to each input that has knowledge of all cells in the system, and can quickly make optimal decisions about routing. Thus, each input must make decisions using knowledge local to its buffers. This results in the possibility of collisions at the network outputs, even though it is a “perfect” crossbar, as shown in Figure 17.

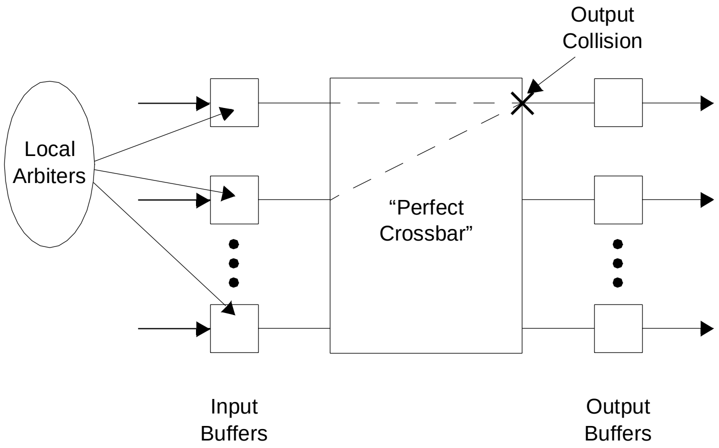


Figure 17. More Realistic Distributed Network

Replacing the idealized crossbar with a buildable, traditional Banyan network increases the possibility of internal network collisions, as shown in Figure 18. Given a particular Banyan network, one can always find a large class of traffic patterns that will cause many internal collisions. For large Banyan networks, the collision problem is greatly increased.

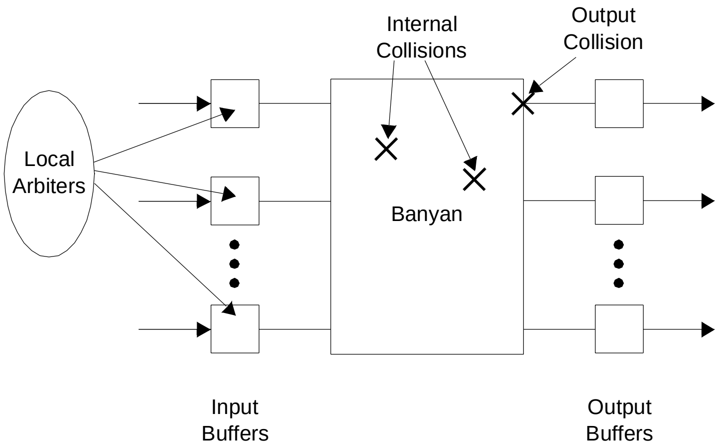


Figure 18. “Large” Distributed Network (Will not Work Well with Banyan Alone)

To reduce internal collisions in the traffic-dependent Banyan networks, the QRT/QSE network adds a distribution/ randomizing network (shown in Figure 19) that uses a patented intelligent configuration algorithm, known as Evil Twin Switching. The algorithm (described in section 2.8.3 “Speedup Factor (SF)” on page 36) allows lower-bounding the network performance, independent of traffic patterns.

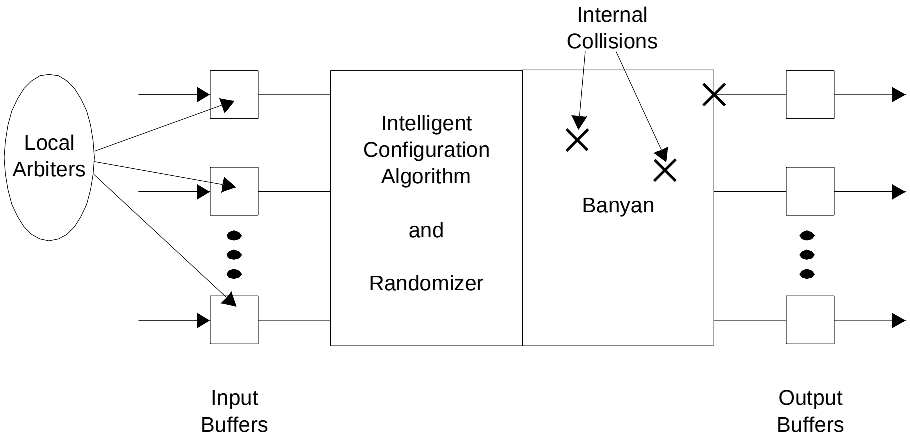


Figure 19. High-Level QRT/QSE System

To overcome the inefficiencies caused by collision in the network, the fabric must be run at a rate greater than line rate. The speedup factor is the minimum rate necessary to guarantee that the network is no longer the system bottleneck. Note that in this case, the network efficiently moves data from the input to the output buffers, and the switch performs similar to a purely output buffered switch.

2.8.2 Network Definition

A large range of switch fabrics can be described as follows: with the following notation: “p” refers to the number of fabric planes, and “x,” “y,” and “z” refer to the routing tag size necessary to make routing decisions in the Banyan section of the network to route cells to the correct output port. This is summarized as follows:

(z)xp — 1-stage network
(y,z)xp — 3-stage network

Hence, the (3) × 1 network shown in Figure 20 refers to a single switch stage, and three routing bits are required to select from one of the eight output port groupings. (Recall that the QSE has 32 output ports that can be configured in groups of 1, 2, 4, 8, 16, or 32. In Figure 20, they are configured in groups of four. The input and output buffers provided by the QRT have four input ports and four output ports to the switch fabric, and are logically broken into the input half of the QRT (IRT) and output half of the QRT (ORT) for convenience.

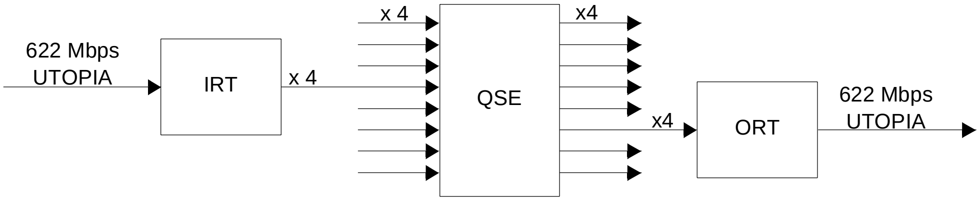


Figure 20. (3) x 1 - 5 Gbps System

The (5) × 4 network shown in Figure 21 is an example of a network with four parallel planes. It demonstrates the flexibility allowed because the QRT has four input and output ports. In this case, randomization is performed in the IRT.

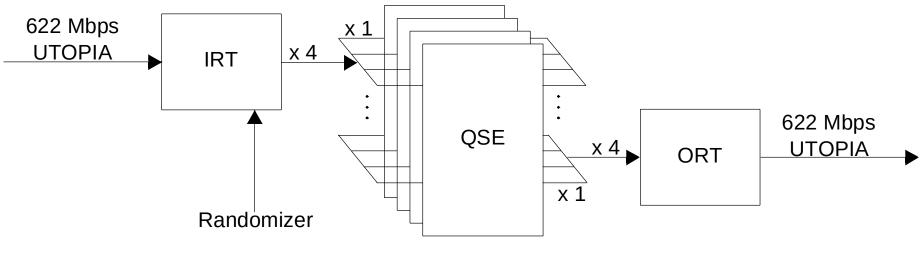


Figure 21. (5) x 4 - 20 Gbps System

In Figure 22, the first stage of QSEs is configured to provide the required randomization, and the next two switch stages route the cells to the final port destination. The second QSE stage needs only to make an "up" or "down" decision requiring a single routing bit, while the third QSE stage needs to select between eight QRTs, requiring three routing bits.

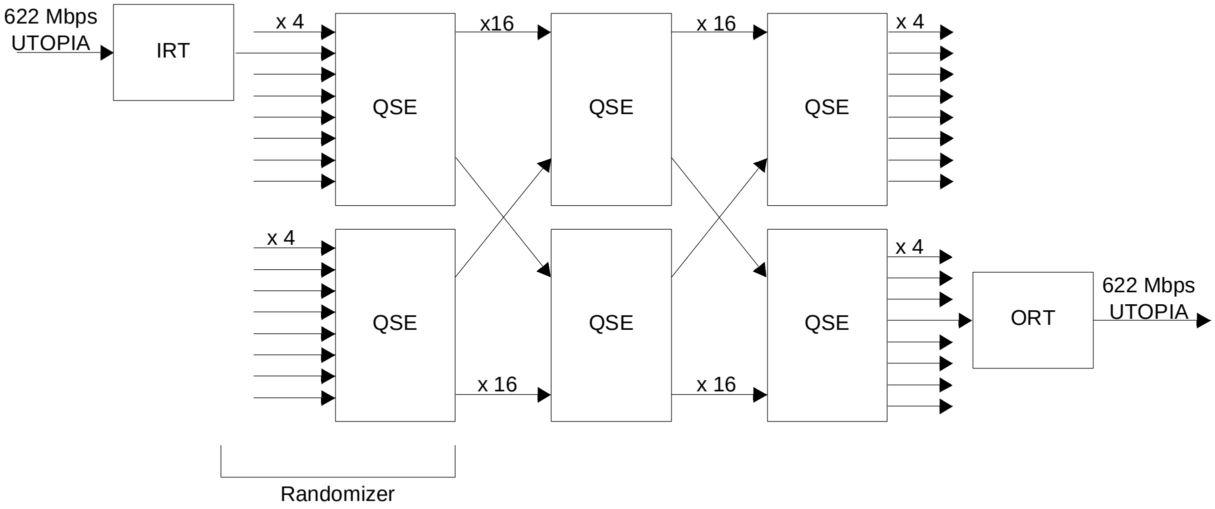


Figure 22. (1,3) x 1 - 10 Gbps System

2.8.3 Speedup Factor (SF)

If the traffic pattern presented to a particular Banyan network results in many internal collision, a shuffling pattern exists that has been proven to result in few internal collisions. Although a purely random reshuffling results in good behavior, we can lower-bound network performance by using randomization along with the Evil Twin Switching algorithm as shown in Figure 23. This algorithm is as follows: randomly choose a configuration, route data, choose the dual or Evil Twin Switching configuration, route data, and repeat. This algorithm minimizes the number of internal collisions. In 3-stage networks, the first stage of the QSEs provide this functionality.

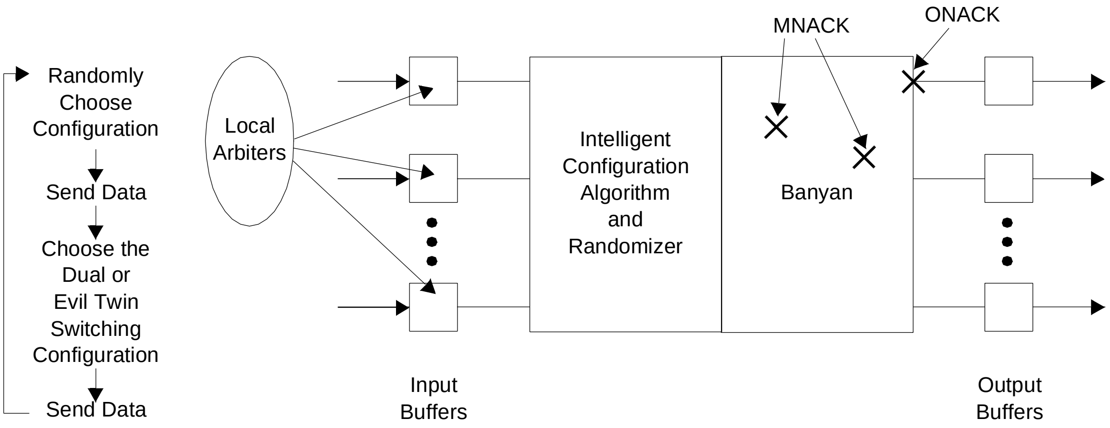

Randomly Choose Configuration
Send Data
Choose the Dual or Evil Twin Switching Configuration
Send Data
Local Arbiters
Intelligent Configuration Algorithm and Randomizer
Banyan
MNACK
ONACK
Input Buffers
Output Buffers

Figure 23. Randomizer (with Evil Twin Switching Algorithm)

Even with a perfect crossbar for a network, there are still output collisions, and despite the Evil Twin Switching algorithm, there are still internal collisions (albeit fewer). Thus, multiple routing attempts must be made per cell to yield full throughput. This can be accomplished by running the switch fabric at a faster clock rate than the buffering logic.

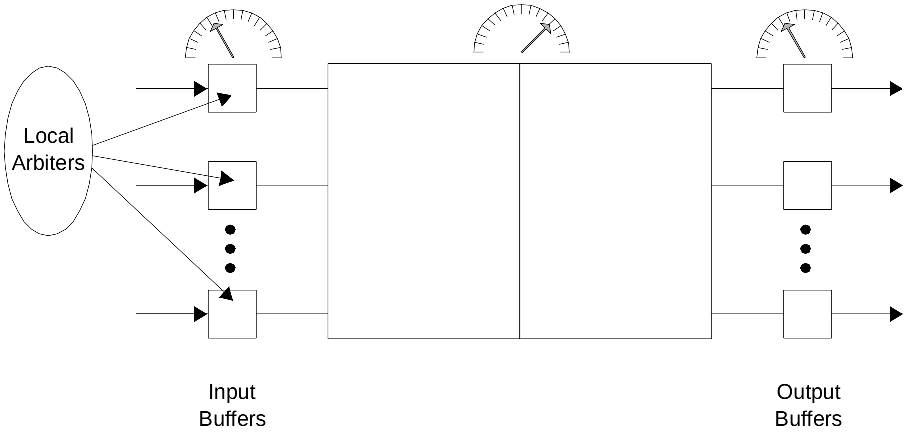


Figure 24. Network Needs to be Run Faster than the Line Rate

The chance for internal collisions increases as the network load increases, and the exact behavior varies with network topology. An example of this behavior is shown in Figure 25 and the SF is inferred from the limiting case where the network is fully loaded.

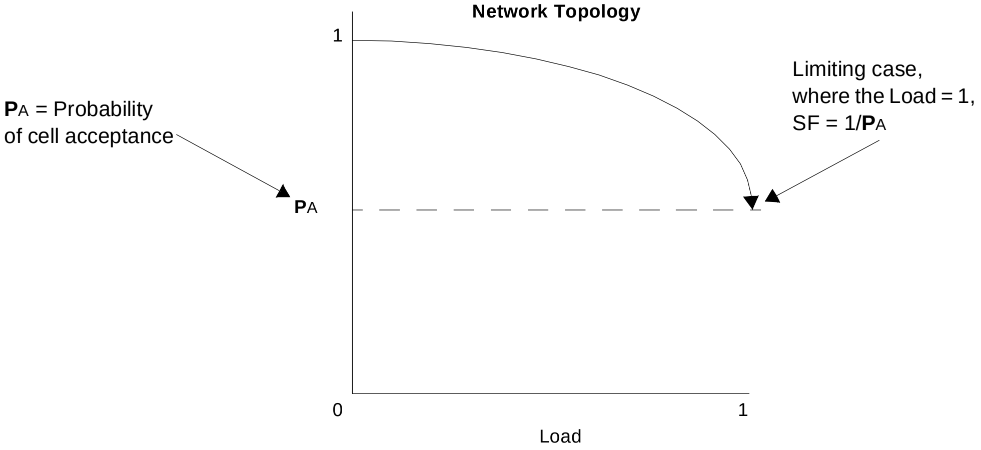


Figure 25. Definition of the Speedup Factor

Given this notion of SF, "how much faster is fast enough?" Theoretical models and simulations can answer that question. Given that the switch fabric can be run at a certain clock rate relative to the buffering logic, we can know which networks to choose to prevent the network from becoming a bottleneck.

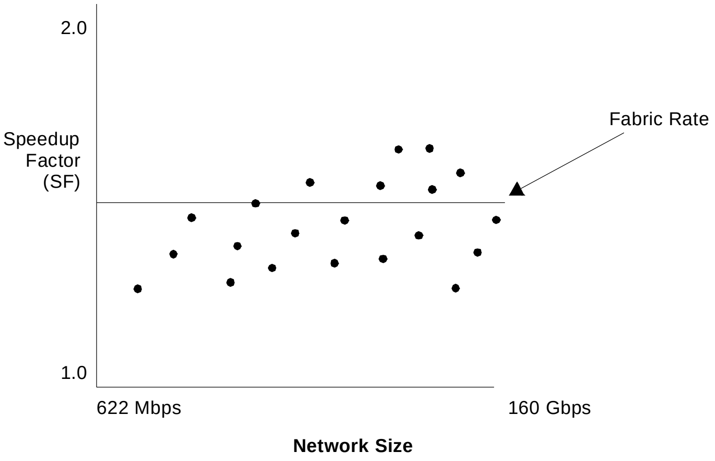


Figure 26. How to Use the SF to Select Favorable Networks

Table 2, Table 3 show all of the 1-, 3-stage network topologies requiring an SF of less than 1.6, which is the maximum speedup allowed by the actual implementation.

Table 2. Speedup Factor (1-Stage Networks)

Network	Size (Gbps)	Speedup Factor (SF)	Number of QSEs
(3) × 1	5	1.22	1
(4) × 2	10	1.36	2
(5) × 4	20	1.57	4

Table 3. Speedup Factor (3-Stage Networks)

Network	Size (Gbps)	Speedup Factor (SF)	Number of QSEs
(1,3) × 1	10	1.28	6
(1,4) × 2	20	1.41	12
(2,3) × 1	20	1.32	12
(2,4) × 2	40	1.46	24
(3,3) × 1	40	1.39	24
(3,4) × 2	80	1.53	48
(4,3) × 1	80	1.49	48

3 EXTERNAL PORT DESCRIPTIONS

3.1 Switch Fabric Port and Interface Description

Each port is a 6-bit interface consisting of:

- a nibble-wide data interface (SE_D_IN and SE_D_OUT),
- an SOC signal (SE_SOC_IN and SE_SOC_OUT), and
- a backpressure/data acknowledge signal (BP_ACK_IN and BP_ACK_OUT).

3.1.1 SE_SOC Encodings

The SE_SOC encodings (SE_SOC_IN(31:0), SE_SOC_OUT(7:0)) provide guaranteed transitions and SOC encodings.

The SE_SOC signals carry a repeating four “0s” and four “1s” pattern to guarantee transitions required by the phase aligner. The SOC signal on data lines associated with an SE_SOC line is indicated by a break in this repeating pattern. The SOC is a single “1” followed by five “0s”. Figure 27 shows the guaranteed transitions. Figure 28 provides an expanded view of the signal transitions and the first nibble after the SOC pulse (nibble #0) corresponds to nibble “0” in Table 5 on page 40.

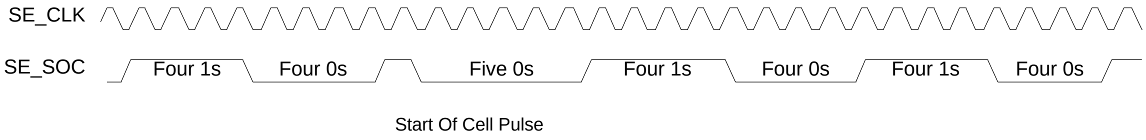


Figure 27. SE_SOC Encodings

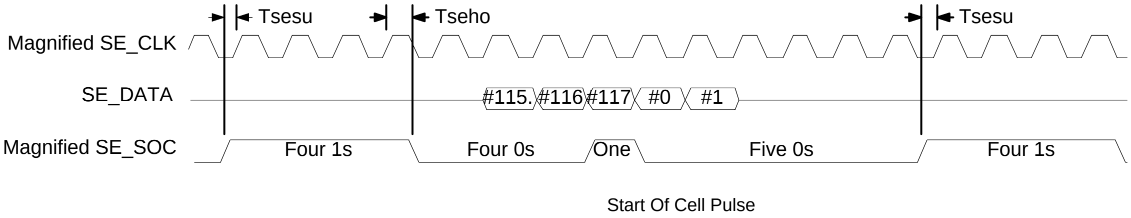


Figure 28. Expanded SE_SOC Encodings

3.1.2 Data Cell Format

The regular cell format is shown in Table 5.

Table 5. Regular Cell Format

Nibble	Symbol	Definition	Comment
0	PRES(1:0), MC, SP	Pres = 10_b: Cell present. 01_b: Cell not present (See Table 6 on page 41). 00_b: Cell assumed to be not present (failure). 11_b: Cell assumed to be not present (failure). MC = 1_b: Multicast Cell. SP Spare bit.	The spare bit is not interpreted or used by the QSE.
1	SP(1:0), PRIORITY(1:0)	SP(1:0) Spare bits. Priority = 11_b: High-priority cell. 10_b: Medium-priority cell. 01_b: Low-priority cell. 00_b: Undefined. Cell discarded by QSE.	Priority for the switching fabric. The QRT should be configured never to generate priority 00_b cells, since they are discarded by the QSE. The spare bits are not interpreted or used by the QSE.
2	TAG_0	Routing tag 0 or MULTICAST_GROUP_INDEX(15:12). Refer to section 9.3.3 “MULTICAST_GROUP_INDEX_REGISTER” on page 96.	
3	TAG_1	Routing tag 1 or MULTICAST_GROUP_INDEX(11:8). Refer to section 9.3.3 “MULTICAST_GROUP_INDEX_REGISTER” on page 96.	
4	TAG_2	Routing tag 2 or MULTICAST_GROUP_INDEX(7:4). Refer to section 9.3.3 “MULTICAST_GROUP_INDEX_REGISTER” on page 96.	Interpretation of TAG_5:0 depends on whether or not the cell is a multicast cell.
5	TAG_3	Routing tag 3 or MULTICAST_GROUP_INDEX(3:0). Refer to section 9.3.3 “MULTICAST_GROUP_INDEX_REGISTER” on page 96.	
6	TAG_4	Routing tag 4 or MULTICAST_GROUP_INDEX(23:20).	Currently, QSE supports only 256K multicast group vectors, i.e. it only uses multicast group index(17:0). Therefore, bits 23:20 are ignored.
7	TAG_5	Routing tag 5 or MULTICAST_GROUP_INDEX(19:16). Refer to section 9.3.8 “MULTICAST_GROUP_INDEX_MSB” on page 98.	Currently, QSE supports only 256K multicast group vectors, i.e. it only uses multicast group index(17:0). Therefore, bits (19:18) are ignored.
8	TAG_6	Routing tag 6.	

Table 5. Regular Cell Format (Continued)

Nibble	Symbol	Definition	Comment
9	TAG_7	Routing tag 7.	
10	OUTCHAN_3	Interpreted as OUTCHAN(15:12) by a QRT.	Not used by the QSE.
11	SP(1:0), MB, PARITY	SP(1:0) Spare bits. MB Mark bit: Cells that are present and have this bit set are counted by the INPUT_MARKED_CELL_COUNT (refer to section 9.3.11 “INPUT_MARKED_CELLS_COUNT” on page 99) and OUTPUT_MARKED_CELL_COUNT (refer to section 9.3.12 “OUTPUT_MARKED_CELLS_COUNT” on page 99) counters. P Should be odd parity over nibbles 0 to 11.	
12	OUTCHAN_2	Interpreted as OUTCHAN(11:8) by a QRT.	Not used by the QSE.
13	OUTCHAN_1	Interpreted as OUTCHAN(7:4) by a QRT.	Not used by the QSE.
14	OUTCHAN_0	Interpreted as OUTCHAN(3:0) by a QRT.	Not used by the QSE.
15	VCI_3	Interpreted as Virtual Channel Identifier (VCI)(15:12) by a QRT.	Not used by the QSE.
16	VCI_2	Interpreted as VCI(11:8) by a QRT.	Not used by the QSE.
17	VCI_1	Interpreted as VCI(7:4) by a QRT.	Not used by the QSE.
18	VCI_0	Interpreted as VCI(3:0) by a QRT.	Not used by the QSE.
19	PTI(2:0)/CLP	Interpreted as the Payload Type Indicator (PTI) and Cell Loss Priority (CLP) fields from the cell by a PM73487A.	Not used by the QSE.
20	SEQ_1	Interpreted as SEQ(7:4) by a QRT.	Not used by the QSE.
21	SEQ_0	Interpreted as SEQ(3:0) by a QRT.	Not used by the QSE.
22-117	Payload	Interpreted as 48 bytes of ATM cell payload by a QRT.	Not used by the QSE.

The idle cell format is shown in Table 6. The idle cell format is chosen to make the interface robust to both stuck-at faults, as well as bridging faults on the data lines.

Table 6. PM73488 Mode Idle Cell Format

Nibble	Symbol	Definition	Comment
0	Pres(3:0)	Pres = 0100_b: Cell not present.	
1	IDLE_0	IDLE_0 = 0000_b: All 0.	
2	IDLE_1	IDLE_1 = 1000_b: Marching 1 pattern, which protects against bridging faults.	

Table 6. PM73488 Mode Idle Cell Format (Continued)

Nibble	Symbol	Definition	Comment
3	IDLE_2	IDLE_2 = 0100_b: Marching 1 pattern, which protects against bridging faults.	
4	IDLE_3	IDLE_3 = 0010_b: Marching 1 pattern, which protects against bridging faults.	
5	IDLE_4	IDLE_4 = 0001_b: Marching 1 pattern, which protects against bridging faults.	
6	IDLE_5	IDLE_5 = 0000_b:	
7	IDLE_6	IDLE_6 = 0000_b.	
8-15	Reserved	(QSE currently outputs 0000_b.)	
16-117	Unused	(QSE currently outputs 0000_b.)	

3.1.3 BP_ACK Encodings

The BP_ACK encodings (BP_ACK_IN and BP_ACK_OUT) guarantee transitions, and BP and ACK encodings are shown in Figure 29.

The BP_ACK signal is used to signal backpressure/cell acknowledgment to the previous stage. To ensure the transitions required by the phase aligner, this line carries a repeating four “0s” and four “1s” pattern. The actual information is transferred by a break in this pattern (shown by BP_ACK signaling in Figure 29). The pattern break is identified by a bit inversion (Inversion 1) on the line, followed by a mode, and two data bits, followed by a second inversion (Inversion2) of the expected bit, if the previous pattern had continued. This is followed by the last two bits. After these information bits, the repeating pattern restarts with four “0s”.

