MVTX2802

Managed 4-Port 1000 Mbps Ethernet Switch

Data Sheet

February 2003

Features

- 4 Gigabit Ports with GMII and PCS interface
  - Gigabit Port can also support 100/10 Mbps MII interface
  - Provide Hot plug support for GMII/PCS module
- High Performance Layer 2 Packet Forwarding (11.904M packets per second) and Filtering at Full-Wire Speed
- Maximum throughput is 4 Gbps non-blocking
- Centralized shared-memory architecture
- Consists of two Memory Domains at 133 MHz
  - Frame Buffer Domain: One bank of ZBT-SRAM with 1M/2MB total
  - Switch Database Domain with 256K/512K SRAM
- Up to 64K MAC addresses to provide large node aggregation in wiring closet switches
- Provides Port based and ID Tagged VLAN (IEEE802.1Q) up to 4K VLAN
- Support IP Multicast with IGMP snooping up to 64K groups.

Traffic Classification

- Classify traffic into 8 transmission priorities per port

Ordering Information

MVTX2802AG 596 Pin HSBGA

-40°C to +85°C

- Supports Delay bounded, Strict Priority, and WFQ
- Provides 2 level dropping precedence with WRED mechanism
  - User controlled thresholds for WRED
- Classification based on layer 2, 3 markings
  - VLAN Priority field in VLAN tagged frame
  - DS/TOS field in IP packet
- The precedence of above two classifications can be programmable

QoS Support

- Supports IEEE 802.1p/Q Quality of Service with 8 Priority
- Buffer Management: reserve buffers on per class and per port basis

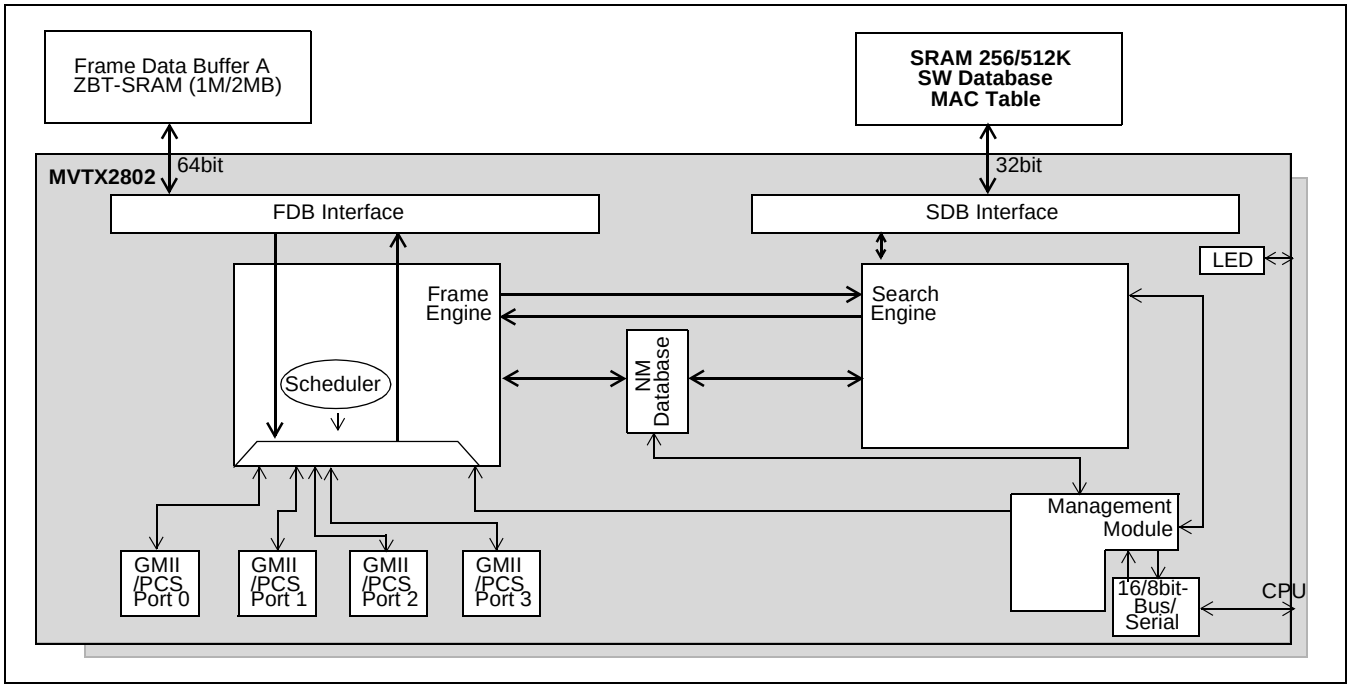


Figure 1 - MVTX2802AG Functional Block Diagram

- Port-based Priority: VLAN Priority with Tagged frame can be overwritten by the priority of PVID
- QoS features can be configured on a per port basis
- Packet Filtering and Port Security
- Static addressing filtering for source and/or destination MAC address
- Static learned MAC addresses will not be aged out
- Secure mode per port: Prevent learning for port in a secure mode
- Support per MAC per Port filtering
- Full Duplex Ethernet IEEE 802.3x Flow Control
- Provides Ethernet Multicast and Broadcast Control
- 4 Port Trunking groups, 4 ports per group (Trunking can be based on source MAC and/or destination MAC and source port)
- LED signals provided by a serial or parallel interface
- CPU interface supports 16/8-bit CPU bus in managed mode and a synchronous Serial Interface and I^2C interface in unmanaged mode.
- SNMP/RMON support with CPU
- Built-in MIB counter
- Spanning tree with CPU
- Multiple Spanning trees (Per Spanning Tree Per VLAN)
- Hardware auto-negotiation through serial management interface (MDIO) for Gigabit Ethernet ports, supports 10/100/1000 Mbps
- BIST for internal and external SRAM-ZBT
- I^2C EEPROM or synchronous serial port for configuration
- Packaged in 596-pin BGA

Description

The MVTX2800 family is a group of 1000 Mbps non-blocking Ethernet switch chips with on-chip address memory. A single chip provides a maximum of eight 1000 Mbps ports and a dedicated CPU interface with a 16/8-bit bus for managed and unmanaged switch applications. The MVTX2800 family consists of the following four products:

- MVTX2804 8 Gigabit ports Managed
- MVTX2803 8 Gigabit ports Unmanaged
- MVTX2802 4 Gigabit ports Managed
- MVTX2801 4 Gigabit ports Unmanaged

The MVTX2802AG supports up to 64K MAC addresses to aggregate traffic from multiple wiring closet stacks. The centralized shared-memory architecture allows a very high performance packet-forwarding rate of 5.952M packets per second at full wire speed. The chip is optimized to provide a low-cost, high performance workgroup, and wiring closet, layer 2 switching solution with 4 Gigabit Ethernet ports.

One Frame Buffer Memory domains utilize cost effective, high–performance ZBT-SRAM with aggregated bandwidth of 8.5Gbps to support full wire speed on all external ports simultaneously.

With Strict priority, Delay Bounded, and WRR transmission scheduling, plus WRED memory congestion scheme, the chip provides powerful QoS functions for convergent network multimedia and mission-critical applications. The chip provides 8 transmission priorities and 2 level drop precedence. Traffic is assigned its transmission priority and dropping precedence based on the frame VLAN Tag priority or DS/TOS fields in IP packets.

IP multicast snooping provides up to 64k simultaneous IP Multicast groups. With 4K IEEE 802.1Q VLANs, the MVTX2802AG provides the ability to logically group users to control multicast traffic.

The MVTX2802AG supports port trunking/load sharing on the 1000 Mbps ports with fail-over capability. The port trunking/load sharing can be used to group ports between interlinked switches to increase the effective network bandwidth.