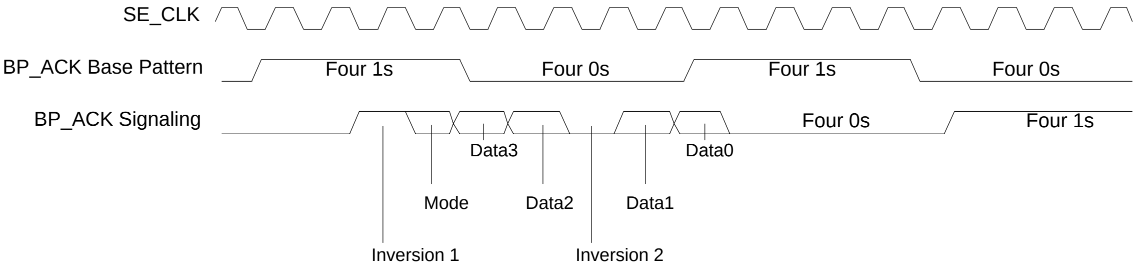


Figure 29. BP_ACK Encodings

The information bits encoding is described in Table 7.

Table 7. Information Bit Encoding

Mode	Data 3	Data 2	Data 1	Data 0	Description
0	1 = Backpressure on high-priority multicast cell.	1 = Backpressure on medium-priority multicast cell.	1 = Backpressure on low-priority multicast cell.	0	Backpressure information. This signal is present each cell time regardless of whether a cell was transmitted or not (on that link). This signal is withheld if any problem is detected on the input port.
1	0	0	0	0	Unassigned.
1	0	1	0	0	Signals MNACK.
1	1	0	0	0	Signals ONACK.
1	1	1	0	0	Signals ACK.

3.2 Data Acknowledge

The data acknowledge signals (BP_ACK_IN and BP_ACK_OUT) are used to indicate if, at the current cell time, a cell was successfully transmitted or not. Data acknowledge is a single line per port that returns from a cell's destination in the reverse direction from the data flow. If the cell is being blocked by the switch, this information is generated directly by the QSE. If the cell is not being blocked by the switch, this information is forwarded from the next switch stage.

The data acknowledge signal provides the following information to the QRT:

- The cell was successfully received by the QRT at the cell destination (ACK).
- The cell was not accepted by the QRT at the cell destination (does not happen by design in the PM73487).
- The cell was blocked by the switch at the output of the switch fabric (refer to section 9.3.30 "ACK_PAYLOAD" on page 109).
- The cell was blocked internal to the switch fabric (refer to section 9.3.30 "ACK_PAYLOAD" on page 109).
- The cell was detected as a parity error cell by a QSE (refer to section 9.3.30 "ACK_PAYLOAD" on page 109).
- The cell was headed to a gang of which all ports are dead (refer to section 9.3.31 "GANG_DEAD_ACK_PAYLOAD" on page 110).

Thus, direct information is provided to the QRT on a per-cell basis and on a per-VC basis.

The QSE behavior to support the above scenario is as follows:

- If the cell was a parity errored cell, and the QSE is configured to check parity in the CHIP_MODE register (refer to the field labeled "PARITY_CHECK" on page 95), then the parity acknowledge in the ACK_PAYLOAD register is sent (the default is ONACK).
- If the cell is dropped due to congestion at an output of the QSE, then Ack Payload for cells dropped due to congestion in the ACK_PAYLOAD register is sent (bits3:0). Refer to bits 3:0 in section 9.3.31 "GANG_DEAD_ACK_PAYLOAD" on page 110.

- If the cell was blocked at an output of the QSE because the entire gang is disabled (the default is ACK), then the cell is to be cleared when all ports to a QRT are known to be unavailable.
- If the cell was successfully routed through the QSE, the return path is set up to route the data-acknowledge signal back from the next switch stage.

For multicast traffic, the BP_ACK_IN and BP_ACK_OUT signals also serve as a backpressure signal, indicating at each cell time, the multicast cell priority the QSE can accept on the following cell time on a given port.

3.3 Microprocessor Interface

The QSE has a non-multiplexed, asynchronous, general-purpose microprocessor interface (PIF) through which the internal registers can be accessed. The external SSRAM is also indirectly accessed through this same interface.

3.4 Multicast SRAM Interface

The QSE supports 128 internal multicast groups, and is expandable up to 256K through an external SSRAM.

3.5 Clocks and Timing Signals

The QSE is driven from a single clock source up to a maximum clock rate of 66 MHz.

To indicate the SOC, there is one SE_SOC_IN signal per input port. There is one SE_SOC_OUT signal per group of four outputs.

Cells must arrive at the input ports within an eight clock-cycle window. A CELL_START is used as a reference for an internal cell start signal to determine the eight clock-cycle window in which the SOC signal on the SE_SOC_IN lines are valid. The internal cell start signal delay from the external CELL_START signal is programmed in the CELL_START_OFFSET (refer to section 9.3.28 “CELL_START_OFFSET” on page 109).

3.6 CTRL_IN

CTRL_IN is a one bit input port. Its function depends on the value of the “ENABLE_STAT_PINS” (bit 7) bit in the CHIP_MODE register. When this bit is “0”, CTRL_IN directly sets the value of the internal “No Data Out” control bit. What this internal bit does is explained later. When this bit is “1”, CTRL_IN expects a data packet which sets the value of both, the internal “/No Data Out” and the “/No Data In” registers.

The format for the data packet is described below:

Data on this line has to be clocked out by its source at one-eighth the QSE clock rate. CTRL_IN is normally “0”. A valid data packet starts with a “0” -> “1” transition on the line (implying that the first “bit” of the data packet is “1”). A valid data packet starts with “100_b” followed by 2 control bits and 6 bits which are ignored. If the first 3 bits of a data packet are not “100_b”, the data packet is ignored (i.e. the next 8 bits are ignored). Data packets may not arrive back to back. At least 4 zero bits must be present between any two data packets.

A valid data packet is therefore: “$100b_0b_1XXXXXX_b$”. b_0 is the desired value of “/No Data In” and b_1 is the desired value of “/No Data Out”.

If the internal “/No Data In” bit is asserted the QSE will continously apply back pressure on all inputs and all priorities. If the internal “/No Data Out” bit is asserted the QSE will behave as if all its outputs are receiving backpressure on all priorities.

3.7 STAT_OUT

This is a bidirectional port whose function depends on the value of the "ENABLE_STAT_PINS" (bit 7) bit in the CHIP_MODE register. When this bit is "0", STAT_OUT is configured as an input port and directly sets the value of the "No Data In" internal register (see CTRL_IN description above for what this internal register does). When this bit (ENABLE_STAT_PINS) is "1", STAT_OUT is configured as an output and periodically outputs an information packet which indicates whether the internal multicast buffers are empty.

STAT_OUT is normally "0" and the information packet generated on the STAT_OUT pin is 5 bits long and is clocked out using the QSE clock. The pattern starts with a "1" and the 5 bits are "$10b_0b_1b_2$" including the "1" that starts it all. If $b_0b_1b_2$ is "000", then it means that all the multicast buffers are empty. If $b_0b_1b_2$ is any other three-bit value, then it means that the multicast buffers are not empty. Note that this packet represents the *instantaneous* status of the multicast buffers. Therefore, if a multicast cell is entering or exiting the chip at just about the time the packet is being output, then the information in the packet must be interpreted with caution. However such delicate race conditions are not a problem in practice. (See "Fabric Switch-Over" on page 45.)

3.8 Fabric Switch-Over

The reason /NO_DATA_IN, /NO_DATA_OUT and STAT_OUT exist is to support hitless fabric switch-over. This means that we wish to detour traffic to a back-up fabric and take the current fabric down for repairs, all without losing a single cell. This can be accomplished in several different ways. We suggest a possible scheme below. Our scheme only uses the /NO_DATA_IN and STAT_OUT features. Other schemes may also use the /NO_DATA_OUT feature.

There are two fabrics, A and B. Each fabric has two kinds of inputs: data_in and bp_ack_in. Assume that these inputs are duplicated to both fabrics. Each fabric also has two kinds of outputs: data_out and bpack_out. Assume that there are muxes that can choose outputs either from fabric A or from fabric B. At any point in time, all muxes must select A, or all muxes must select B, i.e. all muxes must switch in lock-step. Initially, we are using fabric A, and B is the back-up. Thus all muxes are set to choose A. At the end of the process, we want to be using fabric B, with A being the back-up. During the process, no cell must get lost, and there should be no ordering violations.

- Assert /NO_DATA_IN on both A and B. For unicast, the result is that both A and B will reject cells and return nacks. For multicast, the result is that both A and B will assert full back-pressure. Of course, only the nacks and back-pressure from A will reach the ingress QRTs, because the muxes are set to choose A.
- Effectively, the QRT will not be able to deliver even a single cell. All unicast cells will be attempted, but they will bounce back with nacks. Multicast cells can't even be attempted because of full back-pressure.
- Wait for STAT_OUT to go to "000" on all QSEs on both fabrics. This indicates that all multicast cells that were in transit in the fabrics have drained out. Of course, only the cells from A will reach the egress QRTs, because the muxes are set to choose A.
- At a cell time boundary, switch all muxes to choose B.
- Now deassert /NO_DATA_IN on both A and B. Cells will start flowing through B, and A can be taken down safely for maintenance/repair.

3.9 Cell Timing/Latency

The data latency through each QSE depends on the distribution mode. The maximum data latency is listed in Table 8.

Table 8. Data Latencies

Aggregate Mode	Latency
1	13 clock cycles
2, 4, 8, 16, 32	10 clock cycles

The data acknowledge through each QSE is a maximum of five clock cycles.

4 QSE FEATURE DESCRIPTIONS

4.1 Distribution Algorithm

The QSE has an algorithm that allows unicast cells to take advantage of multiple paths in multistage switch fabrics. This algorithm is run simultaneously by all QSEs in a system. Since the position (row and column) of each QSE is known (refer to section 9.3.26 “SWITCH_FABRIC_ROW” on page 107 and to section 9.3.27 “SWITCH_FABRIC_COLUMN” on page 108), and they all receive a synchronizing strobe (CELL_24_START), each QSE can determine exactly what the other QSEs are doing. This enables the QSEs to act globally to minimize cell congestion in the switch fabric.

4.2 Cell Start Offset Logic

Each QSE needs to be informed when the window occurs during which the SE_SOC_IN is valid for the input ports. Generally, since this window can vary from one QSE to another in the fabric, it is made software programmable by setting the CELL_START_OFFSET register (refer to section 9.3.28 “CELL_START_OFFSET” on page 109). The significance of this register is as follows: The QSE generates an internal signal called "Local CELL_START", which is simply a delayed version of external CELL_START input, where the delay is the number of clock cycles given in the CELL_START_OFFSET register. The valid window for accepting SE_SOC_IN is the 8-clock-cycle interval immediately preceding the pulse of local CELL_START signal. (For a detailed timing diagram, see “Relation Between External CELL_START and Local CELL_START” on page 47.)

4.2.1 Relation Between External CELL_START and Local CELL_START

Figure 30 shows the relationship between the external CELL_START signal and the local CELL_START signal, which is used internally by the QSE. The signal offset is programmable through the microprocessor interface (refer to section 9.3.28 “CELL_START_OFFSET” on page 109) to allow for easy system synchronization.

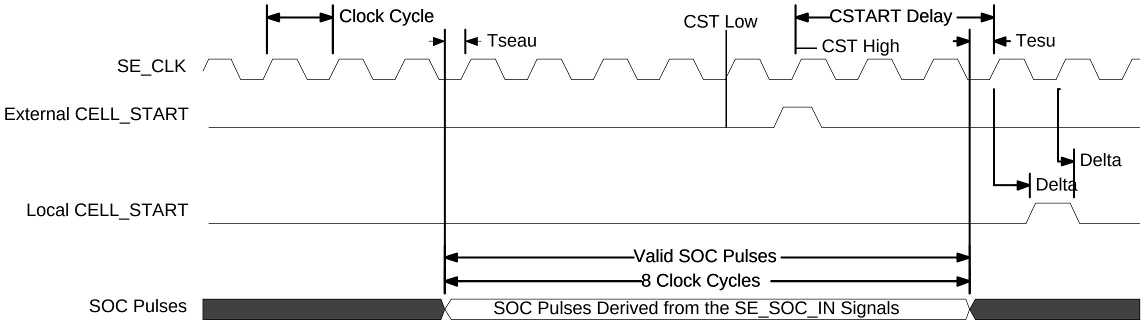


Figure 30. QSE Cell-Level Timing

The QSE performs cut-through routing wherever possible and requires the SOC to be synchronized across all input ports. For greater flexibility, the QSE allows cells starting within a window of eight clock pulses to be considered valid. The end of the 8-clock-cycle window is also indicated by the local CELL_START signal.

4.2.2 Relation Between Local CELL_START and Data Out of the QSE

The QSE switch latency from the local CELL_START signal to the first nibble depends on the gang mode, as shown in Figure 31. The switch latency is 8 clocks from the local CELL_START signal for all unicast gang modes, except for unicast gang mode = 0, in which case the switch latency is 11 clocks..

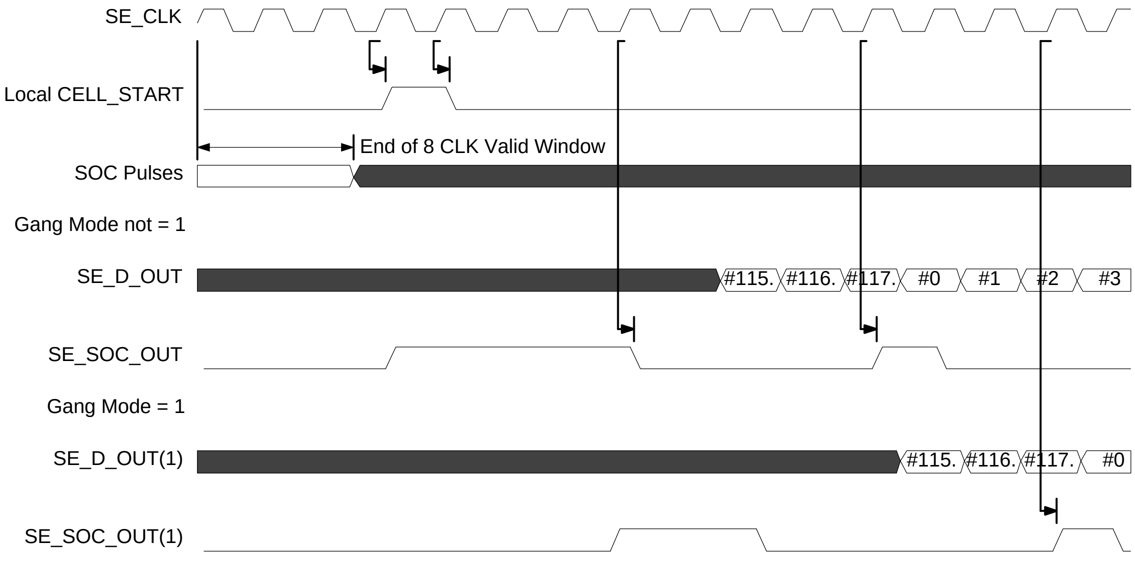


Figure 31. QSE Switch Latency

The CELL_24_START signal is used as a strobe to synchronize the internal state machines of all QSEs and QRTs in the system. When it occurs, the CELL_24_START signal must be coincident with the CELL_START signal and should occur every 4Nth cell time. (The signal is called CELL_24_START for legacy reasons that are no longer relevant.)

4.3 General Description of Phase Aligners

The phase aligners recover a clock from the data in the QSE-to-QSE, QRT-to-QSE, and QSE-to-QRT interfaces as shown in Figure 32 on page 49. The forward cell path consists of five signals, SE_D(3:0) and SE_SOC, while the backward path consists of one signal, BP_ACK.

In the forward cell path, the phase aligners lock to the SE_SOC_IN signal that has guaranteed signal transitions. The recovered clock is then used to sample the other signals, SE_D_IN(3:0).

In the backward path, the phase aligners lock to the BP_ACK_IN signal that has guaranteed signal transitions.

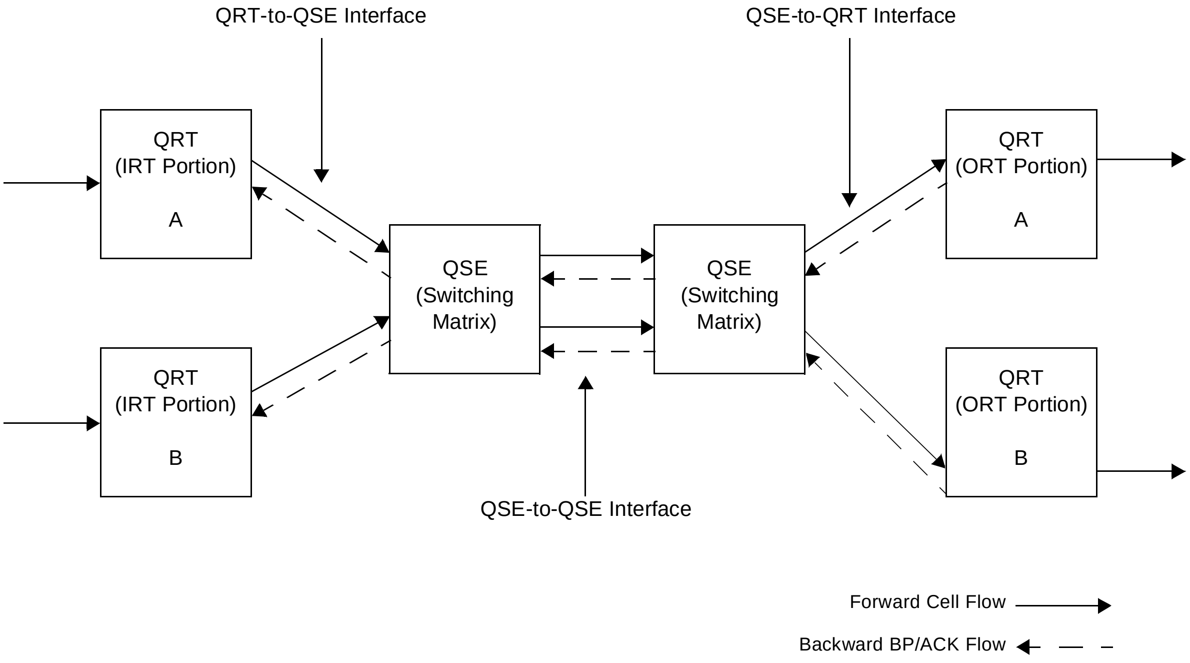


Figure 32. Basic Forward and Backward Data Path

4.4 Multicast Backpressure Control

As described in section 2.4.1 “Multicast Queue Engine” on page 28, the multicast queue engine computes multipriority backpressure (high, medium, or low) based on the following factors:

- Total buffer usage.
- Buffer usage on an individual port.

The buffer use constraints described therein guarantee against one port flooding the QSE and choking other ports (by the per-port buffer limits) or heavy traffic from cells of a lower priority level choking cells of higher priorities (by allowing buffers to be reserved for high- and medium-priority cells).

The QSE is tolerant of the QRT and other QSEs on its input ignoring the backpressure it applies. Depending on the situation, cells that arrive in violation of recommended backpressure may be dropped or may be accepted and treated as normal cells. This is fault behavior since, during normal operation, neither the QSE nor the QRT will ever violate backpressure applied by a downstream QSE.

4.5 Multilevel Reset

When the RESET pin is asserted, the QSE is in total reset. Access is not permitted to any register; and all QSE-driven signals, except for RAM_CLK, are static at either 0 or 1.

When the CHIP_HARDWARE_RESET bit in the CHIP_MODE register (refer to section 9.3.2 “CHIP_MODE” on page 95) is enabled, all registers can be read from and written to, but do not attempt to access the multicast port vectors in the multicast RAM. The rest of the device is in full reset.

When the CHIP_HARDWARE_RESET bit in the CHIP_MODE register (refer to section 9.3.2 “CHIP_MODE” on page 95) is disabled, but the SW_RESET bit in the CONTROL_REGISTER (refer to section 9.3.22 “CONTROL_REGISTER” on page 103) is enabled, the processor has fast access to the multicast RAM. This mode allows the multicast port vectors to be set up quickly at initialization. In normal device operation, the microprocessor has a single multicast RAM access every 118 clocks.

5. FAULT SPECIFICATION

5. 1. Purpose

The purpose of this chapter is to provide system designers with the high-level failure behavior of the system. It documents the algorithms used, as well as the QRT- and QSE-specific behaviors required.

5. 2. Basic Data and BP/ACK Flow

The basic data path through the QRT and QSE is shown in Figure 33. In this example, data enters the switch through a UTOPIA interface at the IRT portion on the QRT and is queued in the IRT. Then, cells are played out to the switch fabric (which consists of one or more stages of QSEs), and finally enters the ORT portion of the QRT where it is queued. Cells are then played out of the switch through a UTOPIA interface. Failures within the switch fabric are looked for, excluding the UTOPIA interfaces.

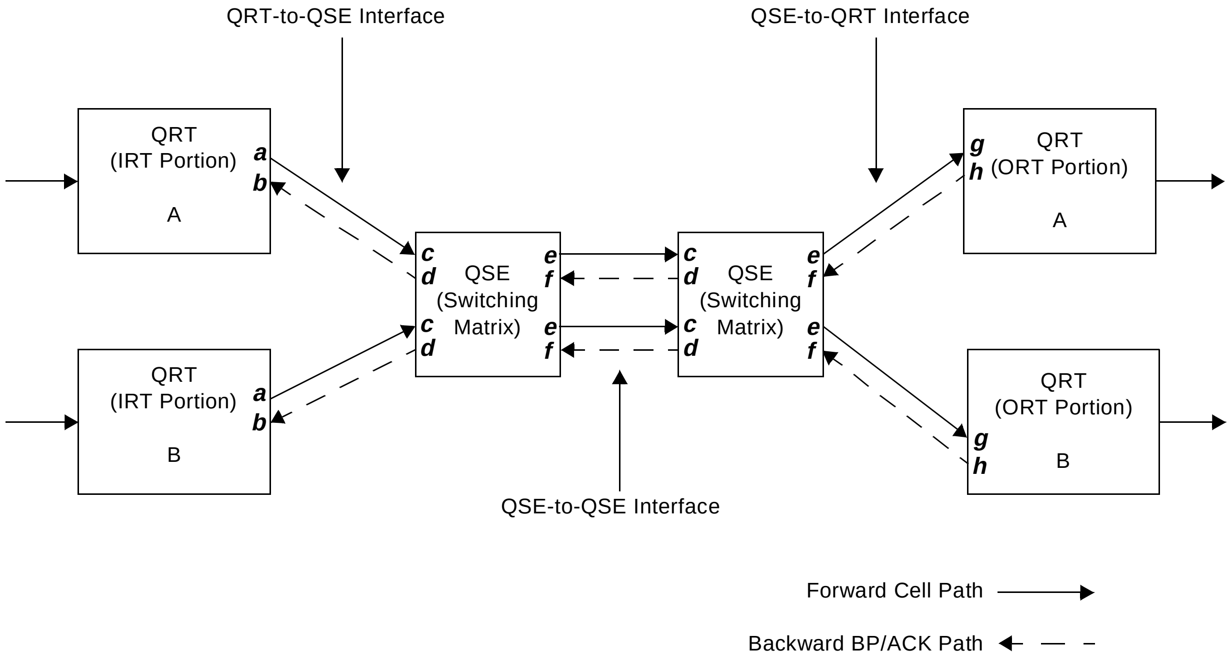


Figure 33. Basic Data Path (SE_D_OUT/IN and SE_SOC_OUT/IN in Forward Path, BP_ACK_OUT/IN in Backward Path)

It is important to decide at the beginning what level of fault diagnosis, recovery, and additional functionality is desired. The goal is to be robust to:

- Any stuck-at fault,
- Any bridging fault within a port, and
- Possible card removal.

In particular, the system should not be totally disabled by any of the above, although it may operate at a reduced performance. In addition, any of the previous failures should be locatable. The system will not necessarily be robust to:

- All dribbling errors,
- Any bridging fault between ports, and
- Complex partial failures.

As much as possible, the following secondary goals will be taken into account in the algorithms implemented.

- Quick and responsive in failure detection,
- Localize the problem, and minimize the effect of the problem,
- Avoid throughput collapse,
- Identify and locate the problem,
- Possibly do strong manufacturing test,
- On line diagnostics, and
- Automatically detect when a failure resolves itself.

5. 3. Fault Detection Mechanisms

Several mechanisms are built into the QSE and the QRT to facilitate online detection and location of faults within a system. These involve:

- Special coding and guaranteed transitions on the BP_ACK line. If this is not detected, the condition is flagged, and no data is sent out on the port.
- Special coding and guaranteed transitions on the SE_SOC line. If this is not detected, the port is flagged as failed, and all data from the port is discarded.
- Cell present being marked by two bits, Nibble 0 is 10xx for cell present or 01xx for cell absent (11xx and 00xx are considered errors, the port is flagged as failed, and all data from the port is discarded).
- Idle cell is coded by five nibbles, (01xx, 0000,1000, 0100, 0010, 0001). This pattern verifies no line has a stuck-at or bridging fault.
- Closed loop port behavior ensures no data is sent to a bad port. If a port is flagged as failed, then no BP signal is sent back on the BP_ACK line. This in turn will be detected by the transmitting QSE, and will be flagged. In addition, no data will be sent to that port while the condition exists.
- Nibbles 1 through 12 of the cell header are parity protected. For unicast data, in the QRT, a parity errored cell is dropped, but an ACK is still issued. In the QSE, an ONACK is issued for parity errored cells. This results in the unicast ONACKed cell being retransmitted if the parity error did not occur in the last stage. For multicast data, parity errored cells are dropped by both the QRT and QSE.
- Marked cell count. All input and output ports have a 4-bit cell counter. Any cell that goes by with a marked cell count bit set increments this count. (Note that unicast traffic has to be ACKed to increment the count.) Modulo 16 arithmetic can be performed on these counts to determine if there was any unexpected cell loss or generation.
- Whenever a port is tagged dead due to BP_ACK failure, there needs to be two consecutive good instances to make the port alive again.

5. 4. Interface Behavior

In Figure 33 on page 51, the various interfaces of interest are labeled ***a***, ***b***, ***c***, ***d***, ***e***, ***f***, ***g***, and ***h*** respectively.

5. 5. IRT-to-Switch Fabric Interface

An IRT interface consists of ***a*** and ***b*** in Figure 33 on page 51. Where ***a*** refers to each of the four SE_SOC_OUT and SE_D_OUT(3:0) data ports, and ***b*** refers to the corresponding BP_ACK_IN signals in the QRT.

The failure conditions detected by the IRT on ***b***, and the actions taken are summarized in Table 9.