In full-duplex mode, IEEE 802.3x flow control is provided. The Physical Coding Sublayer (PCS) is integrated on-chip to provide a direct 10-bit GMII interface, or the PCS can be bypassed to provide an interface to existing fiber-based Gigabit Ethernet transceivers.

Statistical information for Etherstat SNMP and Remote Monitoring Management Information Base (RMON MIB) are collected independently for each of the four ports. Access to these statistical counter/registers is provided via the CPU interface. SNMP Management frames can be received and transmitted via the CPU interface, creating a complete network management solution.

The MVTX2802AG is fabricated using 0.25µm technology. Inputs, however, are 3.3V tolerant and the outputs are capable of directly interfacing to LVTTL levels. The MVTX2802AG is packaged in a 596-pin Ball Grid Array package.

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

List of Figures

List of Tables

1.0 Block Functionality

1.1 Frame Data Buffer (FDB) Interfaces

The FDB interface supports pipelined ZBT-SRAM 64-bit wide memory at 133 MHz. At 133 MHz, the aggregate memory bandwidth is 8.5 Gbps, which is enough to support 4 Gigabit ports at full wire speed switching. A patent pending scheme is used to access the FDB memory. Each slot has one tick to read or write 8 bytes.

1.2 Switch Database (SDB) Interface

A pipelined synchronous burst SRAM (SBRAM) memory is used to store the switch database information including MAC Table, VLAN Table and IP Multicast Table. Search Engine accesses the switch database via SDB interface. The SDB memory has 32-bit wide bus at 133 MHz.

1.3 GMII/PCS MAC Module (GMAC)

The GMII/PCS Media Access Control (GMAC) module provides the necessary buffers and control interface between the Frame Engine (FE) and the external physical device (PHY). The MVTX2802AG has two interfaces, GMII or PCS. The GMAC of the MVTX2802AG meets the IEEE 802.3z specification and supports the MII/GMII and PCS interfaces. It is able to operate in 10M/100M/1G in Full Duplex mode with a flow control mechanism. It has the options to insert Source Address/CRC/VLAN ID to each frame. The GMII/PCS Module also supports hot plug detection.

1.4 CPU Interface Module

One extra port is dedicated to the CPU via the CPU interface module. The CPU interface utilizes a 16/8-bit bus in managed mode. It also supports a serial and an I^2C interface, which provides an easy way to configure the system if unmanaged.

1.5 Management Module

The CPU can send a control frame to access or configure the internal network management database. The Management Module decodes the control frame and executes the functions requested by the CPU.

1.6 Frame Engine

The main function of the frame engine is to forward a frame to its proper destination port or ports. When a frame arrives, the frame engine parses the frame header (64 bytes) and formulates a switching request which is sent to the search engine to resolve the destination port. The arriving frame is moved to the FDB. After receiving a switch response from the search engine, the frame engine performs transmission scheduling based on the frame's priority. The frame engine forwards the frame to the MAC module when the frame is ready to be sent.

1.7 Search Engine

The Search Engine resolves the frame's destination port or ports according to the destination MAC address (L2) or IP multicast address (IP multicast packet) by searching the database. It also performs MAC learning, priority assignment and trunking functions.

1.8 LED Interface

The LED interface can be operated in a serial mode or a parallel mode. In the serial mode, the LED interface uses 3 pins for carrying 4 port status signals. In the parallel mode, the interface can drive LEDs by 8 status pins. The LED port is shared with bootstrap pins. In order to avoid mis-reading, a buffer must be used to isolate the LED circuitry from the bootstrap pins during bootstrap cycle (the bootstraps are sampled at the rising edge of the #Reset).