Table 9. Failure Conditions, IRT-to Switch Fabric Interface

Fault Detected on *b*	Action Taken	Comment
Cannot lock to special coding and guaranteed transitions on BP_ACK_IN.	Idle cells sent out on data interface ***a***. Internally to the IRT, cells that would have gone out are MNACKed, and no multicast cells are generated for the port. BP_ACK_FAIL signaled to the microprocessor.	Port treated as dead. Problem is most likely with the BP_ACK_IN line.
No BP received on BP_ACK_IN line.	Idle cells sent out on data interface ***a***. Internally to the IRT, cells that would have gone out are MNACKed, and no multicast cells are generated for the port. BP_REMOTE_FAIL signaled to the microprocessor.	Port treated as dead. Problem is with the forward data flow, and the QSE is signaling this back to the IRT.
No ACK, MNACK, or ONACK received, although unicast cell sent out.	Cell transmitted treated as sent. ACK_LIVE_FAIL signaled to the microprocessor.	

5. 6. QSE Interface, Receive Data Direction

A QSE Receive interface consists of ***c*** and ***d*** in Figure 33 on page 51. Where ***c*** refers to each of the four SE_SOC_IN and SE_D_IN(3:0) data ports, and ***d*** refers to the corresponding BP_ACK_OUT signals in the QSE.

The failure conditions detected by the ORT on ***c***, and the actions taken are summarized in Table 10.

Table 10. Failure Conditions, QSE Receive Interface

Fault Detected on *c*	Action Taken	Comment
Cannot lock to special coding and guaranteed transitions on SE_SOC_IN.	No BP sent out on ***d***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Withholding BP on ***d*** signals to the previous stage that the port should not be used.
Invalid cell present coding on SE_D_IN(3:0).	No BP sent out on ***d***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Most likely due to unconnected input lines that are pulled up or down. Withholding BP on ***d*** signals to the previous stage that the port should not be used.
Bad idle cell coding on SE_D_IN(3:0).	No BP sent out on ***d***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Withholding BP on ***d*** signals to the previous stage that the port should not be used.
Parity fail.	ONACK sent out on ***d*** for unicast data**.** Multicast data dropped. PARITY_FAIL signaled to the microprocessor.	QSE does not necessarily have time to drop cell by the time it has detected a parity error.

5. 7. QSE Interface, Transmit Data Direction

A QSE Transmit interface consists of ***e*** and ***f*** in Figure 33 on page 51. Where ***e*** refers to each of the 32 SE_SOC_OUT and SE_D_OUT(3:0) data ports, and ***f*** refers to the corresponding BP_ACK_IN signals in the QSE.

The failure conditions detected by the QSE on ***f***, and the actions taken are summarized in Table 11.

Table 11. Failure Conditions, QSE Transmit Interface

Fault Detected on *f*	Action Taken	Comment
Cannot lock to special coding and guaranteed transitions on BP_ACK_IN.	Idle cells sent out on data interface ***e***. Data routed around port if possible. Multicast data is dropped if all possible port choices are dead or off. Unicast data is optionally dropped if all possible port choices are dead or off. BP_ACK_FAIL signaled to the microprocessor.	Port treated as dead. Problem is most likely with the BP_ACK line.
No BP received on BP_ACK_IN line.	Idle cells sent out on data interface ***e***. Data routed around port if possible. Multicast data is dropped if all possible port choices are dead or off. Unicast data is optionally dropped if all possible port choices are dead or off. BP_REMOTE_FAIL signaled to the microprocessor.	Port treated as dead. Problem is with the forward data flow.
No ACK, MNACK, or ONACK received on BP_ACK_IN line.	No action taken.	This contingency is not monitored in the QSE.

5. 8. Switch Fabric-to-ORT Interface

An ORT interface consists of ***g*** and ***h*** in Figure 33 on page 51. Where ***g*** refers to each of the four SE_SOC_IN and SE_D_IN(3:0) data ports, and ***h*** refers to the corresponding BP_ACK_OUT signals in the QRT.

The failure conditions detected by the ORT on ***g***, and the actions taken are summarized in Table 12.

Table 12. Failure Conditions, Switch Fabric-to-ORT Interface

Fault Detected on *g*	Action Taken	Comment
Cannot lock to special coding and guaranteed transitions on SE_SOC_IN.	No BP sent out on ***h***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Withholding BP on ***h*** signals to the previous stage that the port should not be used.
Invalid cell present coding on SE_D_IN(3:0).	No BP sent out on ***h***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Most likely due to unconnected input lines that are pulled up or down. Withholding BP on ***h*** signals to the previous stage that the port should not be used.
Bad idle cell coding on SE_D_IN(3:0).	No BP sent out on ***h***. All data discarded. SE_INPUT_PORT_FAIL signaled to the microprocessor.	Withholding BP on ***h*** signals to the previous stage that the port should not be used.
Parity fail.	ACK sent out on ***h***. Parity errored cell dropped. TX_PARITY_FAIL signaled to the microprocessor.	ACK already sent by the time the QRT has detected a parity error. Note that in this case we have ACKed a cell that was dropped.

5. 9. Types of Failures and Their Manifestation

Possible faults, the effects and how they affect the network are shown in Table 13.

Table 13. Faults

Fault	Manifestation	Effect on Network
Wire Connection		
Data line from SE_D(3:0) stuck at 0 or 1.	Invalid idle cell, with some 10/01 fail and parity error.	Port shut down on receipt of first bad idle cell until condition is fixed, as port failure is sent back to source of data by the lack of BP indication.
SE_SOC line stuck at 0 or 1.	Loss of lock on special coding on SE_SOC_IN.	Port shut down until condition is fixed, as port failure is sent back to source of data by the lack of BP indication.
BP_ACK line stuck at 0 or 1.	Loss of lock on special coding on BP_ACK_IN.	Port shut down until the condition is fixed.
Bridging fault within a port.	Invalid idle cell, with some 10/01 fail and parity error.	Port shut down on receipt of first bad idle cell until condition is fixed, as port failure is sent back to source of data by the lack of BP indication.
QRT and QSE Port Failures		
No SE_SOC_OUT generation.	Loss of lock on special coding on SE_SOC_IN.	Port shut down until condition is fixed, as port failure is sent back to source of data by the lack of BP indication.

Table 13. Faults

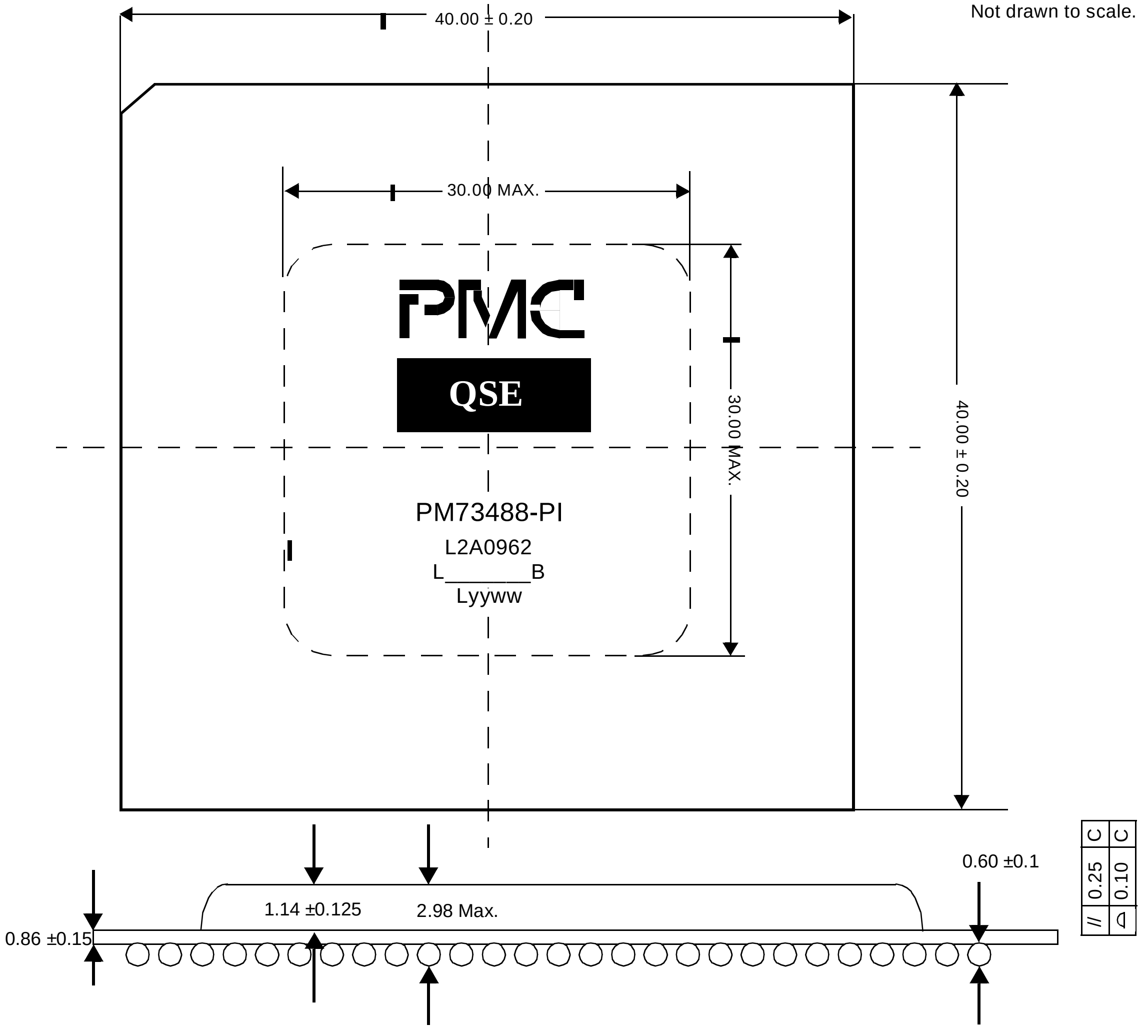
Fault	Manifestation	Effect on Network
No/Invalid data generated.	10/01 Fail, or parity error, invalid idle cell.	Port shut down on receipt of first bad idle cell until condition is fixed, as port failure is sent back to source of data by the lack of BP indication.
No BP_ACK_OUT generation.	Loss of lock on special coding on BP_ACK_IN.	Port shut down until the condition is fixed.
QSE Chip Failures		
Multicast handling.	Cell loss or generation.	Detection possible using marked cell count.
MC Cell pool buffer.	Parity error in header or cell.	Only detection in header, not in payload.
Partial cell buffers.	Parity error in header and cell.	Parity error.
Multicast and Unicast selection networks.	Cell gets out on wrong port, cell duplicated, cell lost.	Cell to wrong port may be noticed by receiving QRT, if that VC is not active. cell duplication and cell loss detection possible using marked cell count.
Arbiter.	Cell lost.	Detection possible using marked cell count.

6 SIGNAL DESCRIPTIONS

6.1 Package Diagram

A 596-pin Enhanced Plastic Ball Grid Array (EPBGA), shown in Figure 34 (part 1 and part 2), is used for the QSE.The package measurements are shown in millimeters.

Measurements are
shown in millimeters.
Not drawn to scale.

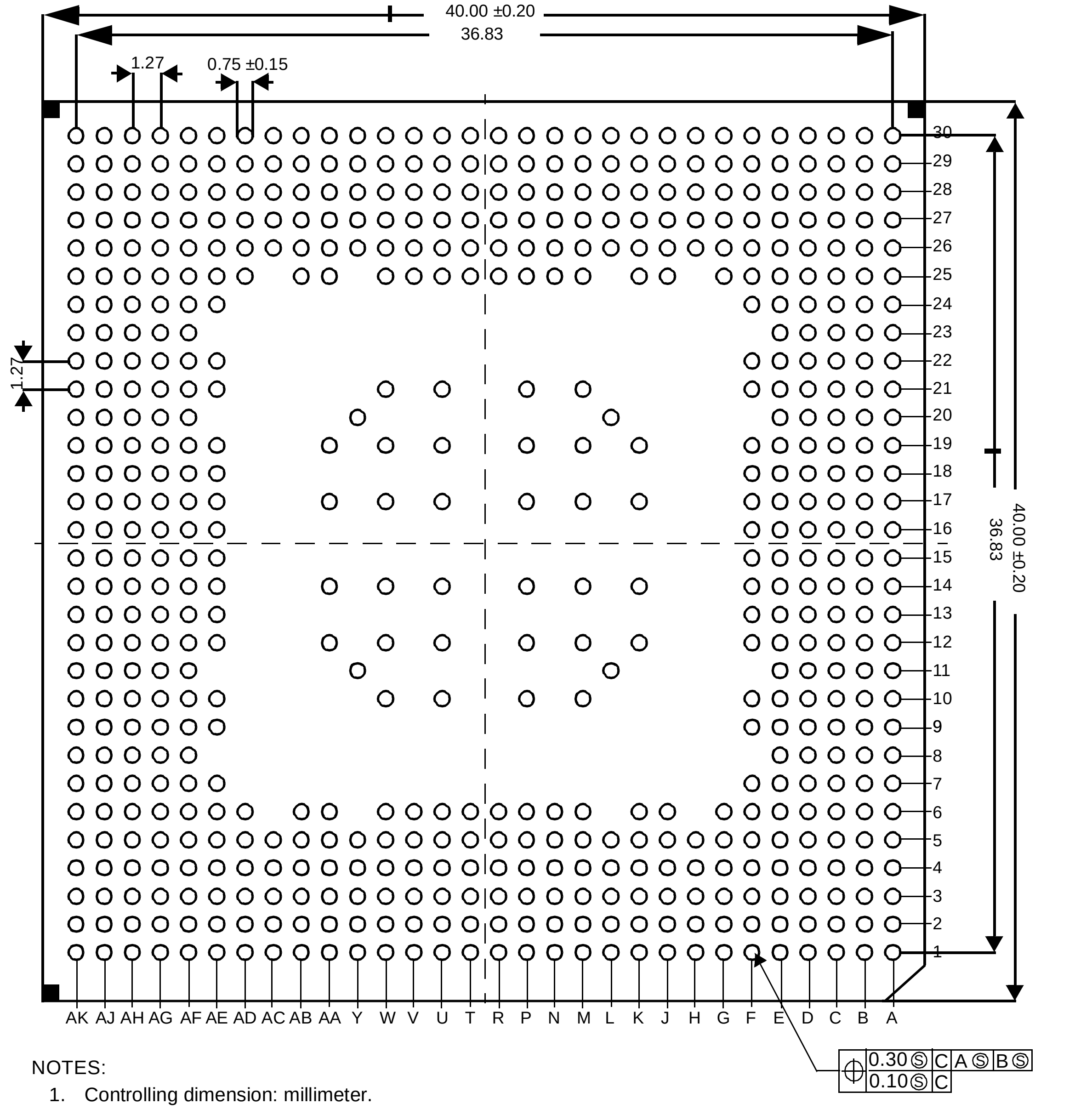


NOTES:

1. “L2A0962” is the LSI part number.
2. “L_____B” is the wafer batch code.
3. “Lyyww” is the assembly date code.
4. Dimensions are for reference.
5. Controlling dimension: millimeter.
6. // = Parallelism tolerance.

Figure 34. 596-Ball Enhanced Plastic BGA Physical Dimensions Diagram (Top view)

Measurements are shown in millimeters. Not drawn to scale.



NOTES:

1. Controlling dimension: millimeter.
2. PCB material: high temperature glass/epoxy resin cloth (that is, driclad, MCL-679, or equivalent). Solder resist: photoimagable (that is, vacrel 8130, DSR3241, PSR4000, or equivalent).
3. If you need a measurement not shown in this figure, contact PMC.

Figure 35. 596-Ball Enhanced Plastic BGA Physical Dimensions Diagram (Bottom view)

6.2 Signal Locations (Signal Name to Ball)

Table 14. Signal Locations (Signal Name to Ball)

Signal Name	Ball	Signal Name	Ball	Signal Name	Ball	Signal Name	Ball
BP_ACK_IN(0)	L29	SE_D_IN11(1)	B7	SE_D_OUT16(2)	AE24	GND	R3
BP_ACK_IN(1)	K30	SE_D_IN11(2)	C9	SE_D_OUT16(3)	AG26	GND	T3
BP_ACK_IN(2)	G30	SE_D_IN11(3)	D10	SE_D_OUT17(0)	AJ28	GND	V1
BP_ACK_IN(3)	N27	SE_D_IN12(0)	A4	SE_D_OUT17(1)	AE22	GND	AA2
BP_ACK_IN(4)	J29	SE_D_IN12(1)	C8	SE_D_OUT17(2)	AH27	GND	W4
BP_ACK_IN(5)	M28	SE_D_IN12(2)	D9	SE_D_OUT17(3)	AF23	GND	AF1
BP_ACK_IN(6)	H30	SE_D_IN12(3)	A3	SE_D_OUT18(0)	AK28	GND	AD3
BP_ACK_IN(7)	L27	SE_D_IN13(0)	B5	SE_D_OUT18(1)	AH25	GND	AJ2
BP_ACK_IN(8)	M26	SE_D_IN13(1)	F10	SE_D_OUT18(2)	AJ26	GND	AG4
BP_ACK_IN(9)	H29	SE_D_IN13(2)	B4	SE_D_OUT18(3)	AE21	GND	AE6
BP_ACK_IN(10)	K28	SE_D_IN13(3)	F9	SE_D_OUT19(0)	AF21	GND	AF3
BP_ACK_IN(11)	F30	SE_D_IN14(0)	E8	SE_D_OUT19(1)	AF22	GND	AH5
BP_ACK_IN(12)	N25	SE_D_IN14(1)	B3	SE_D_OUT19(2)	AH23	GND	AK5
BP_ACK_IN(13)	M25	SE_D_IN14(2)	D7	SE_D_OUT19(3)	AJ27	GND	AH7
BP_ACK_IN(14)	J28	SE_D_IN14(3)	F7	SE_D_OUT20(0)	AF20	GND	AJ10
BP_ACK_IN(15)	L26	SE_D_IN15(0)	D6	SE_D_OUT20(1)	AK27	GND	AG12
BP_ACK_IN(16)	D30	SE_D_IN15(1)	E7	SE_D_OUT20(2)	AJ24	GND	AK13
BP_ACK_IN(17)	G29	SE_D_IN15(2)	E6	SE_D_OUT20(3)	AG22	GND	AH15
BP_ACK_IN(18)	J27	SE_D_IN15(3)	D5	SE_D_OUT21(0)	AK25	GND	AH16
BP_ACK_IN(19)	K27	SE_D_IN16(0)	F3	SE_D_OUT21(1)	AE18	GND	AK18
BP_ACK_IN(20)	K26	SE_D_IN16(1)	C1	SE_D_OUT21(2)	AE19	GND	AJ21
BP_ACK_IN(21)	J26	SE_D_IN16(2)	D2	SE_D_OUT21(3)	AH22	GND	AG19
BP_ACK_IN(22)	H28	SE_D_IN16(3)	H3	SE_D_OUT22(0)	AG20	GND	AK26
BP_ACK_IN(23)	D29	SE_D_IN17(0)	K5	SE_D_OUT22(1)	AF19	GND	AH24
BP_ACK_IN(24)	C30	SE_D_IN17(1)	K4	SE_D_OUT22(2)	AJ23	GND	AH26
BP_ACK_IN(25)	F28	SE_D_IN17(2)	J4	SE_D_OUT22(3)	AH21	GND	AJ29
BP_ACK_IN(26)	E29	SE_D_IN17(3)	G2	SE_D_OUT23(0)	AG18	GND	AG27
BP_ACK_IN(27)	K25	SE_D_IN18(0)	L5	SE_D_OUT23(1)	AJ22	GND	AE25
BP_ACK_IN(28)	C29	SE_D_IN18(1)	J3	SE_D_OUT23(2)	AH19	GND	AF28
BP_ACK_IN(29)	J25	SE_D_IN18(2)	M6	SE_D_OUT23(3)	AK23	GND	AF30
BP_ACK_IN(30)	D28	SE_D_IN18(3)	N6	SE_D_OUT24(0)	AE17	GND	AD28
BP_ACK_IN(31)	H26	SE_D_IN19(0)	K3	SE_D_OUT24(1)	AH18	GND	AA29
BP_ACK_OUT(0)	AB5	SE_D_IN19(1)	H2	SE_D_OUT24(2)	AJ20	GND	W27
BP_ACK_OUT(1)	AF2	SE_D_IN19(2)	M5	SE_D_OUT24(3)	AK21	GND	V30
BP_ACK_OUT(2)	AA6	SE_D_IN19(3)	L4	SE_D_OUT25(0)	AK20	GND	T28

Table 14. Signal Locations (Signal Name to Ball) (Continued)

Signal Name	Ball	Signal Name	Ball	Signal Name	Ball	Signal Name	Ball
BP_ACK_OUT(3)	AG2	SE_D_IN20(0)	M3	SE_D_OUT25(1)	AJ19	GND	R28
BP_ACK_OUT(4)	AB6	SE_D_IN20(1)	J2	SE_D_OUT25(2)	AE16	GND	N30
BP_ACK_OUT(5)	AE3	SE_D_IN20(2)	N4	SE_D_OUT25(3)	AF18	GND	K29
BP_ACK_OUT(6)	AC5	SE_D_IN20(3)	G1	SE_D_OUT26(0)	AG17	GND	M27
BP_ACK_OUT(7)	AH2	SE_D_IN21(0)	L2	SE_D_OUT26(1)	AF16	GND	E30
BP_ACK_OUT(8)	AD4	SE_D_IN21(1)	N3	SE_D_OUT26(2)	AJ18	GND	G28
BP_ACK_OUT(9)	AD6	SE_D_IN21(2)	P6	SE_D_OUT26(3)	AF17	GND	E28
BP_ACK_OUT(10)	AG3	SE_D_IN21(3)	N5	SE_D_OUT27(0)	AK19	GND	B29
BP_ACK_OUT(11)	AE4	SE_D_IN22(0)	M2	SE_D_OUT27(1)	AK17	GND	D27
BP_ACK_OUT(12)	AD5	SE_D_IN22(1)	L1	SE_D_OUT27(2)	AH17	GND	F25
BP_ACK_OUT(13)	AE5	SE_D_IN22(2)	P5	SE_D_OUT27(3)	AG16	GND	C26
BP_ACK_OUT(14)	AF4	SE_D_IN22(3)	N2	SE_D_OUT28(0)	AG15	GND	A26
BP_ACK_OUT(15)	AJ1	SE_D_IN23(0)	P4	SE_D_OUT28(1))	AK15	GND	C24
BP_ACK_OUT(16)	AK2	SE_D_IN23(1)	R4	SE_D_OUT28(2)	AK14	GND	B21
BP_ACK_OUT(17)	AG5	SE_D_IN23(2)	P3	SE_D_OUT28(3)	AK16	GND	D19
BP_ACK_OUT(18)	AF6	SE_D_IN23(3)	P1	SE_D_OUT29(0)	AH14	GND	A18
BP_ACK_OUT(19)	AF7	SE_D_IN24(0)	R2	SE_D_OUT29(1)	AJ13	GND	C16
BP_ACK_OUT(20)	AG6	SE_D_IN24(1)	T2	SE_D_OUT29(2)	AF15	GND	C15
BP_ACK_OUT(21)	AH4	SE_D_IN24(2)	R1	SE_D_OUT29(3)	AK12	GND	A13
BP_ACK_OUT(22)	AE7	SE_D_IN24(3)	U1	SE_D_OUT30(0)	AF14	GND	B10
BP_ACK_OUT(23)	AG7	SE_D_IN25(0)	T4	SE_D_OUT30(1)	AE15	GND	D12
BP_ACK_OUT(24)	AJ3	SE_D_IN25(1)	W1	SE_D_OUT30(2)	AG14	GND	A5
BP_ACK_OUT(25)	AF8	SE_D_IN25(2)	T5	SE_D_OUT30(3)	AK11	GND	C7
BP_ACK_OUT(26)	AH6	SE_D_IN25(3)	V2	SE_D_OUT31(0)	AK10	GND	C5
BP_ACK_OUT(27)	AE9	SE_D_IN26(0)	Y1	SE_D_OUT31(1)	AJ12	GND	Y20
BP_ACK_OUT(28)	AJ4	SE_D_IN26(1)	U4	SE_D_OUT31(2)	AH13	GND	W19
BP_ACK_OUT(29)	AE10	SE_D_IN26(2)	T6	SE_D_OUT31(3)	AE14	GND	U19
BP_ACK_OUT(30)	AJ5	SE_D_IN26(3)	U5	/OE	D3	GND	P19
BP_ACK_OUT(31)	AF9	SE_D_IN27(0)	V3	RESET	AF13	GND	M19
CELL_24_START	C2	SE_D_IN27(1)	W2	SE_SOC_IN(0)	B22	GND	L20
CELL_START	J6	SE_D_IN27(2)	AA1	SE_SOC_IN(1)	C18	GND	W17
RAM_ADD(17)	AH1	SE_D_IN27(3)	V5	SE_SOC_IN(2)	A20	GND	U17
RAM_ADD(18)	AF12	SE_D_IN28(0)	Y2	SE_SOC_IN(3)	D16	GND	P17
/IDDTN	G5	SE_D_IN28(1)	W3	SE_SOC_IN(4)	B15	GND	M17
STAT_OUT	K6	SE_D_IN28(2)	AC1	SE_SOC_IN(5)	A12	GND	W14
CTRL_IN	E2	SE_D_IN28(3)	AD1	SE_SOC_IN(6)	D14	GND	U14

Table 14. Signal Locations (Signal Name to Ball) (Continued)