1.9 Internal Memory

Several internal tables are required and are described as follows:

- Frame Control Block (FCB) - Each FCB entry contains the control information of the associated frame stored in the FDB, e.g. frame size, read/write pointer, transmission priority, etc.
- Network Management (NM) Database - The NM database contains the information in the statistics counters and MIB.
- MCT Link Table - The MCT Link Table stores the linked list of MCT entries that have collisions in the external MAC Table.
- VLAN Port Aging Table **-** This table provides the aging status of VLAN Port association status. Search Engine maintains this table and informs the CPU when the entry is ready to age out.

2.0 System Configuration

2.1 Management and Configuration

Two modes are supported in the MVTX2802AG: managed and unmanaged. In managed mode, the MVTX2802AG uses an 8-or 16-bit CPU interface very similar to the Industry Standard Architecture (ISA) specification. In unmanaged mode, the MVTX2802AG has no CPU but can be configured by EEPROM using an I^2C interface at bootup, or via a synchronous serial interface otherwise.

2.2 Managed Mode

In managed mode, the MVTX2802AG uses an 8-or 16-bit CPU interface very similar to the ISA bus. The MVTX2802AG CPU interface provides for easy and effective management of the switching system. The figure below provides an overview of the CPU interface.

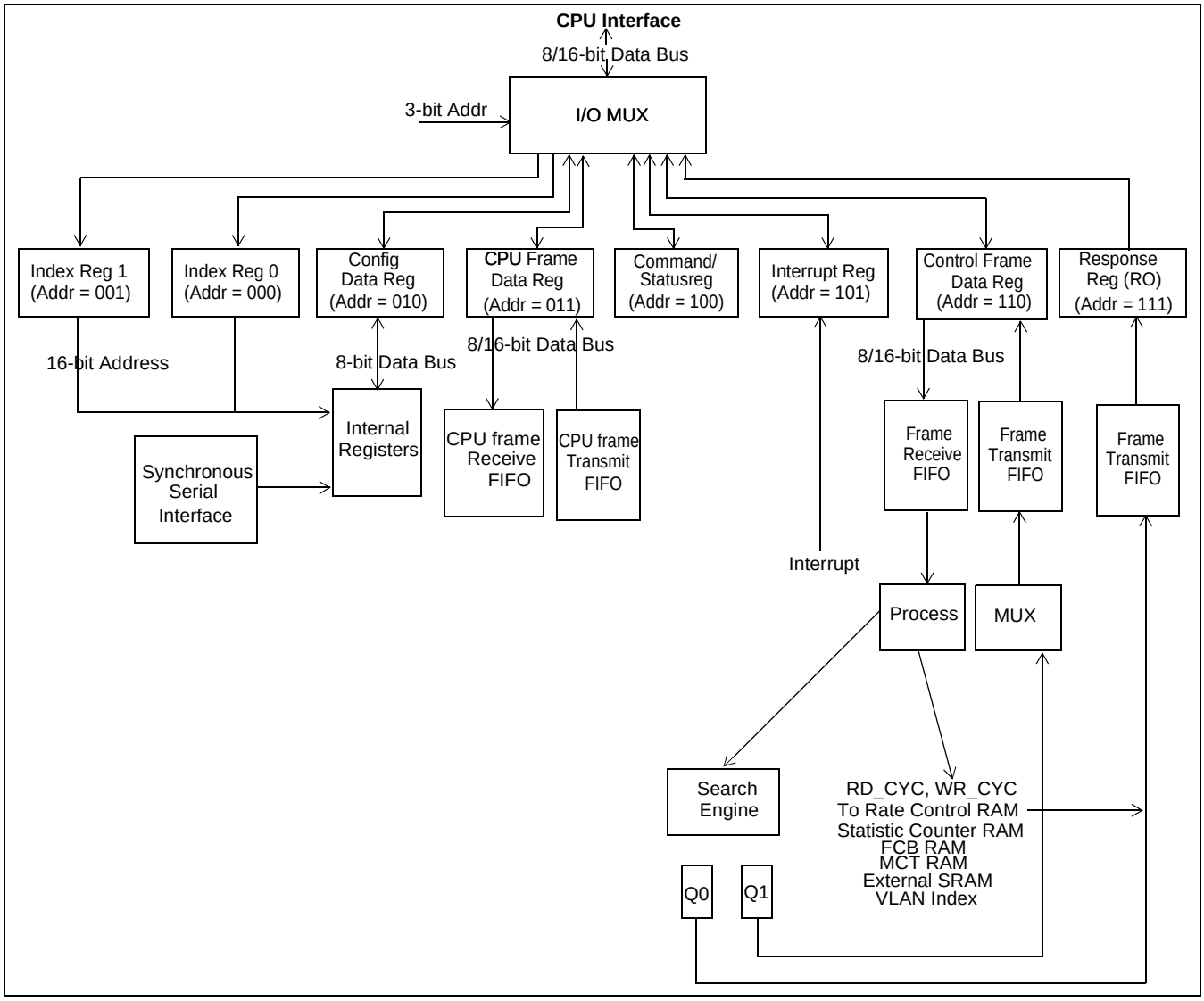


Figure 2 - Overview of the MVTX2802AG CPU Interface

2.3 Register Configuration, Frame Transmission, and Frame Reception

The MVTX2802AG has many programmable parameters, covering such functions as QoS weights, VLAN control. In managed mode, the CPU interface provides an easy way of configuring these parameters. The parameters are contained in 8-bit configuration registers. The MVTX2802AG allows indirect access to these registers, as follows:

- Two “index” registers (addresses 000 and 001) need to be written, to indicate the desired 16-bit register address.
- To indirectly configure the register addressed by the two index registers, a “configure data” register (address 010) must be written with the desired 8-bit data.
- Similarly, to read the value in the register addressed by the two index registers, the “configure data” register can now simply be read.