Signal Name	Ball	Signal Name	Ball	Signal Name	Ball	Signal Name	Ball
/ACK	AJ11	SE_D_IN29(0)	AB2	SE_SOC_IN(7)	B12	GND	P14
ADD(0)	AK3	SE_D_IN29(1)	AA3	SE_SOC_IN(8)	C12	GND	M14
ADD(1)	AG9	SE_D_IN29(2)	Y4	SE_SOC_IN(9)	C10	GND	Y11
ADD(2)	AH8	SE_D_IN29(3)	V6	SE_SOC_IN(10)	E11	GND	W12
ADD(3)	AK4	SE_D_IN30(0)	AC2	SE_SOC_IN(11)	E10	GND	U12
ADD(4)	AF10	SE_D_IN30(1)	Y5	SE_SOC_IN(12)	E9	GND	P12
ADD(5)	AG10	SE_D_IN30(2)	AE1	SE_SOC_IN(13)	C6	GND	M12
ADD(6)	AH9	SE_D_IN30(3)	AD2	SE_SOC_IN(14)	C4	GND	L11
ADD(7)	AJ7	SE_D_IN31(0)	AA4	SE_SOC_IN(15)	A2	GND	W21
/CS	AK7	SE_D_IN31(1)	AA5	SE_SOC_IN(16)	J5	GND	AA12
DATA(0)	AK6	SE_D_IN31(2)	AG1	SE_SOC_IN(17)	D1	GND	M10
DATA(1)	AF11	SE_D_IN31(3)	AC3	SE_SOC_IN(18)	F1	GND	K19
DATA(2)	AJ8	SE_D_OUT00(0)	P26	SE_SOC_IN(19)	H1	GND	P10
DATA(3)	AE12	SE_D_OUT00(1)	L30	SE_SOC_IN(20)	K1	GND	U10
DATA(4)	AE13	SE_D_OUT00(2)	M29	SE_SOC_IN(21)	R6	GND	W10
DATA(5)	AG11	SE_D_OUT00(3)	R25	SE_SOC_IN(22)	R5	GND	K12
DATA(6)	AH10	SE_D_OUT01(0)	R27	SE_SOC_IN(23)	M1	GND	K14
DATA(7)	AJ9	SE_D_OUT01(1)	P27	SE_SOC_IN(24)	T1	GND	AA14
/INTR	AG13	SE_D_OUT01(2)	R26	SE_SOC_IN(25)	U3	GND	K17
/RD	AK8	SE_D_OUT01(3)	N29	SE_SOC_IN(26)	U6	GND	AA17
/WR	AH12	SE_D_OUT02(0)	R29	SE_SOC_IN(27)	V4	GND	AA19
/PLL_BYPASS	P25	SE_D_OUT02(1)	M30	SE_SOC_IN(28)	W5	GND	M21
PLL_VDD	AJ30	SE_D_OUT02(2)	P30	SE_SOC_IN(29)	W6	GND	P21
PLL_VSS	AF27	SE_D_OUT02(3)	P28	SE_SOC_IN(30)	AB3	GND	U21
not used	H5	SE_D_OUT03(0)	T30	SE_SOC_IN(31)	AB4	V_{DD}	A1
/SCAN_EN	G6	SE_D_OUT03(1)	U30	SE_SOC_OUT0	N26	V_{DD}	C3
/SCAN_TRST	E4	SE_D_OUT03(2)	R30	SE_SOC_OUT1	T27	V_{DD}	E5
SCAN_TCK	G4	SE_D_OUT03(3)	T29	SE_SOC_OUT2	AC30	V_{DD}	F2
SCAN_TDI	B1	SE_D_OUT04(0)	U28	SE_SOC_OUT3	AB27	V_{DD}	H4
SCAN_TDO	F5	SE_D_OUT04(1)	V29	SE_SOC_OUT4	AG25	V_{DD}	J1
SCAN_TMS	F4	SE_D_OUT04(2)	T26	SE_SOC_OUT5	AG21	V_{DD}	L3
SE_CLK_BYPASS	AF25	SE_D_OUT04(3)	W30	SE_SOC_OUT6	AK24	V_{DD}	P2
SE_CLK	AJ16	SE_D_OUT05(0)	U26	SE_SOC_OUT7	AJ15	V_{DD}	U2
SE_D_IN00(0)	E19	SE_D_OUT05(1)	T25	RAM_ADD(0)	G26	V_{DD}	AB1
SE_D_IN00(1)	D20	SE_D_OUT05(2)	U27	RAM_ADD(1)	G27	V_{DD}	Y3
SE_D_IN00(2)	A23	SE_D_OUT05(3)	Y30	RAM_ADD(2)	G25	V_{DD}	AE2

Table 14. Signal Locations (Signal Name to Ball) (Continued)

Signal Name	Ball	Signal Name	Ball	Signal Name	Ball	Signal Name	Ball
SE_D_IN00(3)	C19	SE_D_OUT06(0)	AA30	RAM_ADD(3)	E27	V_{DD}	AC4
SE_D_IN01(0)	D18	SE_D_OUT06(1)	W29	RAM_ADD(4)	F27	V_{DD}	AK1
SE_D_IN01(1)	A24	SE_D_OUT06(2)	V28	RAM_ADD(5)	F26	V_{DD}	AH3
SE_D_IN01(2)	A21	SE_D_OUT06(3)	U25	RAM_ADD(6)	B30	V_{DD}	AF5
SE_D_IN01(3)	B20	SE_D_OUT07(0)	W28	RAM_ADD(7)	A29	V_{DD}	AJ6
SE_D_IN02(0)	F17	SE_D_OUT07(1)	Y29	RAM_ADD(8)	E25	V_{DD}	AG8
SE_D_IN02(1)	E18	SE_D_OUT07(2)	V27	RAM_ADD(9)	D25	V_{DD}	AK9
SE_D_IN02(2)	F16	SE_D_OUT07(3)	V26	RAM_ADD(10)	D26	V_{DD}	AH11
SE_D_IN02(3)	B19	SE_D_OUT08(0)	AA28	RAM_ADD(11)	F24	V_{DD}	AJ14
SE_D_IN03(0)	E17	SE_D_OUT08(1)	AB29	RAM_ADD(12)	D24	V_{DD}	AJ17
SE_D_IN03(1)	B18	SE_D_OUT08(2)	W26	RAM_ADD(13)	E24	V_{DD}	AK22
SE_D_IN03(2)	E16	SE_D_OUT08(3)	AD30	RAM_ADD(14)	E23	V_{DD}	AH20
SE_D_IN03(3)	D17	SE_D_OUT09(0)	AC29	RAM_ADD(15)	C27	V_{DD}	AJ25
SE_D_IN04(0)	C17	SE_D_OUT09(1)	W25	RAM_CLK	B23	V_{DD}	AG23
SE_D_IN04(1)	A17	SE_D_OUT09(2)	V25	RAM_DATA(0)	F22	V_{DD}	AK30
SE_D_IN04(2)	A19	SE_D_OUT09(3)	Y27	RAM_DATA(1)	B28	V_{DD}	AH28
SE_D_IN04(3)	B16	SE_D_OUT10(0)	AB28	RAM_DATA(2)	F21	V_{DD}	AF26
SE_D_IN05(0)	A16	SE_D_OUT10(1)	AD29	RAM_DATA(3)	B26	V_{DD}	AE29
SE_D_IN05(1)	A14	SE_D_OUT10(2)	AE30	RAM_DATA(4)	C25	V_{DD}	AC27
SE_D_IN05(2)	A15	SE_D_OUT10(3)	Y26	RAM_DATA(5)	A28	V_{DD}	AB30
SE_D_IN05(3)	D15	SE_D_OUT11(0)	AC28	RAM_DATA(6)	B27	V_{DD}	Y28
SE_D_IN06(0)	E15	SE_D_OUT11(1)	AG30	RAM_DATA(7)	C23	V_{DD}	U29
SE_D_IN06(1)	B13	SE_D_OUT11(2)	AA26	RAM_DATA(8)	E22	V_{DD}	P29
SE_D_IN06(2)	C14	SE_D_OUT11(3)	AA27	RAM_DATA(9)	E21	V_{DD}	J30
SE_D_IN06(3)	A11	SE_D_OUT12(0)	AA25	RAM_DATA(10)	D21	V_{DD}	L28
SE_D_IN07(0)	F15	SE_D_OUT12(1)	AF29	RAM_DATA(11)	D22	V_{DD}	F29
SE_D_IN07(1)	E14	SE_D_OUT12(2)	AB26	RAM_DATA(12)	B24	V_{DD}	H27
SE_D_IN07(2)	F14	SE_D_OUT12(3)	AH30	RAM_DATA(13)	A27	V_{DD}	A30
SE_D_IN07(3)	C13	SE_D_OUT13(0)	AC26	RAM_DATA(14)	E20	V_{DD}	C28
SE_D_IN08(0)	A10	SE_D_OUT13(1)	AE28	RAM_DATA(15)	C22	V_{DD}	E26
SE_D_IN08(1)	E13	SE_D_OUT13(2)	AB25	/RAM_OE	C21	V_{DD}	B25
SE_D_IN08(2)	D13	SE_D_OUT13(3)	AG29	/RAM_WR	A25	V_{DD}	D23
SE_D_IN08(3)	B11	SE_D_OUT14(0)	AG28	/TEST_MODE	N28	V_{DD}	A22
SE_D_IN09(0)	A8	SE_D_OUT14(1)	AD25	GND	B2	V_{DD}	C20
SE_D_IN09(1)	A7	SE_D_OUT14(2)	AD27	GND	D4	V_{DD}	B17
SE_D_IN09(2)	E12	SE_D_OUT14(3)	AH29	GND	F6	V_{DD}	B14

Table 14. Signal Locations (Signal Name to Ball) (Continued)

Signal Name	Ball	Signal Name	Ball	Signal Name	Ball	Signal Name	Ball
SE_D_IN09(3)	B9	SE_D_OUT15(0)	AK29	GND	E3	V_{DD}	A9
SE_D_IN10(0)	D11	SE_D_OUT15(1)	AE26	GND	E1	V_{DD}	C11
SE_D_IN10(1)	F13	SE_D_OUT15(2)	AD26	GND	G3	V_{DD}	B6
SE_D_IN10(2)	F12	SE_D_OUT15(3)	AE27	GND	K2	V_{DD}	D8
SE_D_IN10(3)	B8	SE_D_OUT16(0)	AF24	GND	M4	RAM_ADD(16)	F19
SE_D_IN11(0)	A6	SE_D_OUT16(1)	AG24	GND	N1	RAM_PARITY	F18

6.3 Signal Locations (Ball to Signal Name)

Table 15. Signal Locations (Ball to Signal Name)

Ball	Signal Name	Ball	Signal Name	Ball	Signal Name)	all	Signal Name
A1	V_{DD}	F1	SE_SOC_IN18	T3	GND	AF5	V_{DD}
A2	SE_SOC_IN15	F2	V_{DD}	T4	SE_D_IN25(0)	AF6	BP_ACK_OUT(18)
A3	SE_D_IN12(3)	F3	SE_D_IN16(0)	T5	SE_D_IN25(2)	AF7	BP_ACK_OUT(19)
A4	SE_D_IN12(0)	F4	SCAN_TMS	T6	SE_D_IN26(2)	AF8	BP_ACK_OUT(25)
A5	GND	F5	SCAN_TDO	T25	SE_D_OUT05(1)	AF9	BP_ACK_OUT(31)
A6	SE_D_IN11(0)	F6	GND	T26	SE_D_OUT04(2)	AF10	ADD(4)
A7	SE_D_IN09(1)	F7	SE_D_IN14(3)	T27	SE_SOC_OUT1	AF11	DATA(1)
A8	SE_D_IN09(0)	F9	SE_D_IN13(3)	T28	GND	AF12	RAM_ADD(18)
A9	V_{DD}	F10	SE_D_IN13(1)	T29	SE_D_OUT03(3)	AF13	RESET
A10	SE_D_IN08(0)	F12	SE_D_IN10(2)	T30	SE_D_OUT03(0)	AF14	SE_D_OUT30(0)
A11	SE_D_IN06(3)	F13	SE_D_IN10(1)	U1	SE_D_IN24(3)	AF15	SE_D_OUT29(2)
A12	SE_SOC_IN05	F14	SE_D_IN07(2)	U2	V_{DD}	AF16	SE_D_OUT26(1)
A13	GND	F15	SE_D_IN07(0)	U3	SE_SOC_IN25	AF17	SE_D_OUT26(3)
A14	SE_D_IN05(1)	F16	SE_D_IN02(2)	U4	SE_D_IN26(1)	AF18	SE_D_OUT25(3)
A15	SE_D_IN05(2)	F17	SE_D_IN02(0)	U5	SE_D_IN26(3)	AF19	SE_D_OUT22(1)
A16	SE_D_IN05(0)	F18	RAM_PARITY	U6	SE_SOC_IN26	AF20	SE_D_OUT20(0)
A17	SE_D_IN04(1)	F19	RAM_ADD(16)	U10	GND	AF21	SE_D_OUT19(0)
A18	GND	F21	RAM_DATA(2)	U12	GND	AF22	SE_D_OUT19(1)
A19	SE_D_IN04(2)	F22	RAM_DATA(0)	U14	GND	AF23	SE_D_OUT17(3)
A20	SE_SOC_IN02	F24	RAM_ADD(11)	U17	GND	AF24	SE_D_OUT16(0)
A21	SE_D_IN01(2)	F25	GND	U19	GND	AF25	SE_CLK_BYPASS
A22	V_{DD}	F26	RAM_ADD(5)	U21	GND	AF26	V_{DD}
A23	SE_D_IN00(2)	F27	RAM_ADD(4)	U25	SE_D_OUT06(3)	AF27	PLL_VSS
A24	SE_D_IN01(1)	F28	BP_ACK_IN(25)	U26	SE_D_OUT05(0)	AF28	GND
A25	/RAM_WR	F29	V_{DD}	U27	SE_D_OUT05(2)	AF29	SE_D_OUT12(1)
A26	GND	F30	BP_ACK_IN(11)	U28	SE_D_OUT04(0)	AF30	GND
A27	RAM_DATA(13)	G1	SE_D_IN20(3)	U29	V_{DD}	AG1	SE_D_IN31(2)
A28	RAM_DATA(5)	G2	SE_D_IN17(3)	U30	SE_D_OUT03(1)	AG2	BP_ACK_OUT(3)
A29	RAM_ADD(7)	G3	GND	V1	GND	AG3	BP_ACK_OUT(10)
A30	V_{DD}	G4	SCAN_TCK	V2	SE_D_IN25(3)	AG4	GND
B1	SCAN_TDI	G5	/IDDTN	V3	SE_D_IN27(0)	AG5	BP_ACK_OUT(17)
B2	GND	G6	/SCAN_EN	V4	SE_SOC_IN27	AG6	BP_ACK_OUT(20)
B3	SE_D_IN14(1)	G25	RAM_ADD(2)	V5	SE_D_IN27(3)	AG7	BP_ACK_OUT(23)
B4	SE_D_IN13(2)	G26	RAM_ADD(0)	V6	SE_D_IN29(3)	AG8	V_{DD}
B5	SE_D_IN13(0)	G27	RAM_ADD(1)	V25	SE_D_OUT09(2)	AG9	ADD(1)

Table 15. Signal Locations (Ball to Signal Name) (Continued)

Ball	Signal Name	Ball	Signal Name	Ball	Signal Name)	all	Signal Name
B6	V_{DD}	G28	GND	V26	SE_D_OUT07(3)	AG10	ADD(5)
B7	SE_D_IN11(1)	G29	BP_ACK_IN(17)	V27	SE_D_OUT07(2)	AG11	DATA(5)
B8	SE_D_IN10(3)	G30	BP_ACK_IN(2)	V28	SE_D_OUT06(2)	AG12	GND
B9	SE_D_IN09(3)	H1	SE_SOC_IN19	V29	SE_D_OUT04(1)	AG13	/INTR
B10	GND	H2	SE_D_IN19(1)	V30	GND	AG14	SE_D_OUT30(2)
B11	SE_D_IN08(3)	H3	SE_D_IN16(3)	W1	SE_D_IN25(1)	AG15	SE_D_OUT28(0)
B12	SE_SOC_IN07	H4	V_{DD}	W2	SE_D_IN27(1)	AG16	SE_D_OUT27(3)
B13	SE_D_IN06(1)	H5	not used	W3	SE_D_IN28(1)	AG17	SE_D_OUT26(0)
B14	V_{DD}	H26	BP_ACK_IN(31)	W4	GND	AG18	SE_D_OUT23(0)
B15	SE_SOC_IN04	H27	V_{DD}	W5	SE_SOC_IN28	AG19	GND
B16	SE_D_IN04(3)	H28	BP_ACK_IN(22)	W6	SE_SOC_IN29	AG20	SE_D_OUT22(0)
B17	V_{DD}	H29	BP_ACK_IN(9)	W10	GND	AG21	SE_SOC_OUT5
B18	SE_D_IN03(1)	H30	BP_ACK_IN(6)	W12	GND	AG22	SE_D_OUT20(3)
B19	SE_D_IN02(3)	J1	V_{DD}	W14	GND	AG23	V_{DD}
B20	SE_D_IN01(3)	J2	SE_D_IN20(1)	W17	GND	AG24	SE_D_OUT16(1)
B21	GND	J3	SE_D_IN18(1)	W19	GND	AG25	SE_SOC_OUT4
B22	SE_SOC_IN00	J4	SE_D_IN17(2)	W21	GND	AG26	SE_D_OUT16(3)
B23	RAM_CLK	J5	SE_SOC_IN16	W25	SE_D_OUT09(1)	AG27	GND
B24	RAM_DATA(12)	J6	CELL_START	W26	SE_D_OUT08(2)	AG28	SE_D_OUT14(0)
B25	V_{DD}	J25	BP_ACK_IN(29)	W27	GND	AG29	SE_D_OUT13(3)
B26	RAM_DATA(3)	J26	BP_ACK_IN(21)	W28	SE_D_OUT07(0)	AG30	SE_D_OUT11(1)
B27	RAM_DATA(6)	J27	BP_ACK_IN(18)	W29	SE_D_OUT06(1)	AH1	RAM_ADD(17)
B28	RAM_DATA(1)	J28	BP_ACK_IN(14)	W30	SE_D_OUT04(3)	AH2	BP_ACK_OUT(7)
B29	GND	J29	BP_ACK_IN(4)	Y1	SE_D_IN26(0)	AH3	V_{DD}
B30	RAM_ADD(6)	J30	V_{DD}	Y2	SE_D_IN28(0)	AH4	BP_ACK_OUT(21)
C1	SE_D_IN16(1)	K1	SE_SOC_IN20	Y3	V_{DD}	AH5	GND
C2	CELL_24_START	K2	GND	Y4	SE_D_IN29(2)	AH6	BP_ACK_OUT(26)
C3	V_{DD}	K3	SE_D_IN19(0)	Y5	SE_D_IN30(1)	AH7	GND
C4	SE_SOC_IN14	K4	SE_D_IN17(1)	Y11	GND	AH8	ADD(2)
C5	GND	K5	SE_D_IN17(0)	Y20	GND	AH9	ADD(6)
C6	SE_SOC_IN13	K6	STAT_OUT	Y26	SE_D_OUT10(3)	AH10	DATA(6)
C7	GND	K12	GND	Y27	SE_D_OUT09(3)	AH11	V_{DD}
C8	SE_D_IN12(1)	K14	GND	Y28	V_{DD}	AH12	/WR
C9	SE_D_IN11(2)	K17	GND	Y29	SE_D_OUT07(1)	AH13	SE_D_OUT31(2)
C10	SE_SOC_IN09	K19	GND	Y30	SE_D_OUT05(3)	AH14	SE_D_OUT29(0)
C11	V_{DD}	K25	BP_ACK_IN(27)	AA1	SE_D_IN27(2)	AH15	GND

Table 15. Signal Locations (Ball to Signal Name) (Continued)

Ball	Signal Name	Ball	Signal Name	Ball	Signal Name)	all	Signal Name
C12	SE_SOC_IN08	K26	BP_ACK_IN(20)	AA2	GND	AH16	GND
C13	SE_D_IN07(3)	K27	BP_ACK_IN(19)	AA3	SE_D_IN29(1)	AH17	SE_D_OUT27(2)
C14	SE_D_IN06(2)	K28	BP_ACK_IN(10)	AA4	SE_D_IN31(0)	AH18	SE_D_OUT24(1)
C15	GND	K29	GND	AA5	SE_D_IN31(1)	AH19	SE_D_OUT23(2)
C16	GND	K30	BP_ACK_IN(1)	AA6	BP_ACK_OUT(2)	AH20	V_{DD}
C17	SE_D_IN04(0)	L1	SE_D_IN22(1)	AA12	GND	AH21	SE_D_OUT22(3)
C18	SE_SOC_IN01	L2	SE_D_IN21(0)	AA14	GND	AH22	SE_D_OUT21(3)
C19	SE_D_IN00(3)	L3	V_{DD}	AA17	GND	AH23	SE_D_OUT19(2)
C20	V_{DD}	L4	SE_D_IN19(3)	AA19	GND	AH24	GND
C21	/RAM_OE	L5	SE_D_IN18(0)	AA25	SE_D_OUT12(0)	AH25	SE_D_OUT18(1)
C22	RAM_DATA(15)	L11	GND	AA26	SE_D_OUT11(2)	AH26	GND
C23	RAM_DATA(7)	L20	GND	AA27	SE_D_OUT11(3)	AH27	SE_D_OUT17(2)
C24	GND	L26	BP_ACK_IN(15)	AA28	SE_D_OUT08(0)	AH28	V_{DD}
C25	RAM_DATA(4)	L27	BP_ACK_IN(7)	AA29	GND	AH29	SE_D_OUT14(3)
C26	GND	L28	V_{DD}	AA30	SE_D_OUT06(0)	AH30	SE_D_OUT12(3)
C27	RAM_ADD(15)	L29	BP_ACK_IN(0)	AB1	V_{DD}	AJ1	BP_ACK_OUT(15)
C28	V_{DD}	L30	SE_D_OUT00(1)	AB2	SE_D_IN29(0)	AJ2	GND
C29	BP_ACK_IN(28)	M1	SE_SOC_IN23	AB3	SE_SOC_IN30	AJ3	BP_ACK_OUT(24)
C30	BP_ACK_IN(24)	M2	SE_D_IN22(0)	AB4	SE_SOC_IN31	AJ4	BP_ACK_OUT(28)
D1	SE_SOC_IN17	M3	SE_D_IN20(0)	AB5	BP_ACK_OUT(0)	AJ5	BP_ACK_OUT(30)
D2	SE_D_IN16(2)	M4	GND	AB6	BP_ACK_OUT(4)	AJ6	V_{DD}
D3	/OE	M5	SE_D_IN19(2)	AB25	SE_D_OUT13(2)	AJ7	ADD(7)
D4	GND	M6	SE_D_IN18(2)	AB26	SE_D_OUT12(2)	AJ8	DATA(2)
D5	SE_D_IN15(3)	M10	GND	AB27	SE_SOC_OUT3	AJ9	DATA(7)
D6	SE_D_IN15(0)	M12	GND	AB28	SE_D_OUT10(0)	AJ10	GND
D7	SE_D_IN14(2)	M14	GND	AB29	SE_D_OUT08(1)	AJ11	/ACK
D8	V_{DD}	M17	GND	AB30	V_{DD}	AJ12	SE_D_OUT31(1)
D9	SE_D_IN12(2)	M19	GND	AC1	SE_D_IN28(2)	AJ13	SE_D_OUT29(1)
D10	SE_D_IN11(3)	M21	GND	AC2	SE_D_IN30(0)	AJ14	V_{DD}
D11	SE_D_IN10(0)	M25	BP_ACK_IN(13)	AC3	SE_D_IN31(3)	AJ15	SE_SOC_OUT7
D12	GND	M26	BP_ACK_IN(8)	AC4	V_{DD}	AJ16	SE_CLK
D13	SE_D_IN08(2)	M27	GND	AC5	BP_ACK_OUT(6)	AJ17	V_{DD}
D14	SE_SOC_IN06	M28	BP_ACK_IN(5)	AC26	SE_D_OUT13(0)	AJ18	SE_D_OUT26(2)
D15	SE_D_IN05(3)	M29	SE_D_OUT00(2)	AC27	V_{DD}	AJ19	SE_D_OUT25(1)
D16	SE_SOC_IN03	M30	SE_D_OUT02(1)	AC28	SE_D_OUT11(0)	AJ20	SE_D_OUT24(2)
D17	SE_D_IN03(3)	N1	GND	AC29	SE_D_OUT09(0)	AJ21	GND

Table 15. Signal Locations (Ball to Signal Name) (Continued)

Ball	Signal Name	Ball	Signal Name	Ball	Signal Name)	all	Signal Name
D18	SE_D_IN01(0)	N2	SE_D_IN22(3)	AC30	SE_SOC_OUT2	AJ22	SE_D_OUT23(1)
D19	GND	N3	SE_D_IN21(1)	AD1	SE_D_IN28(3)	AJ23	SE_D_OUT22(2)
D20	SE_D_IN00(1)	N4	SE_D_IN20(2)	AD2	SE_D_IN30(3)	AJ24	SE_D_OUT20(2)
D21	RAM_DATA(10)	N5	SE_D_IN21(3)	AD3	GND	AJ25	V_{DD}
D22	RAM_DATA(11)	N6	SE_D_IN18(3)	AD4	BP_ACK_OUT(8)	AJ26	SE_D_OUT18(2)
D23	V_{DD}	N25	BP_ACK_IN(12)	AD5	BP_ACK_OUT(12)	AJ27	SE_D_OUT19(3)
D24	RAM_ADD(12)	N26	SE_SOC_OUT0	AD6	BP_ACK_OUT(9)	AJ28	SE_D_OUT17(0)
D25	RAM_ADD(9)	N27	BP_ACK_IN(3)	AD25	SE_D_OUT14(1)	AJ29	GND
D26	RAM_ADD(10)	N28	/TEST_MODE	AD26	SE_D_OUT15(2)	AJ30	PLL_VDD
D27	GND	N29	SE_D_OUT01(3)	AD27	SE_D_OUT14(2)	AK1	V_{DD}
D28	BP_ACK_IN(30)	N30	GND	AD28	GND	AK2	BP_ACK_OUT(16)
D29	BP_ACK_IN(23)	P1	SE_D_IN23(3)	AD29	SE_D_OUT10(1)	AK3	ADD(0)
D30	BP_ACK_IN(16)	P2	V_{DD}	AD30	SE_D_OUT08(3)	AK4	ADD(3)
E1	GND	P3	SE_D_IN23(2)	AE1	SE_D_IN30(2)	AK5	GND
E2	CTRL_IN	P4	SE_D_IN23(0)	AE2	V_{DD}	AK6	DATA(0)
E3	GND	P5	SE_D_IN22(2)	AE3	BP_ACK_OUT(5)	AK7	/CS
E4	/SCAN_TRST	P6	SE_D_IN21(2)	AE4	BP_ACK_OUT(11)	AK8	/RD
E5	V_{DD}	P10	GND	AE5	BP_ACK_OUT(13)	AK9	V_{DD}
E6	SE_D_IN15(2)	P12	GND	AE6	GND	AK10	SE_D_OUT31(0)
E7	SE_D_IN15(1)	P14	GND	AE7	BP_ACK_OUT(22)	AK11	SE_D_OUT30(3)
E8	SE_D_IN14(0)	P17	GND	AE9	BP_ACK_OUT(27)	AK12	SE_D_OUT29(3)
E9	SE_SOC_IN12	P19	GND	AE10	BP_ACK_OUT(29)	AK13	GND
E10	SE_SOC_IN11	P21	GND	AE12	DATA(3)	AK14	SE_D_OUT28(2)
E11	SE_SOC_IN10	P25	/PLL_BYPASS	AE13	DATA(4)	AK15	SE_D_OUT28(1)
E12	SE_D_IN09(2)	P26	SE_D_OUT00(0)	AE14	SE_D_OUT31(3)	AK16	SE_D_OUT28(3)
E13	SE_D_IN08(1)	P27	SE_D_OUT01(1)	AE15	SE_D_OUT30(1)	AK17	SE_D_OUT27(1)
E14	SE_D_IN07(1)	P28	SE_D_OUT02(3)	AE16	SE_D_OUT25(2)	AK18	GND
E15	SE_D_IN06(0)	P29	V_{DD}	AE17	SE_D_OUT24(0)	AK19	SE_D_OUT27(0)
E16	SE_D_IN03(2)	P30	SE_D_OUT02(2)	AE18	SE_D_OUT21(1)	AK20	SE_D_OUT25(0)
E17	SE_D_IN03(0)	R1	SE_D_IN24(2)	AE19	SE_D_OUT21(2)	AK21	SE_D_OUT24(3)
E18	SE_D_IN02(1)	R2	SE_D_IN24(0)	AE21	SE_D_OUT18(3)	AK22	V_{DD}
E19	SE_D_IN00(0)	R3	GND	AE22	SE_D_OUT17(1)	AK23	SE_D_OUT23(3)
E20	RAM_DATA(14)	R4	SE_D_IN23(1)	AE24	SE_D_OUT16(2)	AK24	SE_SOC_OUT6
E21	RAM_DATA(9)	R5	SE_SOC_IN22	AE25	GND	AK25	SE_D_OUT21(0)
E22	RAM_DATA(8)	R6	SE_SOC_IN21	AE26	SE_D_OUT15(1)	AK26	GND
E23	RAM_ADD(14)	R25	SE_D_OUT00(3)	AE27	SE_D_OUT15(3)	AK27	SE_D_OUT20(1)

Table 15. Signal Locations (Ball to Signal Name) (Continued)

Ball	Signal Name	Ball	Signal Name	Ball	Signal Name)	all	Signal Name
E24	RAM_ADD(13)	R26	SE_D_OUT01(2)	AE28	SE_D_OUT13(1)	AK28	SE_D_OUT18(0)
E25	RAM_ADD(8)	R27	SE_D_OUT01(0)	AE29	V_{DD}	AK29	SE_D_OUT15(0)
E26	V_{DD}	R28	GND	AE30	SE_D_OUT10(2)	AK30	V_{DD}
E27	RAM_ADD(3)	R29	SE_D_OUT02(0)	AF1	GND		
E28	GND	R30	SE_D_OUT03(2)	AF2	BP_ACK_OUT(1)		
E29	BP_ACK_IN(26)	T1	SE_SOC_IN24	AF3	GND		
E30	GND	T2	SE_D_IN24(1)	AF4	BP_ACK_OUT(14)		

6.4 Pin Descriptions

All inputs except SE_CLK are 5V tolerant. All bidirectional signals are 5 V tolerant. Other outputs are not 5 V tolerant. All pins have pull-ups except /IDDTN.

All inputs have Schmitt triggers, except the SCAN_TDI, SCAN_TMS, /SCAN_TRST, /SCAN_EN, /TEST_MODE, /PLL_BYPASS, DATA[7:0] (which is a bi-di) and RAM_DATA[15:0] (which is also a bi-di).

For outputs, the drive strength listed in the “Type” column (in Table 16 on page 72 through Table 20 on page 81) is in milliamperes. (For example, Out 5 is and output with a drive strength of 5mA.) All switch fabric interface outputs, namely SE_SOC_OUT, SE_D_OUT and BP_ACK_OUT, should be series terminated if the trace is more than four inches long. (Use the series termination resistor as close as possible to the QSE. If the characteristic impedance of the board trace is R ohms, then use a series termination of (R-11) ohms for SE_SOC_OUT, and (R-17) ohms for SE_D_OUT and BP_ACK_OUT .)

Figure 36 shows the signal groupings for the QSE.

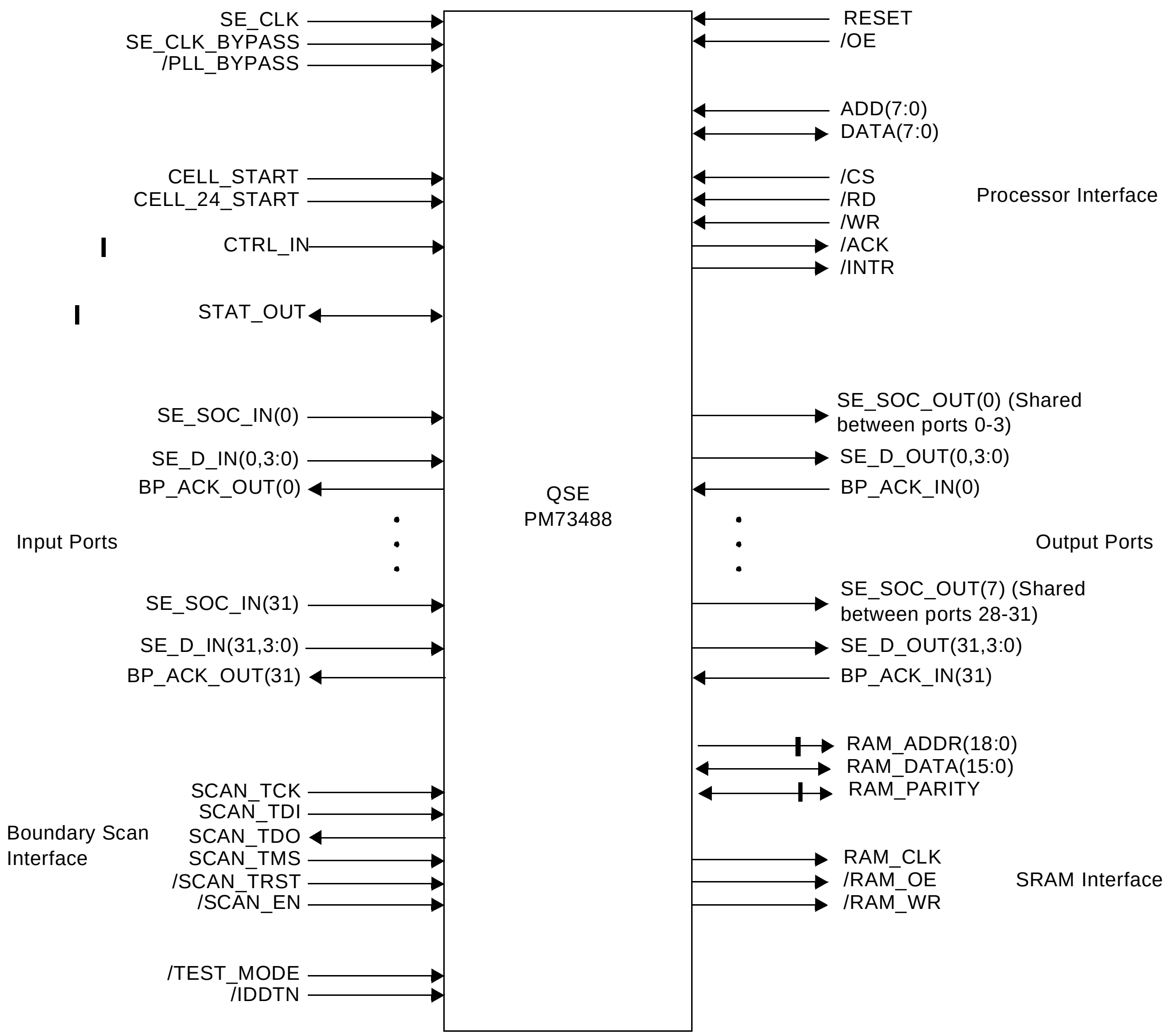


Figure 36. QSE Pinout Block Diagram

6.4.1 Processor Interface Signals

Table 16. Processor Interface Signals (21 Signal Pins)

Signal Name	Ball #	# of Pins	Type	Description
ADD(7:0)	AJ7, AH9, AG10, AF10, AK4, AH8, AG9, AK3	8	In	*Address Bits 7to 0* are part of the 8-bit processor address bus.
DATA(7:0)	AJ9, AH10, AG11, AE13, AE12, AJ8, AF11, AK6	8	Bi 3	*Data Bits 7 to 0 are* part of the 8-bit processor data bus.
/CS	AK7	1	In	*Chip Select* is an active low signal that selects the device for processor access.
/RD	AK8	1	In	*Read* is an active low signal that selects a read cycle.
/WR	AH12	1	In	*Write* is an active low signal that selects a write cycle.
/ACK	AJ11	1	Out 5	*Acknowledge* is an active low signal that indicates the processor cycle is finished.
/INTR	AG13	1	Out 5	*Interrupt* indicates an interrupt is present.

6.4.2 Multicast RAM Interface Signals

Table 17. Multicast RAM Interface Signals (39 Signal Pins)

Signal Name	Ball #	# of Pins	Type	Description
RAM_ADD(18:0)	AF12, AH1, F18, C27, E23, E24, D24, F24, D26, D25, E25, A29, B30, F26, F27, E27, G25, G27, G26	19	Out 5	*RAM Address Bits 18 to 0* are part of the 19-bit SRAM address bus.
RAM_DATA(15:0)	C22, E20, A27, B24, D22, D21, E21, E22, C23, B27, A28, C25, B26, F21, B28, F22	16	Bi 3	*RAM Data Bits 15 to 0 are* part of the 16-bit SRAM data bus.
RAM_PARITY	F19	1	Bi 3	Parity for the RAM Data bits. Generated and checked by the QSE.
RAM_CLK	B23	1	Out 8	*RAM Clock*.

6.4.2 Multicast RAM Interface Signals

Table 17. Multicast RAM Interface Signals (39 Signal Pins)

Signal Name	Ball #	# of Pins	Type	Description
/RAM_OE	C21	1	Out 5	*RAM Output Enable* enables all SRAM output signals.
/RAM_WR	A25	1	Out 5	*RAM Write Enable* strobes data into external SRAM.
NOTE: The external RAM /CE and /ADSC signals are expected to be tied low and the external RAM /ADSP and /ADV signals are expected to be tied high.				

6.4.3 QSE Interface Signals

Table 18. QSE Interface Signals (364 Signal Pins)

Signal Name	Ball #	# of Pins	Type	Description
CELL_START	J6	1	In	The rising edge of *Cell Start* indicates to the QSE it should stop looking for the SE_SOC_IN(31:0) on the input ports. The signal must have the following characteristics: The rising edge should come every 118 clocks exactly. Also, it must be high for at least one clock and low for at least eight clocks during each 118-cycle period. Thus, *Cell Start* must be high for x clocks and low for (118-x) clocks, where $1 \leq x \leq 110$.
CELL_24_START	C2	1	In	*Cell 24 Start* indicates the start of the 4Nth cell time. It should be driven high every 4Nth CELL_START assertions, and should match CELL_START when driven high. Here, N can be any integer ≥ 1, as long as it is the same for all the QSE and QRT devices in the fabric. It is called CELL_24_START because N used to be 6 (so 4N used to be 24) in some legacy systems, but that is no longer relevant.
CTRL_IN (or /No_Data_Out)	E2	1	In	Control In is used to feed in an information packet to the QSE. This information packet can be used to tell the QSE not to accept any incoming cells (which is called a "/No Data In" command) and/or tell it not to send any cells to the next stage (which is called a "/No Data Out" command). There is a software configurable mode which splits the "/No Data In" and "/No Data Out" functionality and assigns them to separate pins. If this mode is turned on, then the CTRL_IN pin performs the "/No Data Out" functionality. (The complementary function of "/No Data In" is performed by the STAT_OUT pin; see below.) That is, whenever CTRL_IN is pulled low, the QSE will not send any cells to the next stage. After Reset, the above mode is on by default; that is, the CTRL_IN pin is configured as "/No Data Out".

Table 18. QSE Interface Signals (364 Signal Pins) (Continued)

Signal Name	Ball #	# of Pins	Type	Description
STAT_OUT (or /No_Data_In)	K6	1	Bi 3	On this pin, the QSE periodically puts out an information packet which indicates if all multicast buffers are empty or not. There is a software configurable mode in which the STAT_OUT pin ceases to be an output pin, and instead turns into an input pin that performs “/No Data In” functionality. (The complementary funcion of “/No Data Out” is performed by the CTRL_IN pin; see above). That is, whenever STAT_OUT is pulled low, the QSE will not accept any incoming cell. After Reset, the above mode is on by default; that is, the STAT_OUT pin is configured as “/No Data In”.
SE_SOC_IN(31:0)	AB4, AB3, W6, W5, V4, U6, U3, T1, M1, R5, R6, K1, H1, F1, D1, J5, A2, C4, C6, E9, E10, E11, C10, C12, B12, D14, A12, B15, D16, A20, C18, B22	32	In	*Receive Cell Start* indicates the start of a cell time. This signal precedes the first nibble of a cell by one clock.

Table 18. QSE Interface Signals (364 Signal Pins) (Continued)

Signal Name	Ball #	# of Pins	Type	Description
SE_D_IN(31:0, 3:0)	**(As follows)**	**128**	**In**	***SE_D_In Ports 31-0, Bits 3 to 0*** **are the nibble-wide data path.**
SE_D_IN31(3:0)	AC3, AG1, AA5, AA4	4	In	
SE_D_IN30(3:0)	AD2, AE1, Y5, AC2	4	In	
SE_D_IN29(3:0)	V6, Y4, AA3, AB2	4	In	
SE_D_IN28(3:0)	AD1, AC1, W3, Y2	4	In	
SE_D_IN27(3:0)	V5, AA1, W2, V3	4	In	
SE_D_IN26(3:0)	U5, T6, U4, Y1	4	In	
SE_D_IN25(3:0)	V2, T5, W1, T4	4	In	
SE_D_IN24(3:0)	U1, R1, T2, R2	4	In	
SE_D_IN23(3:0)	P1, P3, R4, P4	4	In	
SE_D_IN22(3:0)	N2, P5, L1, M2	4	In	
SE_D_IN21(3:0)	N5, P6, N3, L2	4	In	
SE_D_IN20(3:0)	G1, N4, J2, M3	4	In	
SE_D_IN19(3:0)	L4, M5, H2, K3	4	In	
SE_D_IN18(3:0)	N6, M6, J3, L5	4	In	
SE_D_IN17(3:0)	G2, J4, K4, K5	4	In	
SE_D_IN16(3:0)	H3, D2, C1, F3	4	In	
SE_D_IN15(3:0)	D5, E6, E7, D6	4	In	
SE_D_IN14(3:0)	F7, D7, B3, E8	4	In	
SE_D_IN13(3:0)	F9, B4, F10, B5	4	In	
SE_D_IN12(3:0)	A3, D9, C8, A4	4	In	
SE_D_IN11(3:0)	D10, C9, B7, A6	4	In	
SE_D_IN10(3:0)	B8, F12, F13, D11	4	In	
SE_D_IN09(3:0)	B9, E12, A7, A8	4	In	
SE_D_IN08(3:0)	B11, D13, E13, A10	4	In	
SE_D_IN07(3:0)	C13, F14, E14, F15	4	In	
SE_D_IN06(3:0)	A11, C14, B13, E15	4	In	
SE_D_IN05(3:0)	D15, A15, A14, A16	4	In	
SE_D_IN04(3:0)	B16, A19, A17, C17	4	In	
SE_D_IN03(3:0)	D17, E16, B18, E17	4	In	
SE_D_IN02(3:0)	B19, F16, E18, F17	4	In	
SE_D_IN01(3:0)	B20, A21, A24, D18	4	In	
SE_D_IN00(3:0)	C19, A23, D20, E19	4	In	

Table 18. QSE Interface Signals (364 Signal Pins) (Continued)

Signal Name	Ball #	# of Pins	Type	Description
BP_ACK_OUT(31:0)	AF9, AJ5, AE10, AJ4, AE9, AH6, AF8, AJ3, AG7, AE7, AH4, AG6, AF7, AF6, AG5, AK2, AJ1, AF4, AE5, AD5, AE4, AG3, AD6, AD4, AH2, AC5, AE3, AB6, AG2, AA6, AF2, AB5	32	Out 5	*Acknowledge Outputs 31 to 0* assert an acknowledge toward the previous QSE or QRT for unicast cells. It also carries backpressure information for multicast cells.
SE_SOC_OUT(7:0)	AJ15, AK24, AG21, AG25, AB27, AC30, T27, N26	8	Out 8	*Transmit Cell Start* indicates the start of a cell time. This signal precedes the first nibble of a cell by one clock.

Table 18. QSE Interface Signals (364 Signal Pins) (Continued)

Signal Name	Ball #	# of Pins	Type	Description
SE_D_OUT(31:0, 3:0)	**(As follows)**	**128**	**Out 5**	***SE_D_Out Ports 31-0, Bits 3 to 0*** **are 32 nibble-wide output ports.**
SE_D_OUT31(3:0)	AE14, AH13, AJ12, AK10	4	Out 5	
SE_D_OUT30(3:0)	AK11, AG14, AE15, AF14	4	Out 5	
SE_D_OUT29(3:0)	AK12, AF15, AJ13, AH14	4	Out 5	
SE_D_OUT28(3:0)	AK16, AK14, AK15, AG15	4	Out 5	
SE_D_OUT27(3:0)	AG16, AH17, AK17, AK19	4	Out 5	
SE_D_OUT26(3:0)	AF17, AJ18, AF16, AG17	4	Out 5	
SE_D_OUT25(3:0)	AF18, AE16, AJ19, AK20	4	Out 5	
SE_D_OUT24(3:0)	AK21, AJ20, AH18, AE17	4	Out 5	
SE_D_OUT23(3:0)	AK23, AH19, AJ22, AG18	4	Out 5	
SE_D_OUT22(3:0)	AH21, AJ23, AF19, AG20	4	Out 5	
SE_D_OUT21(3:0)	AH22, AE19, AE18, AK25	4	Out 5	
SE_D_OUT20(3:0)	AG22, AJ24, AK27, AF20	4	Out 5	
SE_D_OUT19(3:0)	AJ27, AH23, AF22, AF21	4	Out 5	
SE_D_OUT18(3:0)	AE21, AJ26, AH25, AK28	4	Out 5	
SE_D_OUT17(3:0)	AF23, AH27, AE22, AJ28	4	Out 5	
SE_D_OUT16(3:0)	AG26, AE24, AG24, AF24	4	Out 5	

Table 18. QSE Interface Signals (364 Signal Pins) (Continued)

Signal Name	Ball #	# of Pins	Type	Description
SE_D_OUT15(3:0)	AE27, AD26, AE26, AK29	4	Out 5	
SE_D_OUT14(3:0)	AH29, AD27, AD25, AG28	4	Out 5	
SE_D_OUT13(3:0)	AG29, AB25, AE28, AC26	4	Out 5	
SE_D_OUT12(3:0)	AH30, AB26, AF29, AA25	4	Out 5	
SE_D_OUT11(3:0)	AA27, AA26, AG30, AC28	4	Out 5	
SE_D_OUT10(3:0)	Y26, AE30, AD29, AB28	4	Out 5	
SE_D_OUT09(3:0)	Y27, V25, W25, AC29	4	Out 5	
SE_D_OUT08(3:0)	AD30, W26, AB29, AA28	4	Out 5	
SE_D_OUT07(3:0)	V26, V27, Y29, W28	4	Out 5	
SE_D_OUT06(3:0)	U25, V28, W29, AA30	4	Out 5	
SE_D_OUT05(3:0)	Y30, U27, T25, U26	4	Out 5	
SE_D_OUT04(3:0)	W30, T26, V29, U28	4	Out 5	
SE_D_OUT03(3:0)	T29, R30, U30, T30	4	Out 5	
SE_D_OUT02(3:0)	P28, P30, M30, R29	4	Out 5	
SE_D_OUT01(3:0)	N29, R26, P27, R27	4	Out 5	
SE_D_OUT00(3:0)	R25, M29, L30, P26	4	Out 5	
BP_ACK_IN(31:0)	H26, D28, J25, C29, K25, E29, F28, C30, D29, H28, J26, K26, K27, J27, G29, D30, L26, J28, M25, N25, F30, K28, H29, M26, L27, H30, M28, J29, N27, G30, K30, L29	32	In	*Acknowledge Inputs 31 to 0* receive an acknowledge from the previous QSE or QRT for unicast cells. It also carries backpressure information for multicast cells.

6.4.4 Boundary Scan Signals

Table 19. Boundary Scan Signals (8 Signal Pins)

Signal Name	Ball #	Pin #	Type	Description
SCAN_TCK	G4	1	In	*Scan Test Clock* is an independent clock used to drive the internal boundary scan test logic. (Normal operation = V_{DD} through a pull-up resistor.)

Table 19. Boundary Scan Signals (8 Signal Pins) (Continued)

Signal Name	Ball #	Pin #	Type	Description
SCAN_TDI	B1	1	In	*Scan Test Data Input* is the serial input for boundary scan test data and instruction bits. (Normal operation = V_{DD} through a pull-up resistor.)
SCAN_TDO	F5	1	Out 6	*Scan Test Data Output* is the serial output for boundary scan test data.
SCAN_TMS	F4	1	In	*Scan Test Mode Select* controls the operation of the internal boundary scan test logic. (Normal operation = V_{DD} through a pull-up resistor.)
/SCAN_TRST	E4	1	In	*Scan Test Reset* is used to reset the internal boundary scan test logic. (Normal operation = V_{DD} through a pull-up resistor.)
/SCAN_EN	G6	1	In	*Scan Test Enable* is used to enable the internal scan test logic. (Normal operation = V_{DD} through a pull-up resistor.)
/TEST_MODE	N28	1	In	Test mode. (Normal operation = V_{DD} through a pull-up resistor.)

6.4.5 Miscellaneous Signals

Table 20. Miscellaneous Signals (8 Signal Pins)

Signal Name	Ball #	Pin #	Type	Description
SE_CLK	AJ16	1	In	*QSE Clock* is the main QSE clock.
SE_CLK_BYPASS	AF25	1	In	*QSE Bypass Clock* is the clock used when the Phase Locked Loop (PLL) is bypassed.
/OE	D3	1	In	*Output Enable* is an active low signal that enables the drivers on device outputs.
RESET	AF13	1	In	*Reset* is an active high signal used to initialize or re-initialize the device. SE_CLK must be present for the reset to take effect.
/PLL_BYPASS	P25	1	In	Bypass PLL, and use clock from SE_CLK_BYPASS for the QSE instead of SE_CLK. (Normal operation = V_{DD} through a pull-up resistor.)
PLL_VDD	AJ30	1	In	PLL power. Connect to V_{DD}.
PLL_VSS	AF27	1	In	PLL ground. Connect to GND.
/IDDTN	G5	1	In	Global output disable. (Normal operation = GND.)
VDD	D8, B6, C11, A9, B14, B17, C20, A22, D23, B25, E26, C28, A30, H27, F29, L28, J30, P29, U29, Y28, AB30, AC27, AE29, AF26, AH28, AK30, AG23, AJ25, AH20, AK22, AJ17, AJ14, AH11, AK9, AG8, AJ6, AF5, AH3, AK1, AC4, AE2, Y3, AB1, U2, P2, L3, J1, H4, F2, E5, C3, A1,	52	In	Supply voltage 3.3 V ± 10%.

Table 20. Miscellaneous Signals (8 Signal Pins)

Signal Name	Ball #	Pin #	Type	Description
VSS	N1, M4, K2, G3, E1, E3, F6, D4, B2, U21, P21, M21, AA19, AA17, K17, AA14, K14, K12, W10, U10, P10, K19, M10, AA12, W21, L11, M12, P12, U12, W12, Y11, M14, P14, U14, W14, M17, P17, U17, W17, L20, M19, P19, U19, W19, Y20, C5, C7, A5, D12, B10, A13, C15, C16, A18, D19, B21, C24, A26, C26, F25, D27, B29, E28, G28, E30, M27, K29, N30, R28, T28, V30, W27, AA29, AD28, AF30, AF28, AE25, AG27, AJ29, AH26, AH24, AK26, AG19, AJ21, AK18, AH16, AH15, AK13, AG12, AJ10, AH7, AK5, AH5, AF3, AE6, AG4, AJ2, AD3, AF1, W4, AA2, V1, T3, R3	104	In	Ground.

6.4.6 Total Pin Count

Table 21. Pin Allocations

Signal Name	Pin #	Type	Description
Total processor interface signals	21		
Total multicast RAM signals	39		
Total QSE interface signals	364		
Total boundary scan signals	8		
Total miscellaneous signals	8		
Total signal pins	440		
V_{DD}	52	In	Supply voltage 3.3 V ± 10%.
GND	104	In	Ground.
Total pins	596		

7 PHYSICAL CHARACTERISTICS

Table 22. Absolute Maximum Ratings

Symbol	Parameter	Conditions	Min	Max	Unit
V_{DD}	Supply voltage	With respect to GND	-0.3	3.9	V
I_{OUT}	DC output current, per pin	All outputs	-12	12	mA
T_{STG}	Storage temperature		-65	125	°C
T_J	Junction operating temperature		-40	125	°C
t_R	Input rise time			10	ns
t_F	Input fall time			10	ns
	ESD tolerance			1	kV
	Latch-up current			80	mA

Table 23. Recommended Operating Conditions

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{DD}	Supply voltage		3.0	3.3	3.6	V
V_I	Input voltage		V_{SS} - 0.5	V_{DD}	V_{DD} + 0.3	V
T_A	Ambient operating temperature	See note about junction operating temperature after Table 26 on page 85.	-40	25	85	°C
t_R	Input rise time			1.5	2	ns
t_F	Input fall time			1.5	2	ns

.

Table 24. DC Operating Conditions

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{IH}	High-level TTL input voltage	5 V tolerant inputs	2.2	V_{DD}	5.5	V
V_{IL}	Low-level TTL input voltage		GND-0.3	0.0	0.8	V
V_{OH}	High-level TTL output voltage	$\lvert I_{OH}\rvert \leq$ Specified DC drive current (in Signal Descriptions section)	2.4			V
V_{OL}	Low-level TTL output voltage	$\lvert I_{OL}\rvert \leq$ Specified DC drive current (in Signal Descriptions section)			0.4	V
I_{TYP}	Typical operating current	66 MHz clock rate		900		mA

Table 25. Capacitance

Symbol	Parameter	Conditions	Min	Max	Unit
C_{IN}	Input capacitance		1.5	6	pF
C_{OUT}	Output capacitance		1.5	6	pF
C_{LOAD}	Load capacitance	To meet timing on any output signal		30	pF
NOTES: • Capacitance measured at 25^{o}C. • Sample tested only.					

Table 26. Estimated Package Thermal Characteristics

Symbol	Parameter	Condition	Typ	Unit
θ_{JC}	Junction-to-Case thermal resistance		2.5	°C/Watt
θ_{JA}	Junction-to-Ambient thermal resistance	Still air	12.0	°C/Watt
θ_{JA}	Junction-to-Ambient thermal resistance	200 lfpm	10.2	°C/Watt
θ_{JA}	Junction-to-Ambient thermal resistance	400 lfpm	9.4	°C/Watt
θ_{JA}	Junction-to-Ambient thermal resistance	600 lfpm	8.9	°C/Watt

NOTE: The junction temperature must be kept below 125°C while the device is operating.