In summary, access to the many internal registers is carried out simply by directly accessing only three registers – two registers to indicate the address of the desired parameter, and one register to read or write a value. Of course, because there is only one bus master, there can never be any conflict between reading and writing the configuration registers.

2.3.1 Ethernet Frames

The CPU interface is also responsible for receiving and transmitting standard Ethernet frames to and from the CPU. To transmit a frame from the CPU:

- The CPU writes a “data frame” register (address 011) with the data it wants to transmit. After writing all the data, it then writes the frame size, destination port number, and frame status.
- The MVTX2802AG forwards the Ethernet frame to the desired destination port, no longer distinguishing the fact that the frame originated from the CPU.

To receive a frame into the CPU:

- The CPU receives an interrupt when an Ethernet frame is available to be received.
- Frame information arrives first in the data frame register. This includes source port number, frame size, and VLAN tag.
- The actual data follows the frame information. The CPU uses the frame size information to read the frame out.

In summary, receiving and transmitting frames to and from the CPU is a simple process that uses one direct access register only.

2.3.2 Control Frames

In addition to standard Ethernet frames described in the preceding section, the CPU is also called upon to handle special “Control frames,” generated by the MVTX2802AG and sent to the CPU. These proprietary frames are related to such tasks as statistics collection, MAC address learning, aging, etc. All Control frames are 64 bytes long. Transmitting and receiving these frames is similar to transmitting and receiving Ethernet frames, except that the register accessed is the “Control frame data” register (address 110).

Specifically, there are eight types of control frames generated by the CPU and sent to the MVTX2802AG:

- Memory read request
- Memory write request
- Learn MAC address
- Delete MAC address
- Search MAC address
- Learn IP Multicast address
- Delete IP Multicast address
- Search IP Multicast address

Note: Memory read and write requests by the CPU may include VLAN table, spanning tree, statistic counters, and similar updates.