8 TIMING DIAGRAMS

All signal names are described in section 6.4 “Pin Descriptions” starting on page 70. Unless otherwise indicated, all output timing delays assume a capacitive loading of 30 pF and that the internal PLL is enabled. The use of the internal PLL is controlled through the /PLL_BYPASS signal. It is recommended that the internal PLL remains enabled

8.1 Microprocessor Timing

A microprocessor cycle starts when the chip select (/CS) and either read (/RD) or write (/WR) are asserted. During read cycles, the QSE asserts /ACK to indicate data on the data bus is valid, and during write cycles the QSE asserts /ACK to indicate the write has finished and data can be removed from the bus. The microprocessor can terminate the current cycle at anytime. As shown in Figure 37, the QSE stops driving the data bus and deasserts the /ACK control line when the cycle terminates. The current cycle terminates when the chip select is deasserted, or when both read and write are deasserted. A new cycle can start once the /ACK has been deasserted. If the cycle was terminated prematurely before the /ACK was asserted, then a new microprocessor cycle can start after one clock cycle.

NOTE: Asserting both read and write lines together while the chip select is asserted (/RD = 0, /WR = 0, and /CS = 0) will cause the device to operate in an undefined manner.

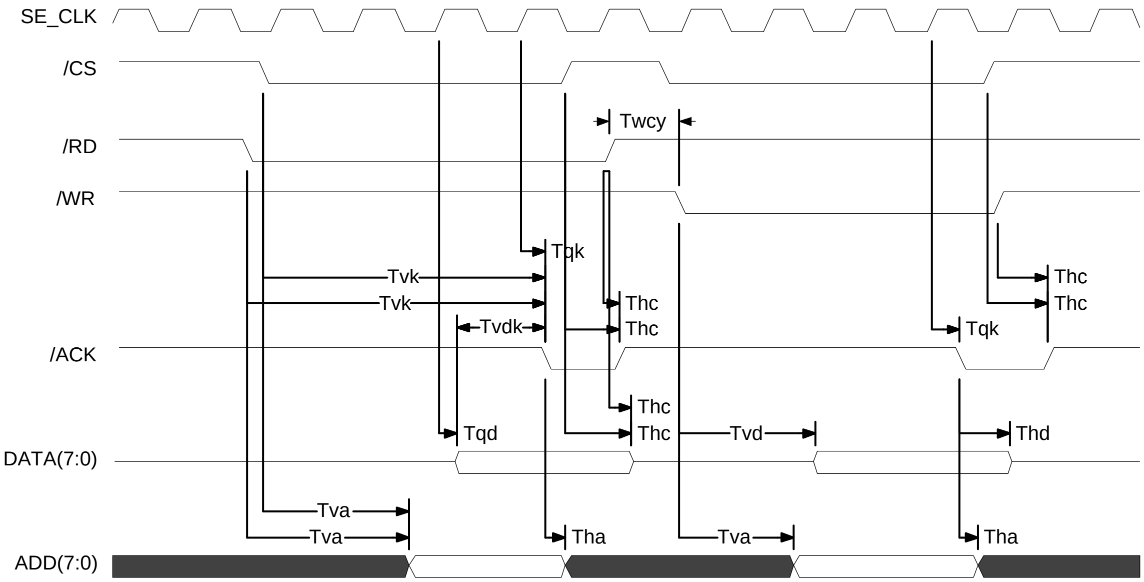


Figure 37. Microprocessor Timing

Table 27. Microprocessor Timing

Symbol	Parameter	Conditions	Min	Max	Unit
Tvk	/ACK valid after /CS, /RD, or /WR, whichever is low last	/ACK	2	118	SE_CLK cycles
Tqk	SE_CLK-to-output delay	/ACK	1	10	ns

Table 27. Microprocessor Timing (Continued)

Symbol	Parameter	Conditions	Min	Max	Unit
Tqd	SE_CLK-to-output delay	DATA(7:0)	1	13.5	ns
Tvdk	Data valid prior to /ACK assertion	DATA(7:0)	SE_CLK cycle - 10.3		ns
Tvd	Data valid after /CS or /WR, whichever is low last	DATA(7:0)		1	SE_CLK cycle
Tva	Address valid after /CS, /RD, or /WR, whichever is low last	ADD(7:0)		1	SE_CLK cycles
Tha	Address hold after /ACK assertion	ADD(7:0)	0		ns
Thd	Data hold after /ACK assertion for write cycle	DATA(7:0)	0		ns
Thc	Hold time after /CS, /RD, or / WR, whichever is high first	/ACK, DATA(7:0)	1.2		ns
Twcy	Wait time between two consecutive cycles	/CS, /RD, /WR	1		SE_CLK cycles

8.2 RAM Timing

The RAM interface is a synchronous interface, with respect to the RAM_CLK. Each read or write operation lasts for at least two clock cycles because of the internal 32-bit data bus. Recall that the RAM_DATA bus is covered by one bit of parity, named RAM_PARITY; this parity bit signal follows the same timing constraints and timing guarantees as the rest of the data bus.

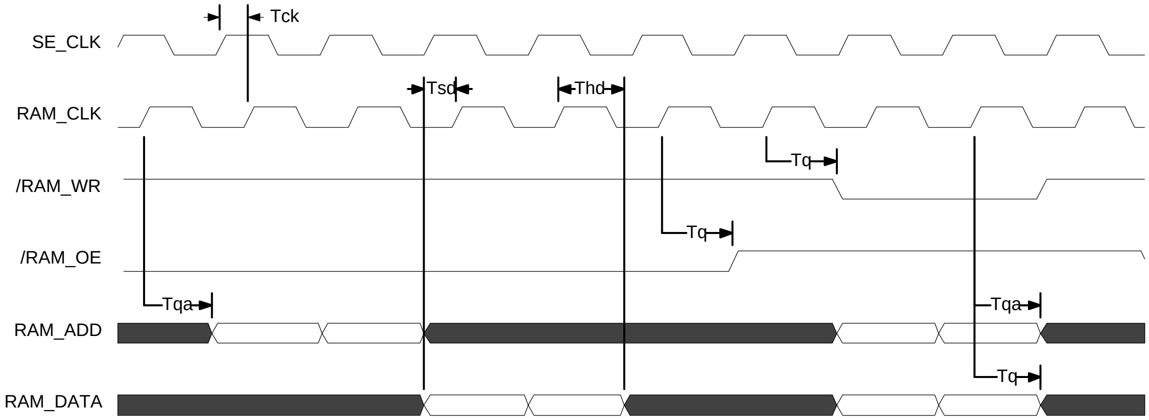


Figure 38. RAM Interface

Table 28. RAM Interface Timing

Symbol	Parameter	Conditions	Min	Max	Unit
Tck	SE_CLK to RAM_CLK	RAM_CLK	0.5	2.5	ns

Table 28. RAM Interface Timing (Continued)

Symbol	Parameter	Conditions	Min	Max	Unit
Tq	RAM_CLK-to-output delay	/RAM_WR, /RAM_OE, RAM_DATA	1.5	9	ns
Tsd	RAM_CLK setup time	RAM_DATA	5.2		ns
Thd	RAM_CLK hold time	RAM_DATA	0		ns
Tqa	RAM_CLK-to-output delay	RAM_ADD	1.5	10	ns

8.3 QSE Interface Timing

Figure 39 shows the bit-level timing for the QSE.

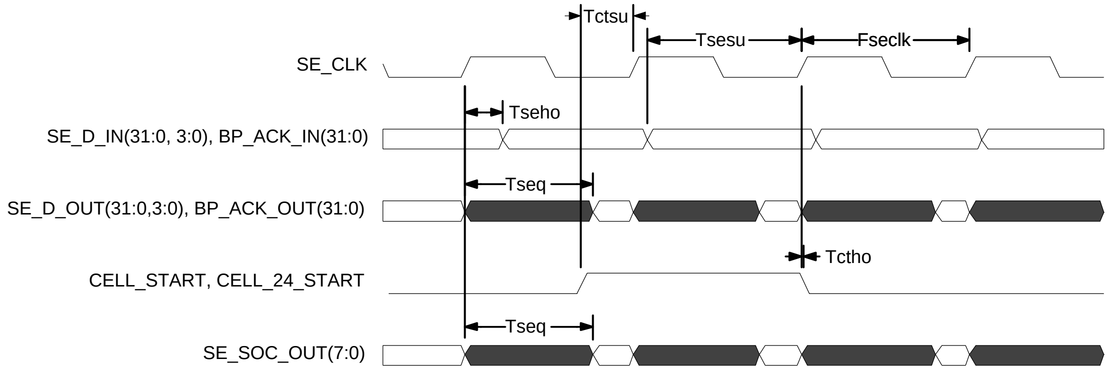

Tctsu
Tsesu
Fseclk
SE_CLK
Tseho
SE_D_IN(31:0, 3:0), BP_ACK_IN(31:0)
Tseq
SE_D_OUT(31:0,3:0), BP_ACK_OUT(31:0)
Tctho
CELL_START, CELL_24_START
Tseq
SE_SOC_OUT(7:0)

Figure 39. QSE Bit-Level Timing

Symbol	Parameter	Signals	Min	Max	Unit
Fseclk	Frequency of SE_CLK	SE_CLK	35[a]	66	MHz
Tctsu	Control signal setup	CELL_START, CELL_24_START	8.0		ns
Tctho	Control signal hold	CELL_START, CELL_24_START	0		ns
Tseq	Output delay from SE_CLK	SE_D_OUT (15 pF), BP_ACK_OUT(31:0), SE_SOC_OUT(7:0)[b]	1	6	ns
	Output delay skew *	SE_D_OUT(0,3:0) and SE_SOC_OUT SE_D_OUT(1,3:0) and SE_SOC_OUT SE_D_OUT(2,3:0) and SE_SOC_OUT SE_D_OUT(3,3:0) and SE_SOC_OUT		1.9	ns

	Input delay skew *	SE_D_IN(0,3:0) and SE_SOC_IN(0)
		SE_D_IN(1,3:0) and SE_SOC_IN(1)
		SE_D_IN(2,3:0) and SE_SOC_IN(2)
		SE_D_IN(3,3:0) and SE_SOC_IN(3)
				3.5	ns

* When the phase aligners are turned on, Tsesu and Tseho are no longer defined. However, the maximum input and output skew and jitter on these signals with respect to the SE_SOC_IN is constrained to specification listed in this table.					

a. For the phase aligners to lock.
b. In real applications the output skew will be lower than 1.9ns. The reason for this is as follows. When all pins are equally loaded, SE_SOC_OUT is faster than all the SE_D_OUTs by (upto) 1.9ns. However, in real applications SE_SOC_OUT will have fan-out of four, and hence will be loaded four times as much as the other pins. This will slow down SE_SOC_OUT and hence lower the output skew.

8.4 Miscellaneous Timing

Timing for the CTRL_IN, STAT_OUT, TEST_MODE, IDDTN and DEBUG(1:0) signals is shown in Table 29.

Table 29. CTRL_IN, STAT_OUT, TEST_MODE and DEBUG Timing

Symbol	Parameter	Signals	Min	Max	Unit
Tdasu	Control signal setup	STAT_OUT (when it behaves as i/p)	4		ns
Tdasu	Control signal setup	CTRL_IN	4		ns
Tdaho	Control signal setup	TEST_MODE	10		ns
Tdaho	Control signal setup	IDDTN	10		ns
Tdaho	Control signal hold	STAT_OUT (when it behaves as i/p)	0		ns
Tdaho	Control signal hold	CTRL_IN	0		ns
Tdaho	Control signal hold	TEST_MODE	10		ns
Tdaho	Control signal hold	IDDTN	10		ns
Tdaq	Output delay from SE_CLK	STAT_OUT (when it behaves as o/p)	1	10	ns
Tdeq	Output delay from SE_CLK	DEBUG(1,0),	1.5	14	ns

Figure 40 shows the reset pin (RESET) timing. The RESET signal must be asserted for a minimum time (*Tres*) to be properly processed internal to the QSE. The QSE remains in reset while RESET is asserted, and starts performing normally after *Trstproc*.

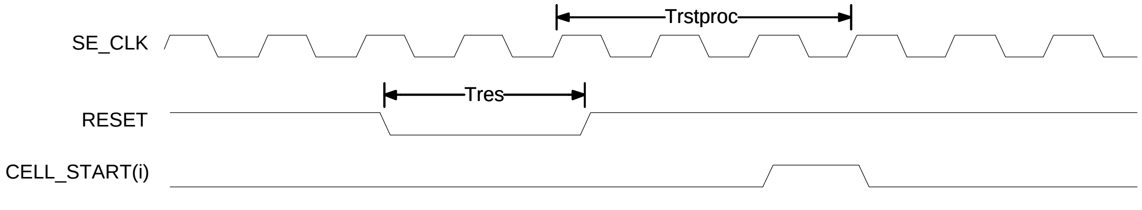


Figure 40. Reset Timing

Symbol	Parameter	Signals	Min	Max	Unit
Tres	Reset assertion time	RESET	10		SE_CLK periods
Trstproc	Reset processing time	RESET	2	3	SE_CLK periods

NOTE: RESET assertion and deassertion is asynchronous to the clock.

Timing information for the SOC, BP, and ACK is given in Table 30.

Table 30. Valid Window Timing

Symbol	Parameter	Min	Max	Unit
Vsoc	SOC valid window	Local_CELL_START - 8	Local_CELL_START	SE_CLK periods
Vbprec	Valid window when BP is accepted by QSE	SE_SOC_OUT + 0	Local_CELL_START + 60	SE_CLK periods
Vbpgen	Valid window when BP is generated by QSE	Local_CELL_START + 15 *(But in early BP mode: Local CELL_START + 0 See “BP_CONTROL_REGISTER” on page 109)*	Local_CELL_START + 35 *(But in early BP mode: Local_CELL_START + 15 See“BP_CONTROL_REGISTER” on page 109)*	SE_CLK periods
Vack	Valid window when ACK is accepted by QSE	SE_SOC_OUT + 0	(Next cell time’s) Local_CELL_START - 8	SE_CLK periods

Figure 41 shows the timing for the JTAG port. The /SCAN_TRST signal is asynchronous to SCAN_TCK.

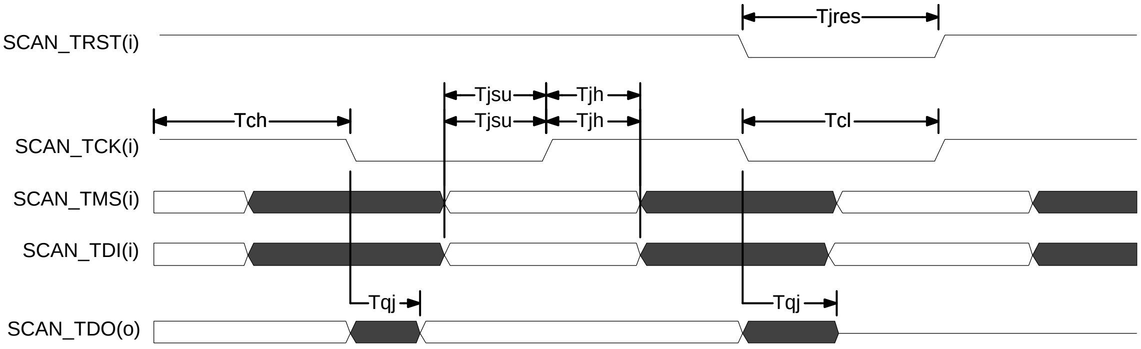


Figure 41. JTAG Timing

Symbol	Parameter	Signals	Min	Max	Unit
	SCAN_TCK frequency			10	MHz
Tch	SCAN_TCK high		40		ns

Symbol	Parameter	Signals	Min	Max	Unit
Tcl	SCAN_TCK low		40		ns
Tjh	SCAN_TCK hold time	SCAN_TMS, SCAN_TDI	20		ns
Tjsu	SCAN_TCK setup time	SCAN_TMS, SCAN_TDI	20		ns
Tjres	/SCAN_TRST low		40		ns
Tqj	SCAN_TCK-to-output delay /SCAN_TRST-to-output delay	SCAN_TDO		20	ns

9 MICROPROCESSOR PORTS

9.1 Microprocessor Ports Summary

NOTES:

- All read/write port bits marked “Not used” must be written with the value 0 to maintain software compatibility with future versions.
- All port bits marked “Reserved” should not be written. Software modifications to these locations after setup may cause incorrect operation.
- For 16-bit registers at addresses *X* and (*X*+1), bit 15 is address *X* bit 7 and bit 0 is address (*X*+1) bit 0.
- For 32-bit registers at addresses *X* to (*X*+3), bit 31 is address *X* bit 7 and bit 0 is address (*X*+3) bit 0. For example, the INPUT_PORT_ENABLE register.
- For 128-bit registers at addresses *X* to (*X*+F_h), nibble 31 is address *X* bits 7 to 4 and nibble 0 is address (*X*+F_h) bits 3 to 0. For example, the INPUT_MARKED_CELLS_COUNT register.
- Registers marked with a “t” should only be modified while the chip is in software reset.

Table 31. Microprocessor Ports Summary

Address (in Hex)	Name	Read or Write	Description
Chip Control/Status Registers			
0	REVISION	R	Contains the device revision number (namely, 01_h).
1	CHIP_MODE	R/Wt	Assorted chip-configuration bits.
2-3	MULTICAST_GROUP_INDEX	R/W	Multicast group to be modified or read.
4-7	MULTICAST_GROUP_VECTOR	R/W	Set of destinations comprising the multicast group.
8	MULTICAST_GROUP_OP	R/W	Operation to be performed.
9-A	UC/MC_FAIRNESS_REGISTER	R/W	Unicast/Multicast behavior for cells of the same priority.
B	EXTENDED_CHIP_MODE	R/Wt	Extended chip mode
C	MULTICAST_GROUP_INDEX_MSB	R/W	Highest byte of Multicast group to be modified or read.
D-F	RESERVED	—	
Port Control/Status Registers			
10-13	INPUT_PORT_ENABLE	R/W	Enable input ports and associated interrupts.
14-17	OUTPUT_PORT_ENABLE	R/W	Enable output ports and associated interrupts.
18-27	INPUT_MARKED_CELLS_COUNT	R	Count of marked cells arriving at inputs.
28-37	OUTPUT_MARKED_CELLS_COUNT	R	Count of marked cells leaving at outputs.
38-3B	PARITY_ERROR_PRESENT	R	Parity error status on inputs during the last cell time.
3C-3F	PARITY_ERROR_LATCH	R	Indicates if any parity errors have occurred since the last read.
40-43	PARITY_ERROR_INT_MASK	R/W	Enables/disables interrupt due to parity error.

Table 31. Microprocessor Ports Summary (Continued)

Address (in Hex)	Name	Read or Write	Description
44-47	SE_INPUT_PORT_FAIL_PRESENT	R	Indicates absence of special pattern on SOC or invalid cell present code on data lines or invalid idle cell code on data lines during the last cell time.
48-4B	SE_INPUT_PORT_FAIL_LATCH	R	Indicates if an SE_INPUT_PORT_FAIL has occurred since the last read.
4C-4F	BP_ACK_FAIL_PRESENT	R	Indicates absence of special coding on BP_ACK_IN line on output ports during the last cell time.
50-53	BP_ACK_FAIL_LATCH	R	Indicates if a BP_ACK_FAIL has occurred since the last read.
54-57	BP_REMOTE_FAIL_PRESENT	R	Indicates absence of back pressure on BP_ACK_IN line on output ports during the last cell time.
58-5B	BP_REMOTE_FAIL_LATCH	R	Indicates if a BP_REMOTE_FAIL condition has occurred since the last read.
5C-7F	RESERVED	—	
Switch Control/Status Registers			
80	CONTROL_REGISTER	R/Wt	Various switch parameters.
81	INTERRUPT_STATUS_REGISTER	R	Identifies if an interrupt condition is present.
82	MULTICAST_AGGREGATE_OUTPUT_MODE	R/Wt	Aggregate mode for multicast cells.
83	UNICAST_AGGREGATE_OUTPUT_MODE	R/Wt	Aggregate mode for unicast cells.
84	SWITCH_FABRIC_ROW	R/Wt	Row number in switch fabric.
85	SWITCH_FABRIC_COLUMN	R/Wt	Column number in switch fabric.
86	CELL_START_OFFSET	R/Wt	Offset between internal and external CELL_START signals.
87	BP_CONTROL_REGISTER	R/Wt	Control backpressure functionality.
88	ACK_PAYLOAD	R/W	Payload for ACK packet when ACK needs to be generated by the QSE for parity fail and regular congestion.
89	GANG_DEAD_ACK	R/W	Payload for ACK packet when ACK needs to be generated by the QSE because the entire gang is dead.
8A	EXTENDED_SWITCH_MODE	Rt	Extended switch control register.
8B-EF	RESERVED	—	

9.2 Note on Error Detection and Reporting

The QSE detects six classes of errors and each error in every class is reported using two bits:

- Error_present: There is an error at the present moment
- Error_latched: There was an error sometime in the past, between now and the last time this register was read.

Of these two bits, errors latched in the Error_latched registers can be further used to generate interrupts to the microprocessor.

The six detected classes of errors fall into two categories:

Category 1: Errors that can be associated with an input or output port.
Errors in this category are only detected if the corresponding port is enabled.

- Input port failed: This means that one of the SOC_IN or DATA_IN wires is stuck or glitchy. The error_present register is at address 44-47, and the error_latched register is at address 48-4B. You can stop checking for this error by turning off the appropriate input ports using the register at address 10-13.
- BpAck failed: This means that one of the BPACK_IN wires is stuck or glitchy. The error_present register is at address 4C-4F, and the error_latched register is at address 50-53. So you can stop checking for this error by turning off the appropriate output port using the register at address 14-17.
- Remote failure: This means that the downstream QSE did not sent a BP packet on some BPACK_IN wire during some cell-time. By implication, it means that one of the SOC_OUT or DATA_OUT wires is stuck or glitchy (to which the downstream QSE responds by withholding the BP packet). The error_present register is at address 54-57, and the error_latched register is at address 58-5B. You can stop checking for this error by turning off the appropriate output port using the register at address 14-17.
- Parity error in a cell. The error_present register is at address 38-3B, and the error_latched register is at address 3C-3F. You can stop checking for this error by turning off the appropriate input ports using the register at address 10-13. A separate set of registers at address 40-43 allow you to disable interrupts due to this error. You can also globally disable all parity checks on input ports using the CHIP_MODE register (bit 6).

Each of the above four classes of errors has a "summary" bit in the interrupt status register (ISR) at address 81. The summary bit for a class is set if any enabled error is latched in that class. An actual interrupt to the microprocessor (due to these classes of errors) will be generated if any of the four summary bits in the ISR are set and if the global interrupt mask is enabled.

Category 2: Errors in this category are global to the entire chip.

- CSTART is out of lock. The error_present register is at address 80 (bit 7), and the error_latched register is at address 8A (bit 0). You can turn off the interrupt from this error using the register at address 80 (bit 6). If this error is causing an interrupt, this is indicated by bit 4 of the ISR (address 81).
- Parity error in external MC connection RAM. There is no error_present register. The error_latched register is at address B(bit 6). You can turn off the parity check using the register at address B(bit 4) and you can disable interrupts due to this error using address B(bit 5). This error will cause an interrupt if it has been latched and if interrupts from this error have not been disabled using the register at address B (bit 5).

9.3 Microprocessor Ports Bit Definitions

NOTE: The bits reset to 0_b unless otherwise indicated.

9.3.1 REVISION

This register contains the device revision number.

Address: 0_h
Type: Read Only
Format: Refer to the following table.

Field (Bits)	Description
REVISION (7:0)	Revision number of the QSE device. Revision numbers start at 0.

9.3.2 CHIP_MODE

Address: 1_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
ENABLE_STAT_PINS (7)	1 Enable StatOut and CtrlIn pin functionality 0 StatOut behaves like No Data In, and Ctrl In behaves like No Data Out.
PARITY_CHECK (6)	1 Parity checks on cell header disabled. 0 Normal operation.
/NO_DATA_OUT (5)	Current value at the /NO_DATA_OUT pin.
/NO_DATA_IN (4)	Current value at the /NO_DATA_IN pin.
MULTICAST_MODE (3)	1 External RAM present. 0 No external RAM.
CHIP_HARDWARE_RESET (2)	1 Writing a one to this bit will put the chip is in hardware reset (except the processor interface, which remains untouched). 0 Writing a zero to this bit will take the chip out of hardware reset. Upon pin-reset, this bit comes up as a one. A zero must be explicitly written to this bit before the chip can function normally.
SWITCH_MODE (1)	1 Double switch mode. 0 Single switch mode.
Reserved (1:0)	Write with a 0 to maintain future software compatibility.

9.3.3 MULTICAST_GROUP_INDEX_REGISTER

Address: 2-3_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
MC_ADD (15:0)	Multicast group index to be used by MULTICAST_GROUP_OP (refer to section “9.3.5 MULTICAST_GROUP_OP” on page 97). This register has bits 15 to 0 of the index. The MULTICAST_GROUP_INDEX_MSB register has the remaining.

9.3.4 MULTICAST_GROUP_VECTOR_REGISTER

Address: 4-7_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
MC_GROUP (31:0)	Multicast Group Vector (MGV) data to be used by MULTICAST_GROUP_OP (refer to section “9.3.5 MULTICAST_GROUP_OP” on page 97). Address 45_h bit 7 corresponds to the highest register bit, and 42_h bit 0 corresponds to the lowest register bit. Depending on the multicast gang mode, only certain bits are active, and the active bits are as follows: Gang 1 mask FFFFFFFF_h Gang 2 mask 0F0F0F0F_h Gang 4 mask 03030303_h 1 Enables the transmission of a cell on the multicast group corresponding to the active bit number. 0 Disables the transmission of a cell on the multicast group corresponding to the active bit number.

9.3.5 MULTICAST_GROUP_OP

Address: 8_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
Not used (7:2)	Write with a 0 to maintain future software compatibility.
INC_BIT (1)	Increment Bit. 1 Autoincrement MULTICAST_GROUP_INDEX_REGISTER (refer to section “9.3.3 MULTICAST_GROUP_INDEX_REGISTER” on page 96) after each operation. 0 Leave MULTICAST_GROUP_INDEX_REGISTER unchanged.
OPERATION_BIT (0)	Operation Bit. 1 Enables the write of MULTICAST_GROUP_VECTOR_REGISTER to the multicast group vector equal to the address referenced by MULTICAST_GROUP_INDEX_REGISTER. 0 Enables the read of MULTICAST_GROUP_VECTOR_REGISTER from the multicast group vector equal to the address referenced by MULTICAST_GROUP_INDEX_REGISTER.