In addition, there are nine types of Control Frames generated by the MVTX2802AG and sent to the CPU:

- Interrupt CPU when statistics counter rolls over
- Response to memory read request from CPU
- Learn MAC address
- Delete MAC address
- Delete IP Multicast address
- New VLAN port
- Age out VLAN port
- Response to search MAC address request from CPU
- Response to search IP Multicast address request from CPU

Note: Deleting IP Multicast address requests by the MVTX2802AG occur when the CPU issues a Learn IP Multicast address command but the search engine discovers no RAM space for storage.

The format of the Control Frame is described in the processor interface application note.

2.4 Unmanaged Mode

In unmanaged mode, the MVTX2802AG can be configured by EEPROM (24C02 or compatible) via an I^2C interface at boot time, or via a synchronous serial interface during operation. When the bootstrap Td[8] is set to '0' meaning EEPROM installed, the MVTX2802, acting as a master starts the data transfer from the memory to the switch.

2.5 I^2C Interface

The I^2C interface uses two bus lines, a serial data line (SDA) and a serial clock line (SCL). The SCL line carries the control signals that facilitate the transfer of information from EEPROM to the switch. Data transfer is 8-bit serial and bi-directional, at 50 Kbps. Data transfer is performed between master and slave IC using a request / acknowledgment style of protocol. The master IC generates the timing signals and terminates data transfer. The figure below shows the data transfer format.

START	SLAVE ADDRESS	R/W	ACK	DATA 1 (8 bits)	ACK	DATA 2	ACK	DATA M	ACK	STOP

Figure 3 - Data Transfer Format for I^2C Interface

2.5.1 Start Condition

Generated by the master, the MVTX2802AG. The bus is considered to be busy after the Start condition is generated. The Start condition occurs if while the SCL line is High, there is a High-to-Low transition of the SDA line.

Other than in the Start condition (and Stop condition), the data on the SDA line must be stable during the High period of SCL. The High or Low state of SDA can only change when SCL is Low. In addition, when the I^2C bus is free, both lines are High.

2.5.2 Address

The first byte after the Start condition determines which slave the master will select. The slave in our case is the EEPROM. The first seven bits of the first data byte make up the slave address.

2.5.3 Data Direction

The eighth bit in the first byte after the Start condition determines the direction (R/W) of the message. A master transmitter sets this bit to W; a master receiver sets this bit to R.

2.5.4 Acknowledgment

Like all clock pulses, the master generates the acknowledgment-related clock pulse. However, the transmitter releases the SDA line (High) during the acknowledgment clock pulse. Furthermore, the receiver must pull down the SDA line during acknowledge pulse so that it remains stable Low during the High period of this clock pulse. An acknowledgment pulse follows every byte transfer.

If a slave receiver does not acknowledge after any byte, then the master generates a Stop condition and aborts the transfer.

If a master receiver does not acknowledge after any byte, then the slave transmitter must release the SDA line to let the master generate the Stop condition.

2.5.5 Data

After the first byte containing the address, all bytes that follow are data bytes. Each byte must be followed by an acknowledge bit. Data is transferred MSB-first.

2.5.6 Stop Condition

Generated by the master. The bus is considered to be free after the Stop condition is generated. The Stop condition occurs if while the SCL line is High, there is a Low-to-High transition of the SDA line.

The I^2C interface serves the function of configuring the MVTX2802AG at boot time. The master is the MVTX2802AG, and the slave is the EEPROM memory.

2.6 Synchronous Serial Interface

The synchronous serial interface serves the function of configuring the MVTX2802AG *not* at boot time but via a PC. The PC serves as master and the MVTX2802AG serves as slave. The protocol for the synchronous serial interface is nearly identical to the I^2C protocol. The main difference is that there is no acknowledgment bit after each byte of data transferred.

The unmanaged MVTX2802AG uses a synchronous serial interface to program the internal registers. To reduce the number of signals required, the register address, command and data are shifted in serially through the PS_DO pin. PS_STROBE- pin is used as the shift clock. PS_DI- pin is used as data return path.

Each command consists of four parts.