9.3.6 UC/MC_FAIRNESS_REGISTER

Address: 9-A_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
UPPER PORTS (15:8)	Suppose a UC cell and an MC cell of the same priority are contending for the same output port, where the output port number is between 31 and 16. If *x* bits are set, then the UC cell has an x/8 probability of winning over the MC cell. For example, if (any) 4 of the 8 bits are set, then a tie is broken randomly with a 50-50 chance of either one winning. If none of the bits are set, then MC always wins, and if all the bits are set then UC always wins. This register resets to $3A_h$.
LOWER PORTS (7:0)	Same as above, except this register controls output ports between 15 and 0. Another difference is that this register resets to $A3_h$.

9.3.7 EXTENDED_CHIP_MODE

Address: B_h
Type: Read/Write

Format: Refer to the following table.

Field (Bits)	Description
Not used (7)	Write with a 0 to maintain future software compatibility.
RAM_PARITY_ERR_SENSED (6)	1: A parity error was sensed in the external multicast RAM 0: No parity error was sensed or parity is not enabled
RAM_PARITY_INT_ENABLE (5)	1: Enable interrupt on External Multicast Vector RAM parity error 0: No interrupt on RAM parity error
RAM_PARITY_ENABLE (4)	1: Enable parity checking for the external multicast vector RAM 0: Disable parity checking for the external multicast vector RAM
Not used (3:1)	Write with a 0 to maintain future software compatibility.
SHORT_TAG_ENABLE	1: Rotate only 5 nibbles of the routing tag. 0: Rotate all 8 nibbles of the routing tag. When the QSE receives a unicast cell, it looks at the initial portion of the cell's routing tag, and interprets it to be the destination gang of the cell. Before sending the cell out on that destination, the QSE cyclically shifts the routing tag leftwards. The purpose of this shift is to move new bits into the initial portion of the routing tag, thus making the routing tag suitable for use by the next-stage QSE. The amount of the rotation is equal to (5 - UC output gang mode) bits. (For a discussion on UC output gang mode, see section "9.3.25 UNICAST_AGGREGATE_OUTPUT_MODE" on page 106.) If SHORT_TAG_ENABLE is set to 1, then only 5 nibbles are rotated. Hence, the last 3 nibbles are left untouched, and they could potentially be used by the traffic manager to send diagnostic information. The QRT currently does not *NOT* use these 3 nibbles for anything. Therefore, when the QSE is used in conjunction with the QRT, there is no advantage to short tags, and the SHORT_TAG_ENABLE bit may remain at the reset-default value of 0.

9.3.8 MULTICAST_GROUP_INDEX_MSB

Address: C_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
MGI_MSB (1:0)	Bits 17 and 16 of the multicast group index. Use along with the MULTICAST_GROUP_INDEX_REGISTER

9.3.9 INPUT_PORT_ENABLE

Address: $10\text{-}13_h$

Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Enable input port *x*. 0 Disable input port *x* and interrupts due to SE_INPUT_PORT_FAIL_PRESENT (refer to section "9.3.16 SE_INPUT_PORT_FAIL_PRESENT" on page 101).

9.3.10 OUTPUT_PORT_ENABLE

Address: $14\text{-}17_h$
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Enable output port *x*. 0 Disable output port *x* and interrupts due to BP_ACK_FAIL_PRESENT (refer to section "9.3.18 BP_ACK_FAIL_PRESENT" on page 102) and BP_REMOTE_FAIL_PRESENT (refer to section "9.3.20 BP_REMOTE_FAIL_PRESENT" on page 103).

9.3.11 INPUT_MARKED_CELLS_COUNT

Address: $18\text{-}27_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
Nibble 31 - Nibble 0	Nibble *x*: Number of cells mod 16 on input port *x* that had Tag(9,1) set to 1. All marked cells that enter on that port will be counted, even if they are discarded later on due to other reasons (e.g. multicast cell with parity errored header, or a multicast cell sent in violation of back-pressure.)

9.3.12 OUTPUT_MARKED_CELLS_COUNT

Address: $28\text{-}37_h$
Type: Read only

Format: Refer to the following table.

Field (Bits)	Description
Nibble 31 - Nibble 0	Nibble *x*: Number of cells mod 16 on output port *x* that had Tag(9,1) set to 1.

9.3.13 PARITY_ERROR_PRESENT

Address: 38-3B$_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Indicates if a parity error was present on the input port data lines during the last cell time. Bit *x*: 1 Error detected on input port *x*. 0 No error on input port *x*.

9.3.14 PARITY_ERROR_LATCH

Address: 3C-3F$_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Indicates if a parity error occurred on an input port since the last time this register was read. Bit *x*: 1 Error detected on input port *x*. 0 No error on input port *x*. Reset to 0 on read.

9.3.15 PARITY_ERROR_INT_MASK

Address: 40-43$_h$
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Enable interrupt due to parity condition latched for input port *x*. 0 Disable interrupt due to parity condition latched for input port *x*.

9.3.16 SE_INPUT_PORT_FAIL_PRESENT

Address: 44-47$_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit x: 1 Indicates that one or more of the following conditions were true for input port *x* during the last cell time: - Special pattern on SE_SOC_IN is absent. - Presence of an invalid cell present code. - Presence of an invalid idle cell code. 0 Normal.

9.3.17 SE_INPUT_PORT_FAIL_LATCH

Address: $48\text{-}4B_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 An SE_INPUT_PORT_FAIL_PRESENT (refer to section “9.3.16 SE_INPUT_PORT_FAIL_PRESENT” on page 101) has occurred on input port *x* since the last time this register was read. 0 Normal. Reset to 0 on read.

9.3.18 BP_ACK_FAIL_PRESENT

Address: $4C\text{-}4F_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Indicates absence of special pattern on the BP_ACK line for output *x*. 0 Normal

9.3.19 BP_ACK_FAIL_LATCH

Address: $50\text{-}53_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 A BP_ACK_FAIL_PRESENT has occurred on output *x* since the last time this register was read. 0 Normal. Reset to 0 on read.

9.3.20 BP_REMOTE_FAIL_PRESENT

Address: 54-57$_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Indicates absence of back pressure on BP_ACK line for output *x* during last cell time. 0 Normal.

9.3.21 BP_REMOTE_FAIL_LATCH

Address: 58-5B$_h$
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
(31:0)	Bit *x*: 1 Indicates a BP_REMOTE_FAIL_PRESENT (refer to section “9.3.18 BP_ACK_FAIL_PRESENT” on page 102) has occurred on output *x* since the last time this register was read. 0 Normal. Reset to 0 on read.

9.3.22 CONTROL_REGISTER

Address: 80$_h$
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
CELL_START_IN_LOCK (7)	1 CELL_START is in lock. 0 CELL_START is not in lock. This bit may be viewed as the complemented form of the error_present indicator CELL_START_OUT_OF_LOCK_PRESENT. The corresponding error_latched indicator may be found in section “9.3.32 EXTENDED_SWITCH_MODE” on page 110. For a general discussion on error_present and error_latched indicators, see section “9.2 Note on Error Detection and Reporting” on page 94.
CELL_START_OUT_OF_LOCK_ INT_MASK (6)	1 Interrupt when CELL_START out of lock. 0 No interrupt when CELL_START is out of lock.

Field (Bits)	Description
Reserved (5)	Write with a 0 to maintain future software compatibility.
PHASE_ALIGNER_MODE (4)	1 Phase aligner off. 0 Phase aligner on. This bit should remain cleared (i.e. 0) for normal operation.
Reserved (3)	Write with a 0 to maintain future software compatibility.
Reserved (2)	Write with a 0 to maintain future software compatibility.
INT_ENABLE (1)	1 Global interrupt enabled. 0 Global interrupt disabled. The interrupt will remain asserted as long as this bit is set and at least one of the bits in the interrupt status register is set. Unfortunately, setting this bit to 0 does not disable interrupts due to ram parity-error and cstart out-of-lock. They need to be disabled separately. Ram parity-error interrupt may be disabled using bit 5 of “EXTENDED_CHIP_MODE” on page 97. Cstart out-of-lock interrupt may be disabled using bit 6 of “CONTROL_REGISTER” on page 103.
SW_RESET (0)	1 Writing a one to this bit will put the chip in software reset. This means that the processor interface will remain untouched, and the remaining blocks in the chip will be reset only some portion of their state (depending on the discretion of the designer). 0 Writing a zero to this bit will take the chip out of software reset. Upon pin-reset, this bit comes up as a one. A zero must be explicitly written to this bit before the chip can function normally.

9.3.23 INTERRUPT_STATUS_REGISTER

Address: 81_h
Type: Read only
Format: Refer to the following table.

Field (Bits)	Description
Not used (7:4)	Driven with a 0. Mask on reads to maintain compatibility with future versions.
CSTART_OUT_OF_LOCK (4)	CELL_START out-of-lock interrupt is enabled and CELL_START out-of-lock latch is on.
PARITY_ERROR (3)	An input in which PARITY_ERROR_INT_MASK (refer to section “9.3.15 PARITY_ERROR_INT_MASK” on page 100) is enabled has a latched parity error.
INPUT_PORT_FAIL (2)	An enabled input has a latched SE_INPUT_PORT_FAIL_LATCH (refer to section “9.3.17 SE_INPUT_PORT_FAIL_LATCH” on page 102).
BP_ACK_FAIL (1)	An enabled output has a latched BP_ACK_FAIL_LATCH (refer to section “9.3.21 BP_REMOTE_FAIL_LATCH” on page 103).
BP_REMOTE_FAIL (0)	An enabled output has a latched BP_REMOTE_FAIL_LATCH (refer to section “9.3.21 BP_REMOTE_FAIL_LATCH” on page 103).

This register can be used to check status in polled mode even if interrupts are disabled in the CONTROL_REGISTER (refer to section “9.3.22 CONTROL_REGISTER” on page 103).

9.3.24 MULTICAST_AGGREGATE_OUTPUT_AND_INPUT_MODES

Address: 82_h
Type: Read/Write
Note: Also called multicast gang mode register
Format: Refer to the following table.

Field (Bits)	Description
Not used (7)	Write with a 0 to maintain future software compatibility.

Field (Bits)	Description
MULTICAST_AGG_OUT (6:4)	Selects aggregate of N. That is, N consecutive outputs are treated as a single output by the switch for multicast traffic. (2:0) = 3 - 7 are invalid. (2:0) = 2, N = 4. (2:0) = 1, N = 2. (2:0) = 0, N = 1. Aggregate mode is also called "gang mode" in other parts of this document. *Note: The unicast output gang mode (see section "9.3.25 UNICAST_AGGREGATE_OUTPUT_MODE" on page 106) must be set to a value greater than or equal to the multicast output gang mode.*
Not used (3)	Write with a 0 to maintain future software compatibility.
MULTICAST_AGG_IN (2:0)	Selects aggregate of N. That is, N consecutive inputs are treated as a single input by the switch for multicast traffic. (2:0) = 3 - 7 are invalid. (2:0) = 2, N = 4. (2:0) = 1, N = 2. (2:0) = 0, N = 1. Aggregate mode is also called "gang mode" in other parts of this document.

9.3.25 UNICAST_AGGREGATE_OUTPUT_MODE

Address: 83_h
Type: Read/Write
Note: Also called unicast gang mode register

Format: Refer to the following table.

Field (Bits)	Description
UNICAST_AGG_OUT (2:0)	Selects aggregate of N. N consecutive outputs are treated as a single output by the switch for unicast traffic. (2:0) = 6 - 7 are invalid. (2:0) = 5 puts in the QSE in randomization mode. (2:0) = 4, N = 16. (2:0) = 3, N = 8. (2:0) = 2, N = 4. (2:0) = 1, N = 2. (2:0) = 0, N = 1. Note that this register determines whether the QSE is in **randomization** mode or **switching** mode: when bits (2:0) have the value 5 then the QSE is in randomization mode, and when bits (2:0) have a value between 4 and 0 then the QSE is in switching mode. Thus, for example, in a 3-stage switch fabric, all the QSEs in the 1st stage should have UNICAST_AGG_OUT set to 5, and all the QSEs in the 2nd and 3rd stage should have UNICAST_AGG_OUT set to values between 4 and 0. The rationale behind this encoding is that randomization mode may be viewed as switching mode with N = 32, because in randomization mode a unicast cell is "switched" to any of the 32 output ports. Note that "aggregate mode" is also called "gang mode" in other parts of this document. *Note: The multicast output gang mode (see section “9.3.24 MULTICAST_AGGREGATE_OUTPUT_AND_INPUT_MODES” on page 105) must be set to a value less than or equal to the unicast output gang mode*

9.3.26 SWITCH_FABRIC_ROW

Address: 84_h
Type: Read/Write

Format: Refer to the following table.

Field (Bits)	Description
(7:0)	R × PG where: R= the Row number in the switching fabric for this switch element. The numbering of rows starts from 0. PG = the Physical Gang of the QSE, which is defined as the number of output ports that physically connect this QSE to a chip (QSE or QRT) in the next stage. Note that PG can have a value of 1,2,4,8, or 16 if the next stage consists of QSEs, and it can have a value of 1,2, or 4 if the next stage consists of QRTs. If the value (R x PG) exceeds 8 bit, the upper bits (i.e. MSBs) should be truncated, to leave the lower 8 bits in the SWITCH_FABRIC_ROW register.

9.3.27 SWITCH_FABRIC_COLUMN

Address: 85_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
(7:0)	C + 16P where: C = the Column number in the switching fabric for this switch element. The numbering of columns starts from 0. P = the plane number if there are multiple parallel switch planes. The numbering of planes also starts from 0.

9.3.28 CELL_START_OFFSET

Address: 86_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
Not used (7)	Write with a 0 to maintain future software compatibility.
(6:0)	Offset between (external) CELL_START and Local CELL_START (NOTE: The CSTART offset must only be changed when the device is in software reset.) Legal values for this register are between 0 and 117.

9.3.29 BP_CONTROL_REGISTER

Address: 87_h
Type: Read/Write
Format: Refer to the following table.
Note: The BP_CONTROL_REGISTER is typically used for fine-tuning multicast performance. For initial system bring-up, this register may be left at the power-up default value.

Field (Bits)	Description
Not used (7:4)	Write with a 0 to maintain future software compatibility.
GLOBAL_LIMIT_2 (3)	If 1, second port threshold is off.
GLOBAL_LIMIT_1 (2)	If 1, first port threshold is off.
PER_PORT_LIMIT (1)	1 Each port allowed to have a maximum of 4 cells pending. 0 Each port allowed to have a maximum of 3 cells pending.
EARLY_BP (0)	1 Early (hence conservative) backpressure. 0 Optimal backpressure.

9.3.30 ACK_PAYLOAD

Address: 88_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
PARITY_NACK (7:4)	ACK Payload for Parity Error Cells. Reset to 8_h (Default is ONACK).

Field (Bits)	Description
CONGESTION_NACK (3:0)	ACK Payload for cells dropped due to congestion. Reset to 4_h (Default is MNACK).

9.3.31 GANG_DEAD_ACK_PAYLOAD

Address: 89_h
Type: Read/Write
Format: Refer to the following table.

Field (Bits)	Description
Not used (7:4)	Write with a 0 to maintain future software compatibility.
GANG_DEAD_NACK (3:0)	ACK Payload for cells dropped when an entire gang is disabled. A gang is defined as a set of consecutive outputs that is treated as a single output by the switch for unicast traffic (see “UNICAST_AGGREGATE_OUTPUT_MODE” on page 106). This field resets to C_h (which is interpreted by the QRT as an ACK).

9.3.32 EXTENDED_SWITCH_MODE

Address: $8A_h$
Type: Read
Format: Refer to the following table.

Field (Bits)	Description
LATCHED_CELL_START_OUT_ OF_LOCK (0)	Cleared on read. Set anytime cstart goes out of lock.

10 JTAG

10.1 JTAG Support

The QRT supports the IEEE Boundary Scan Specification as described in the IEEE 1149.1 standards. The Test Access Port (TAP) consists of the five standard pins, TRSTB, TCK, TMS, TDI and TDO, used to control the TAP controller and the boundary scan registers. The TRSTB input is the active low reset signal used to reset the TAP controller. TCK is the test clock used to sample data on input, TDI and to output data on output, TDO. The TMS input is used to direct the TAP controller through its states. The basic boundary scan architecture is shown below.

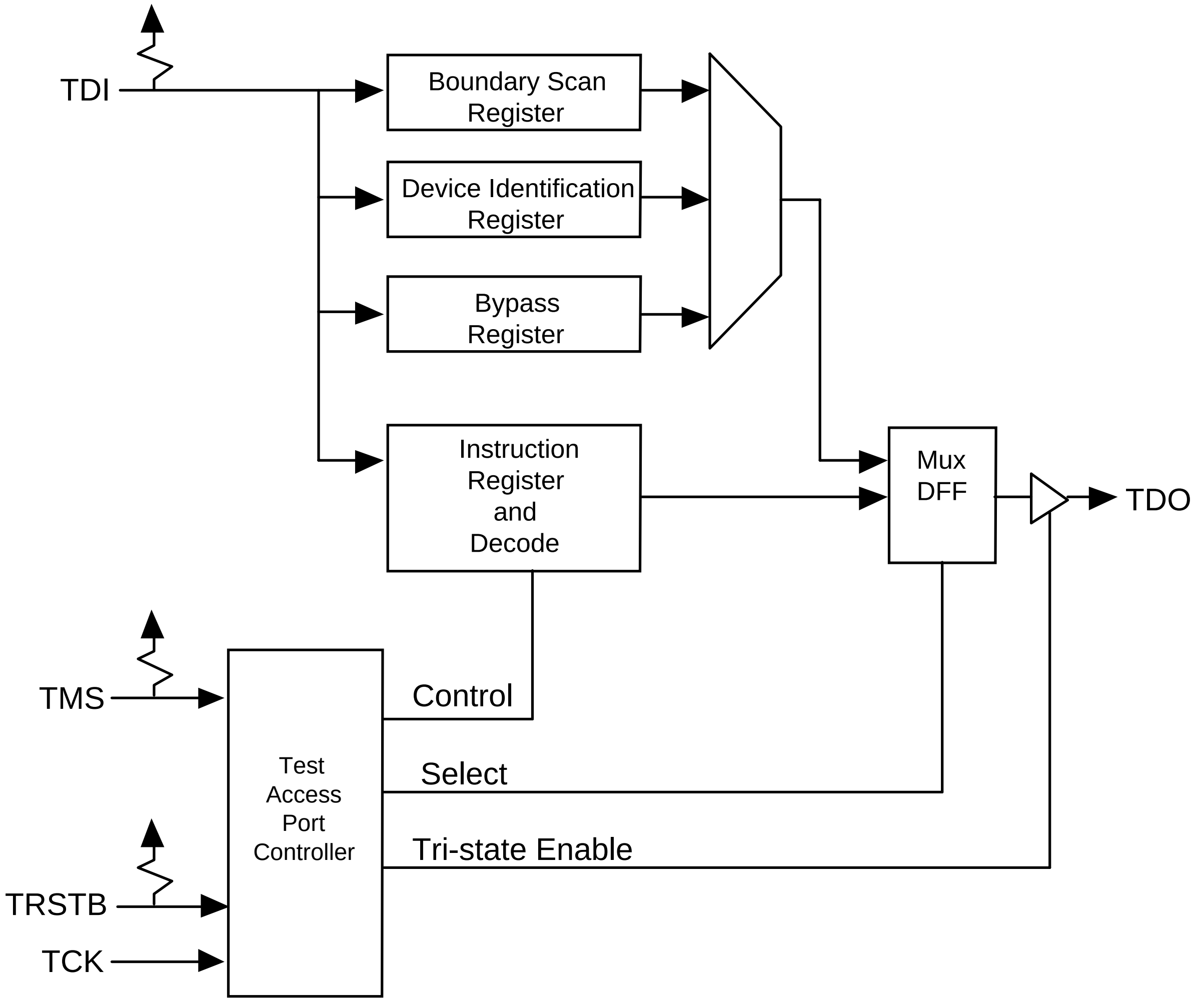


Figure 42. Boundary Scan Architecture

The boundary scan architecture consists of a TAP controller, an instruction register with instruction decode, a bypass register, a device identification register and a boundary scan register. The TAP controller interprets the TMS input and generates control signals to load the instruction and data registers. The instruction register with instruction decode block is used to select the test to be executed and/or the register to be accessed. The bypass register offers a single bit delay from primary input, TDI to primary output, TDO. The device identification register contains the device identification code.

The boundary scan register allows testing of board inter-connectivity. The boundary scan register consists of a shift register place in series with device inputs and outputs. Using the boundary scan register, all digital inputs can be sampled and shifted out on primary output, TDO. In addition, patterns can be shifted in on primary input, TDI and forced onto all digital outputs.

10.2 TAP Controller

The TAP controller is a synchronous finite state machine clocked by the rising edge of primary input, TCK. All state transitions are controlled using primary input, TMS. The finite state machine is described below.

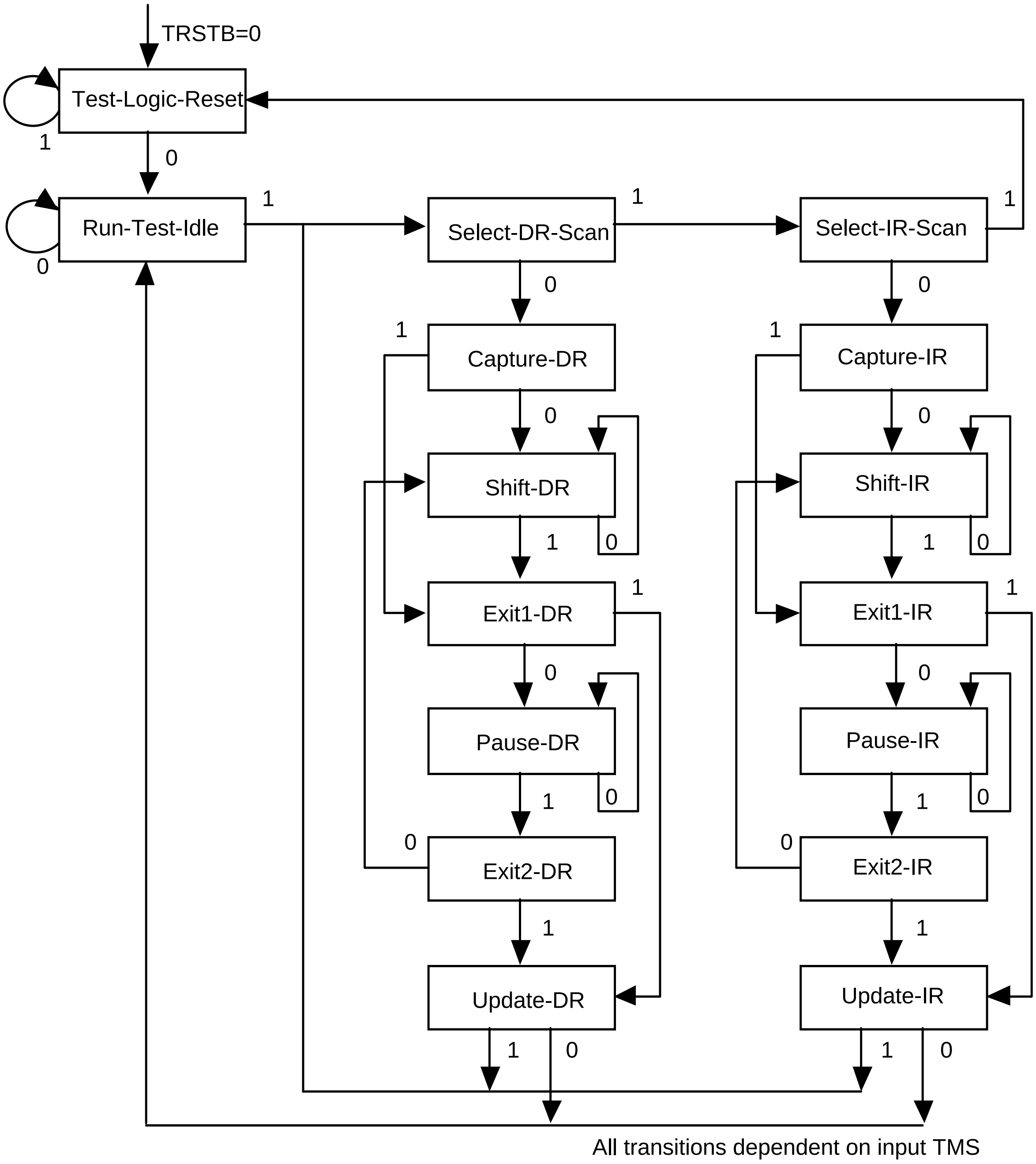


Figure 43. TAP Controller Finite State Machine

10.2.1 Test-Logic-Reset:

The test logic reset state is used to disable the TAP logic when the device is in normal mode operation. The state is entered asynchronously by asserting input, TRSTB. The state is entered synchronously regardless of the current TAP controller state by forcing input, TMS high for 5 TCK clock cycles. While in this state, the instruction register is set to the IDCODE instruction.

10.2.2 Run-Test-Idle:

The run test/idle state is used to execute tests.

10.2.3 Capture-DR:

The capture data register state is used to load parallel data into the test data registers selected by the current instruction. If the selected register does not allow parallel loads or no loading is required by the current instruction, the test register maintains its value. Loading occurs on the rising edge of TCK.

10.2.4 Shift-DR:

The shift data register state is used to shift the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

10.2.5 Update-DR:

The update data register state is used to load a test register's parallel output latch. In general, the output latches are used to control the device. For example, for the EXTEST instruction, the boundary scan test register's parallel output latches are used to control the device's outputs. The parallel output latches are updated on the falling edge of TCK.

10.2.6 Capture-IR:

The capture instruction register state is used to load the instruction register with a fixed instruction. The load occurs on the rising edge of TCK.

10.2.7 Shift-IR:

The shift instruction register state is used to shift both the instruction register and the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

10.2.8 Update-IR:

The update instruction register state is used to load a new instruction into the instruction register. The new instruction must be scanned in using the Shift-IR state. The load occurs on the falling edge of TCK.

The Pause-DR and Pause-IR states are provided to allow shifting through the test data and/or instruction registers to be momentarily paused.

The TDO output is enabled during states Shift-DR and Shift-IR. Otherwise, it is tri-stated.

10.3 Boundary Scan Instructions

The following is a description of the standard instructions. Each instruction selects an serial test data register path between input, TDI, and output, TDO.