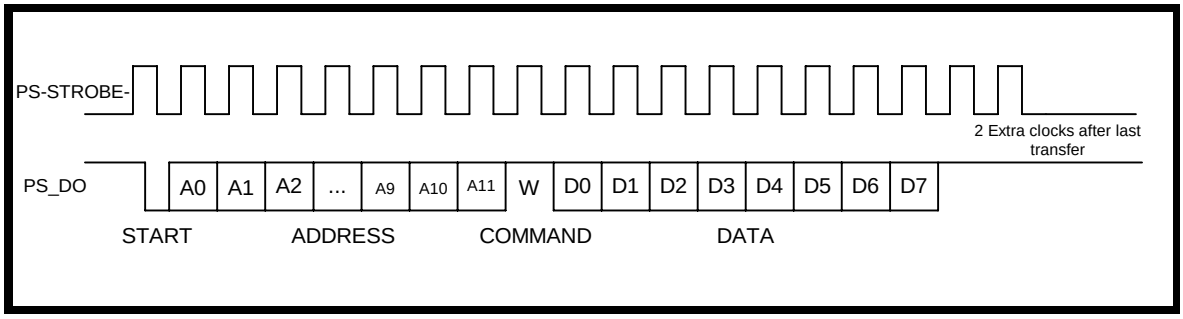
- START pulse
- Register Address
- Read or Write command
- Data to be written or read back

Any command can be aborted in the middle by sending an ABORT pulse to the MVTX2802AG.

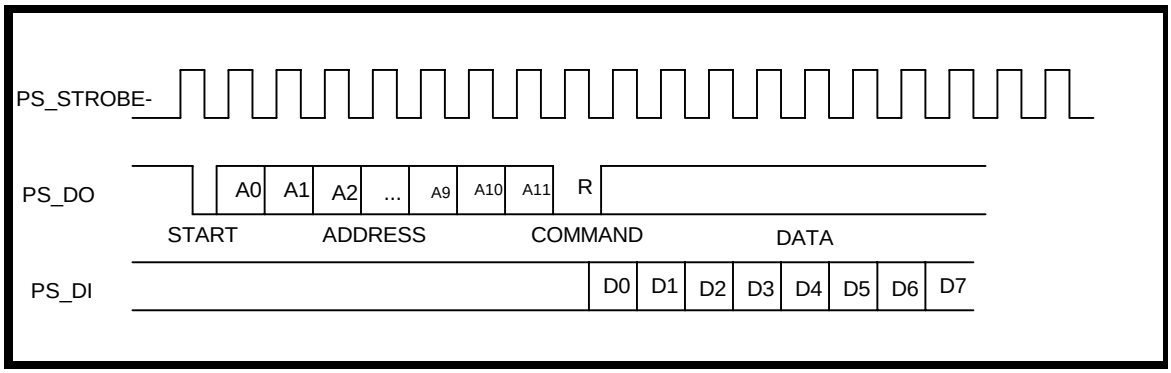
A START command is detected when PS_DO is sampled high at PS_STROBE - leading edge, and PS_DO is sampled low when PS_STROBE- falls.

An ABORT command is detected when PS_DO is sampled low at PS_STROBE - leading edge, and PS_DO is sampled high when PS_STROBE - falls.

2.6.1 Write Command



2.6.2 Read Command



All registers in the MVTX2802AG can be modified through this synchronous serial interface.

3.0 Data Forwarding Protocol

3.1 Unicast Data Frame Forwarding

When a frame arrives, it is assigned a handle in memory by the Frame Control Buffer Manager (FCB Manager). An FCB handle will always be available, because of advance buffer reservations.

The memory (ZBT-SRAM) interface is a 64-bit bus, connected to a ZBT-SRAM domain. The Receive DMA (RxDMA) is responsible for multiplexing the data and the address. On a port's "turn," the RxDMA will move 8 bytes (or up to the end-of-frame) from the port's associated RxFIFO into memory (Frame Data Buffer, or FDB).