10.3.1 BYPASS

The bypass instruction shifts data from input TDI to output TDO with one TCK clock period delay. The instruction is used to bypass the device.

10.3.2 EXTEST

The external test instruction allows testing of the interconnection to other devices. When the current instruction is the EXTEST instruction, the boundary scan register is place between input TDI and output TDO. Primary device inputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state. Primary device outputs can be controlled by loading patterns shifted in through input TDI into the boundary scan register using the Update-DR state.

10.3.3 SAMPLE

The sample instruction samples all the device inputs and outputs. For this instruction, the boundary scan register is placed between TDI and TDO. Primary device inputs and outputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state.

10.3.4 IDCODE

The identification instruction is used to connect the identification register between TDI and TDO. The device's identification code can then be shifted out using the Shift-DR state.

10.3.5 STCTEST

The single transport chain instruction is used to test out the TAP controller and the boundary scan register during production test. When this instruction is the current instruction, the boundary scan register is connected between TDI and TDO. During the Capture-DR state, the device identification code is loaded into the boundary scan register. The code can then be shifted out on output TDO using the Shift-DR state.

10.4 Boundary Scan Pin Order

Table 32. Boundary Scan Pin order

Order #	Pin #	Pin name	Pin Type
0		HIZ	output enable
1		HIZ	output enable
2		HIZ	output enable
3		HIZ	output enable
4		HIZ	output enable
5		HIZ	output enable
6		HIZ	output enable
7		HIZ	output enable
8		HIZ	output enable
9		HIZ	output enable
10	G6	SCAN_ENN	clock
11	D3	OEN	clock
12	J6	CELL_START	clock
13	C2	CELL_24_START	clock
14	K6	STAT_OUT_NDI	output3
15	K6	STAT_OUT_NDI	input
16	E2	CTRL_IN_NDO	clock

17	F3	SE_D_IN16.0	clock
18	C1	SE_D_IN16.1	clock
19	D2	SE_D_IN16.2	clock
20	H3	SE_D_IN16.3	clock
21	J5	SE_SOC_IN16	clock
22	K5	SE_D_IN17.0	clock
23	K4	SE_D_IN17.1	clock
24	J4	SE_D_IN17.2	clock
25	G2	SE_D_IN17.3	clock
26	D1	SE_SOC_IN17	clock
27	L5	SE_D_IN18.0	clock
28	J3	SE_D_IN18.1	clock
29	M6	SE_D_IN18.2	clock
30	N6	SE_D_IN18.3	clock
31	F1	SE_SOC_IN18	clock
32	K3	SE_D_IN19.0	clock
33	H2	SE_D_IN19.1	clock
34	M5	SE_D_IN19.2	clock
35	L4	SE_D_IN19.3	clock
36	H1	SE_SOC_IN19	clock
37	M3	SE_D_IN20.0	clock
38	J2	SE_D_IN20.1	clock
39	N4	SE_D_IN20.2	clock
40	G1	SE_D_IN20.3	clock
41	K1	SE_SOC_IN20	clock
42	L2	SE_D_IN21.0	clock
43	N3	SE_D_IN21.1	clock
44	P6	SE_D_IN21.2	clock
45	N5	SE_D_IN21.3	clock

46	R6	SE_SOC_IN21	clock
47	M2	SE_D_IN22.0	clock
48	L1	SE_D_IN22.1	clock
49	P5	SE_D_IN22.2	clock
50	N2	SE_D_IN22.3	clock
51	R5	SE_SOC_IN22	clock
52	P4	SE_D_IN23.0	clock
53	R4	SE_D_IN23.1	clock
54	P3	SE_D_IN23.2	clock
55	P1	SE_D_IN23.3	clock
56	M1	SE_SOC_IN23	clock
57	R2	SE_D_IN24.0	clock
58	T2	SE_D_IN24.1	clock
59	R1	SE_D_IN24.2	clock
60	U1	SE_D_IN24.3	clock
61	T1	SE_SOC_IN24	clock
62	T4	SE_D_IN25.0	clock
63	W1	SE_D_IN25.1	clock
64	T5	SE_D_IN25.2	clock
65	V2	SE_D_IN25.3	clock
66	U3	SE_SOC_IN25	clock
67	Y1	SE_D_IN26.0	clock
68	U4	SE_D_IN26.1	clock
69	T6	SE_D_IN26.2	clock
70	U5	SE_D_IN26.3	clock
71	U6	SE_SOC_IN26	clock
72	V3	SE_D_IN27.0	clock
73	W2	SE_D_IN27.1	clock
74	AA1	SE_D_IN27.2	clock

75	V5	SE_D_IN27.3	clock
76	V4	SE_SOC_IN27	clock
77	Y2	SE_D_IN28.0	clock
78	W3	SE_D_IN28.1	clock
79	AC1	SE_D_IN28.2	clock
80	AD1	SE_D_IN28.3	clock
81	W5	SE_SOC_IN28	clock
82	AB2	SE_D_IN29.0	clock
83	AA3	SE_D_IN29.1	clock
84	Y4	SE_D_IN29.2	clock
85	V6	SE_D_IN29.3	clock
86	W6	SE_SOC_IN29	clock
87	AC2	SE_D_IN30.0	clock
88	Y5	SE_D_IN30.1	clock
89	AE1	SE_D_IN30.2	clock
90	AD2	SE_D_IN30.3	clock
91	AB3	SE_SOC_IN30	clock
92	AA4	SE_D_IN31.0	clock
93	AA5	SE_D_IN31.1	clock
94	AG1	SE_D_IN31.2	clock
95	AC3	SE_D_IN31.3	clock
96	AB4	SE_SOC_IN31	clock
97	AH1	RAM_ADD.17	output3
98	AB5	BP_ACK_OUT.0	output3
99	AF2	BP_ACK_OUT.1	output3
100	AA6	BP_ACK_OUT.2	output3
101	AG2	BP_ACK_OUT.3	output3
102	AB6	BP_ACK_OUT.4	output3
103	AE3	BP_ACK_OUT.5	output3

104	AC5	BP_ACK_OUT.6	output3
105	AH2	BP_ACK_OUT.7	output3
106	AD4	BP_ACK_OUT.8	output3
107	AD6	BP_ACK_OUT.9	output3
108	AG3	BP_ACK_OUT.10	output3
109	AE4	BP_ACK_OUT.11	output3
110	AD5	BP_ACK_OUT.12	output3
111	AE5	BP_ACK_OUT.13	output3
112	AF4	BP_ACK_OUT.14	output3
113	AJ1	BP_ACK_OUT.15	output3
114	AK2	BP_ACK_OUT.16	output3
115	AG5	BP_ACK_OUT.17	output3
116	AF6	BP_ACK_OUT.18	output3
117	AF7	BP_ACK_OUT.19	output3
118	AG6	BP_ACK_OUT.20	output3
119	AH4	BP_ACK_OUT.21	output3
120	AE7	BP_ACK_OUT.22	output3
121	AG7	BP_ACK_OUT.23	output3
122	AJ3	BP_ACK_OUT.24	output3
123	AF8	BP_ACK_OUT.25	output3
124	AH6	BP_ACK_OUT.26	output3
125	AE9	BP_ACK_OUT.27	output3
126	AJ4	BP_ACK_OUT.28	output3
127	AE10	BP_ACK_OUT.29	output3
128	AJ5	BP_ACK_OUT.30	output3
129	AF9	BP_ACK_OUT.31	output3
130	AK3	ADD.0	clock
131	AG9	ADD.1	clock
132	AH8	ADD.2	clock

133	AK4	ADD.3	clock
134	AF10	ADD.4	clock
135	AG10	ADD.5	clock
136	AH9	ADD.6	clock
137	AJ7	ADD.7	clock
138	AK6	DATA.0	output3
139	AK6	DATA.0	input
140	AF11	DATA.1	output3
141	AF11	DATA.1	input
142	AJ8	DATA.2	output3
143	AJ8	DATA.2	input
144	AE12	DATA.3	output3
145	AE12	DATA.3	input
146	AE13	DATA.4	output3
147	AE13	DATA.4	input
148	AG11	DATA.5	output3
149	AG11	DATA.5	input
150	AH10	DATA.6	output3
151	AH10	DATA.6	input
152	AJ9	DATA.7	output3
153	AJ9	DATA.7	input
154	AF12	RAM_ADD.18	output3
155	AK7	CSN	clock
156	AK8	RDN	clock
157	AH12	WRN	clock
158	AJ11	ACKN	output3
159	AG13	INTRN	output3
160	AF13	RESET	clock
161	AK10	SE_D_OUT31.0	output3

162	AJ12	SE_D_OUT31.1	output3
163	AH13	SE_D_OUT31.2	output3
164	AE14	SE_D_OUT31.3	output3
165	AF14	SE_D_OUT30.0	output3
166	AE15	SE_D_OUT30.1	output3
167	AG14	SE_D_OUT30.2	output3
168	AK11	SE_D_OUT30.3	output3
169	AH14	SE_D_OUT29.0	output3
170	AJ13	SE_D_OUT29.1	output3
171	AF15	SE_D_OUT29.2	output3
172	AK12	SE_D_OUT29.3	output3
173	AG15	SE_D_OUT28.0	output3
174	AK15	SE_D_OUT28.1	output3
175	AK14	SE_D_OUT28.2	output3
176	AK16	SE_D_OUT28.3	output3
177	AJ15	SE_SOC_OUT7	output3
178	AK19	SE_D_OUT27.0	output3
179	AK17	SE_D_OUT27.1	output3
180	AH17	SE_D_OUT27.2	output3
181	AG16	SE_D_OUT27.3	output3
182	AG17	SE_D_OUT26.0	output3
183	AF16	SE_D_OUT26.1	output3
184	AJ18	SE_D_OUT26.2	output3
185	AF17	SE_D_OUT26.3	output3
186	AK20	SE_D_OUT25.0	output3
187	AJ19	SE_D_OUT25.1	output3
188	AE16	SE_D_OUT25.2	output3
189	AF18	SE_D_OUT25.3	output3
190	AE17	SE_D_OUT24.0	output3

191	AH18	SE_D_OUT24.1	output3
192	AJ20	SE_D_OUT24.2	output3
193	AK21	SE_D_OUT24.3	output3
194	AK24	SE_SOC_OUT6	output3
195	AG18	SE_D_OUT23.0	output3
196	AJ22	SE_D_OUT23.1	output3
197	AH19	SE_D_OUT23.2	output3
198	AK23	SE_D_OUT23.3	output3
199	AG20	SE_D_OUT22.0	output3
200	AF19	SE_D_OUT22.1	output3
201	AJ23	SE_D_OUT22.2	output3
202	AH21	SE_D_OUT22.3	output3
203	AK25	SE_D_OUT21.0	output3
204	AE18	SE_D_OUT21.1	output3
205	AE19	SE_D_OUT21.2	output3
206	AH22	SE_D_OUT21.3	output3
207	AF20	SE_D_OUT20.0	output3
208	AK27	SE_D_OUT20.1	output3
209	AJ24	SE_D_OUT20.2	output3
210	AG22	SE_D_OUT20.3	output3
211	AG21	SE_SOC_OUT5	output3
212	AF21	SE_D_OUT19.0	output3
213	AF22	SE_D_OUT19.1	output3
214	AH23	SE_D_OUT19.2	output3
215	AJ27	SE_D_OUT19.3	output3
216	AK28	SE_D_OUT18.0	output3
217	AH25	SE_D_OUT18.1	output3
218	AJ26	SE_D_OUT18.2	output3
219	AE21	SE_D_OUT18.3	output3

220	AJ28	SE_D_OUT17.0	output3
221	AE22	SE_D_OUT17.1	output3
222	AH27	SE_D_OUT17.2	output3
223	AF23	SE_D_OUT17.3	output3
224	AF24	SE_D_OUT16.0	output3
225	AG24	SE_D_OUT16.1	output3
226	AE24	SE_D_OUT16.2	output3
227	AG26	SE_D_OUT16.3	output3
228	AG25	SE_SOC_OUT4	output3
229	AF25	SE_CLK_BYPASS	clock
230	AK29	SE_D_OUT15.0	output3
231	AE26	SE_D_OUT15.1	output3
232	AD26	SE_D_OUT15.2	output3
233	AE27	SE_D_OUT15.3	output3
234	AG28	SE_D_OUT14.0	output3
235	AD25	SE_D_OUT14.1	output3
236	AD27	SE_D_OUT14.2	output3
237	AH29	SE_D_OUT14.3	output3
238	AC26	SE_D_OUT13.0	output3
239	AE28	SE_D_OUT13.1	output3
240	AB25	SE_D_OUT13.2	output3
241	AG29	SE_D_OUT13.3	output3
242	AA25	SE_D_OUT12.0	output3
243	AF29	SE_D_OUT12.1	output3
244	AB26	SE_D_OUT12.2	output3
245	AH30	SE_D_OUT12.3	output3
246	AB27	SE_SOC_OUT3	output3
247	AC28	SE_D_OUT11.0	output3
248	AG30	SE_D_OUT11.1	output3

249	AA26	SE_D_OUT11.2	output3
250	AA27	SE_D_OUT11.3	output3
251	AB28	SE_D_OUT10.0	output3
252	AD29	SE_D_OUT10.1	output3
253	AE30	SE_D_OUT10.2	output3
254	Y26	SE_D_OUT10.3	output3
255	AC29	SE_D_OUT09.0	output3
256	W25	SE_D_OUT09.1	output3
257	V25	SE_D_OUT09.2	output3
258	Y27	SE_D_OUT09.3	output3
259	AA28	SE_D_OUT08.0	output3
260	AB29	SE_D_OUT08.1	output3
261	W26	SE_D_OUT08.2	output3
262	AD30	SE_D_OUT08.3	output3
263	AC30	SE_SOC_OUT2	output3
264	W28	SE_D_OUT07.0	output3
265	Y29	SE_D_OUT07.1	output3
266	V27	SE_D_OUT07.2	output3
267	V26	SE_D_OUT07.3	output3
268	AA30	SE_D_OUT06.0	output3
269	W29	SE_D_OUT06.1	output3
270	V28	SE_D_OUT06.2	output3
271	U25	SE_D_OUT06.3	output3
272	U26	SE_D_OUT05.0	output3
273	T25	SE_D_OUT05.1	output3
274	U27	SE_D_OUT05.2	output3
275	Y30	SE_D_OUT05.3	output3
276	U28	SE_D_OUT04.0	output3
277	V29	SE_D_OUT04.1	output3

278	T26	SE_D_OUT04.2	output3
279	W30	SE_D_OUT04.3	output3
280	T27	SE_SOC_OUT1	output3
281	T30	SE_D_OUT03.0	output3
282	U30	SE_D_OUT03.1	output3
283	R30	SE_D_OUT03.2	output3
284	T29	SE_D_OUT03.3	output3
285	R29	SE_D_OUT02.0	output3
286	M30	SE_D_OUT02.1	output3
287	P30	SE_D_OUT02.2	output3
288	P28	SE_D_OUT02.3	output3
289	R27	SE_D_OUT01.0	output3
290	P27	SE_D_OUT01.1	output3
291	R26	SE_D_OUT01.2	output3
292	N29	SE_D_OUT01.3	output3
293	P26	SE_D_OUT00.0	output3
294	L30	SE_D_OUT00.1	output3
295	M29	SE_D_OUT00.2	output3
296	R25	SE_D_OUT00.3	output3
297	N26	SE_SOC_OUT0	output3
298	P25	PLL_BYPASS_N	clock
299	L29	BP_ACK_IN.0	clock
300	K30	BP_ACK_IN.1	clock
301	G30	BP_ACK_IN.2	clock
302	N27	BP_ACK_IN.3	clock
303	J29	BP_ACK_IN.4	clock
304	M28	BP_ACK_IN.5	clock
305	H30	BP_ACK_IN.6	clock
306	L27	BP_ACK_IN.7	clock

307	M26	BP_ACK_IN.8	clock
308	H29	BP_ACK_IN.9	clock
309	K28	BP_ACK_IN.10	clock
310	F30	BP_ACK_IN.11	clock
311	N25	BP_ACK_IN.12	clock
312	M25	BP_ACK_IN.13	clock
313	J28	BP_ACK_IN.14	clock
314	L26	BP_ACK_IN.15	clock
315	D30	BP_ACK_IN.16	clock
316	G29	BP_ACK_IN.17	clock
317	J27	BP_ACK_IN.18	clock
318	K27	BP_ACK_IN.19	clock
319	K26	BP_ACK_IN.20	clock
320	J26	BP_ACK_IN.21	clock
321	H28	BP_ACK_IN.22	clock
322	D29	BP_ACK_IN.23	clock
323	C30	BP_ACK_IN.24	clock
324	F28	BP_ACK_IN.25	clock
325	E29	BP_ACK_IN.26	clock
326	K25	BP_ACK_IN.27	clock
327	C29	BP_ACK_IN.28	clock
328	J25	BP_ACK_IN.29	clock
329	D28	BP_ACK_IN.30	clock
330	H26	BP_ACK_IN.31	clock
331	G26	RAM_ADD.0	output3
332	G27	RAM_ADD.1	output3
333	G25	RAM_ADD.2	output3
334	E27	RAM_ADD.3	output3
335	F27	RAM_ADD.4	output3

336	F26	RAM_ADD.5	output3
337	B30	RAM_ADD.6	output3
338	A29	RAM_ADD.7	output3
339	E25	RAM_ADD.8	output3
340	D25	RAM_ADD.9	output3
341	D26	RAM_ADD.10	output3
342	F24	RAM_ADD.11	output3
343	D24	RAM_ADD.12	output3
344	E24	RAM_ADD.13	output3
345	E23	RAM_ADD.14	output3
346	C27	RAM_ADD.15	output3
347	F22	RAM_DATA.0	output3
348	F22	RAM_DATA.0	input
349	B28	RAM_DATA.1	output3
350	B28	RAM_DATA.1	input
351	F21	RAM_DATA.2	output3
352	F21	RAM_DATA.2	input
353	B26	RAM_DATA.3	output3
354	B26	RAM_DATA.3	input
355	C25	RAM_DATA.4	output3
356	C25	RAM_DATA.4	input
357	A28	RAM_DATA.5	output3
358	A28	RAM_DATA.5	input
359	B27	RAM_DATA.6	output3
360	B27	RAM_DATA.6	input
361	C23	RAM_DATA.7	output3
362	C23	RAM_DATA.7	input
363	E22	RAM_DATA.8	output3
364	E22	RAM_DATA.8	input

365	E21	RAM_DATA.9	output3
366	E21	RAM_DATA.9	input
367	D21	RAM_DATA.10	output3
368	D21	RAM_DATA.10	input
369	D22	RAM_DATA.11	output3
370	D22	RAM_DATA.11	input
371	B24	RAM_DATA.12	output3
372	B24	RAM_DATA.12	input
373	A27	RAM_DATA.13	output3
374	A27	RAM_DATA.13	input
375	E20	RAM_DATA.14	output3
376	E20	RAM_DATA.14	input
377	C22	RAM_DATA.15	output3
378	C22	RAM_DATA.15	input
379	F19	RAM_ADD.16	output3
380	F18	RAM_PARITY	output3
381	F18	RAM_PARITY	input
382	A25	RAM_WRN	output3
383	C21	RAM_OEN	output3
384	B23	RAM_CLK	output3
385	E19	SE_D_IN00.0	clock
386	D20	SE_D_IN00.1	clock
387	A23	SE_D_IN00.2	clock
388	C19	SE_D_IN00.3	clock
389	B22	SE_SOC_IN00	clock
390	D18	SE_D_IN01.0	clock
391	A24	SE_D_IN01.1	clock
392	A21	SE_D_IN01.2	clock
393	B20	SE_D_IN01.3	clock

394	C18	SE_SOC_IN01	clock
395	F17	SE_D_IN02.0	clock
396	E18	SE_D_IN02.1	clock
397	F16	SE_D_IN02.2	clock
398	B19	SE_D_IN02.3	clock
399	A20	SE_SOC_IN02	clock
400	E17	SE_D_IN03.0	clock
401	B18	SE_D_IN03.1	clock
402	E16	SE_D_IN03.2	clock
403	D17	SE_D_IN03.3	clock
404	D16	SE_SOC_IN03	clock
405	C17	SE_D_IN04.0	clock
406	A17	SE_D_IN04.1	clock
407	A19	SE_D_IN04.2	clock
408	B16	SE_D_IN04.3	clock
409	B15	SE_SOC_IN04	clock
410	A16	SE_D_IN05.0	clock
411	A14	SE_D_IN05.1	clock
412	A15	SE_D_IN05.2	clock
413	D15	SE_D_IN05.3	clock
414	A12	SE_SOC_IN05	clock
415	E15	SE_D_IN06.0	clock
416	B13	SE_D_IN06.1	clock
417	C14	SE_D_IN06.2	clock
418	A11	SE_D_IN06.3	clock
419	D14	SE_SOC_IN06	clock
420	F15	SE_D_IN07.0	clock
421	E14	SE_D_IN07.1	clock
422	F14	SE_D_IN07.2	clock

423	C13	SE_D_IN07.3	clock
424	B12	SE_SOC_IN07	clock
425	A10	SE_D_IN08.0	clock
426	E13	SE_D_IN08.1	clock
427	D13	SE_D_IN08.2	clock
428	B11	SE_D_IN08.3	clock
429	C12	SE_SOC_IN08	clock
430	A8	SE_D_IN09.0	clock
431	A7	SE_D_IN09.1	clock
432	E12	SE_D_IN09.2	clock
433	B9	SE_D_IN09.3	clock
434	C10	SE_SOC_IN09	clock
435	D11	SE_D_IN10.0	clock
436	F13	SE_D_IN10.1	clock
437	F12	SE_D_IN10.2	clock
438	B8	SE_D_IN10.3	clock
439	E11	SE_SOC_IN10	clock
440	A6	SE_D_IN11.0	clock
441	B7	SE_D_IN11.1	clock
442	C9	SE_D_IN11.2	clock
443	D10	SE_D_IN11.3	clock
444	E10	SE_SOC_IN11	clock
445	A4	SE_D_IN12.0	clock
446	C8	SE_D_IN12.1	clock
447	D9	SE_D_IN12.2	clock
448	A3	SE_D_IN12.3	clock
449	E9	SE_SOC_IN12	clock
450	B5	SE_D_IN13.0	clock
451	F10	SE_D_IN13.1	clock

452	B4	SE_D_IN13.2	clock
453	F9	SE_D_IN13.3	clock
454	C6	SE_SOC_IN13	clock
455	E8	SE_D_IN14.0	clock
456	B3	SE_D_IN14.1	clock
457	D7	SE_D_IN14.2	clock
458	F7	SE_D_IN14.3	clock
459	C4	SE_SOC_IN14	clock
460	D6	SE_D_IN15.0	clock
461	E7	SE_D_IN15.1	clock
462	E6	SE_D_IN15.2	clock
463	D5	SE_D_IN15.3	clock
464	A2	SE_SOC_IN15	clock

APPENDIX A NOMENCLATURE

A.1 Definitions

Transmit signals: all signals related to processing the data heading towards the optical/electrical layer.
Receive signals: all signals related to processing the data heading towards the ATM layer.

A.2 Numbers

- Hexadecimal numbers are followed by the suffix “$_h$”, for example: 1_h, $2C_h$.
- Binary numbers are followed by the suffix “$_b$”, for example: 00_b.
- Decimal numbers appear without suffixes.

A.3 Glossary of Abbreviations

Table 33. Standard Abbreviations

Abbreviation	Description
ACK	Acknowledgment
ATM	Asynchronous Transfer Mode
CLP	Cell Loss Priority
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit
EPBGA	Enhanced Plastic Ball Grid Array
IRT	Input half of QRT
JTAG	Joint Test Access Group
MB	Mark Bit
MC	Multicast
MGI	Multicast Group Index
MGV	Multicast Group Vector
MNACK	Mid Switch Negative ACKnowledgment
MPV	Multicast Port Vector
NACK	Negative ACKnowledgment
ONACK	Output Negative ACKnowledgment
ORT	Output half of QRT
PG	Physical Gang
PHY	Physical

Table 33. Standard Abbreviations (Continued)

Abbreviation	Description
PIF	General-purpose microprocessor interface
PLL	Phase Locked Loop
PTI	Payload Type Indicator
QRT	PMC’s ATM traffic management chip (PM73487)
QSE	PMC’s ATM switch fabric chip (PM73488)
RAM	Random Access Memory
SF	Speedup Factor
SOC	Start-Of-Cell
SP	Spare Bit
SRAM	Static Random Access Memory
SSRAM	Synchronous Static Random Access Memory
UTOPIA	Universal Test and Operations PHY Interface for ATM
VC	Virtual Channel
VCI	Virtual Channel Identifier

APPENDIX B REFERENCES

- ATM Forum, *ATM User-Network Interface Specification*, V3.0, September 10, 1993.
- IEEE 1149.1, *Standard Test Access Port and Boundary Scan Architecture*, May 21, 1990.
- ITU (CCITT) Recommendation I.432, *B-ISDN User-Network Interface - Physical Interface Specification*, June 1990.
- UTOPIA, *An ATM PHY Data Path Interface, Level 1*, V2.01, February, 1994.

ORDERING INFORMATION

Table 34 lists the ordering information.

Table 34. Ordering Information

Part Number	Description
PM73488-PI	596-pin Enhanced Plastic Ball Grid Array (EPBGA) package

NOTES

CONTACTING PMC-SIERRA, INC.

PMC-Sierra, Inc.
105-8555 Baxter Place
Burnaby B.C. Canada V5A 4V7

Tel: (604) 415-6000
Fax: (604) 415-6200

Document Information:	document@pmc-sierra.com
Corporate Information:	info@pmc-sierra.com
Application Information:	apps@pmc-sierra.com (604) 415-4533
Web Site:	http://www.pmc-sierra.com

None of the information contained in this document constitutes an express or implied warranty by PMC-Sierra, Inc. as to the sufficiency, fitness, or suitability for a particular purpose of any such information of the fitness or suitability for a particular purpose, merchantability, performance, compatibility with other parts or systems, of any of the products of PMC-Sierra, Inc., or any portion thereof, referred to in this document. PMC-Sierra, Inc. expressly disclaims all representations and warranties of any kind regarding the contents or use of the information, including, but not limited to, express and implied warranties of accuracy, completeness, merchantability, fitness for a particular use, or non-infringement.

In no event will PMC-Sierra, Inc. be liable for any direct, indirect, special, incidental or consequential damages, including, but not limited to, lost profits, lost business or lost data resulting from any use or reliance upon the information, whether or not PMC-Sierra, Inc. has been advised of the possibility of such damage.

PMC-980616 (R3) ref PMC-981002 (R2) Issue date: June 1999