Once an entire frame has been moved to the FDB, and a good end-of-frame (EOF) has been received, the Rx interface makes a switch request. The RxDMA arbitrates among multiple switch requests.

The switch request consists of the first 64 bytes of a frame, containing among other things, the source and destination MAC addresses of the frame. The search engine places a switch response in the switch response queue of the frame engine when done. Among other information, the search engine will have resolved the destination port of the frame and will have determined that the frame is unicast.

After processing the switch response, the Transmission Queue Manager (TxQ manager) of the frame engine is responsible for notifying the destination port that it has a frame to forward to it. But first, the TxQ manager has to decide whether or not to drop the frame, based on global FDB reservations and usage, as well as TxQ occupancy at the destination. If the frame is not dropped, then the TxQ manager links the frame's FCB to the

correct per-port-per-class TxQ. Unicast TxQ's are linked lists of transmission jobs, represented by their associated frames' FCB's. There is one linked list for each transmission class for each port. There are 8 classes for each of the 4 Gigabit ports – a total of 32 unicast queues.

The TxQ manager is responsible for scheduling transmission among the queues representing different classes for a port. When the port control module determines that there is room in the MAC Transmission FIFO (TxFIFO) for another frame, it requests the handle of a new frame from the TxQ manager. The TxQ manager chooses among the head-of-line (HOL) frames from the per-class queues for that port, using a Zarlink Semiconductor scheduling algorithm.

As at the transmit end, each of the 4 ports has time slots devoted solely to reading data from memory at the address calculated by port control. The Transmission DMA (TxDMA) is responsible for multiplexing the data and the address. On a port's turn, the TxDMA will move 8 bytes (or up to the EOF) from memory into the port's associated TxFIFO. After reading the EOF, the port control requests a FCB release for that frame. The TxDMA arbitrates among multiple buffer release requests.

The frame is transmitted from the TxFIFO to the line.

3.2 Multicast Data Frame Forwarding

After receiving the switch response, the TxQ manager has to make the dropping decision. A global decision to drop can be made, based on global FDB utilization and reservations. If so, then the FCB is released and the frame is dropped. In addition, a selective decision to drop can be made, based on the TxQ occupancy at some subset of the multicast packet's destinations. If so, then the frame is dropped at some destinations but not others, and the FCB is not released.

If the frame is not dropped at a particular destination port, then the TxQ manager formats an entry in the multicast queue for that port and class. Multicast queues are physical queues (unlike the linked lists for unicast frames). There are 4 multicast queues for each of the 4 Gigabit ports. During scheduling, the TxQ manager treats the unicast queue and the multicast queue of the same class as one logical queue.

The port control requests a FCB release only after the EOF for the multicast frame has been read by all ports to which the frame is destined.

3.3 Frame Forwarding To and From CPU

Frame forwarding **from** the CPU port to a regular transmission port is nearly the same as forwarding between transmission ports. The only difference is that the physical destination port must be indicated in addition to the destination MAC address. If an invalid port is indicated the frame is forwarded accordingly to the destination MAC address.

Frame forwarding **to** the CPU port is nearly the same as forwarding to a regular transmission port. The only difference is in frame scheduling. Instead of using the patent-pending scheduling algorithms, scheduling for the CPU port is simply based on strict priority. That is, a frame in a high priority queue will always be transmitted before a frame in a lower priority queue. There are four output queues to the CPU and one received queue.

4.0 Memory Interface

4.1 Overview

The figure below illustrates the first part of the ZBT-SRAM interface for the MVTX2802AG. As shown, a 64 bit bus ZBT-SRAM bank A is used for Tx/RxDMA access. Because the clock frequency is 133 MHz, the total memory bandwidth is 64 bits × 133 MHz = 8.5 Gbps, for frame data buffer (FDB) access.

Not shown in the figure are the CPU port RxDMA's and TxDMA's, each separately connected to its own bank selector.

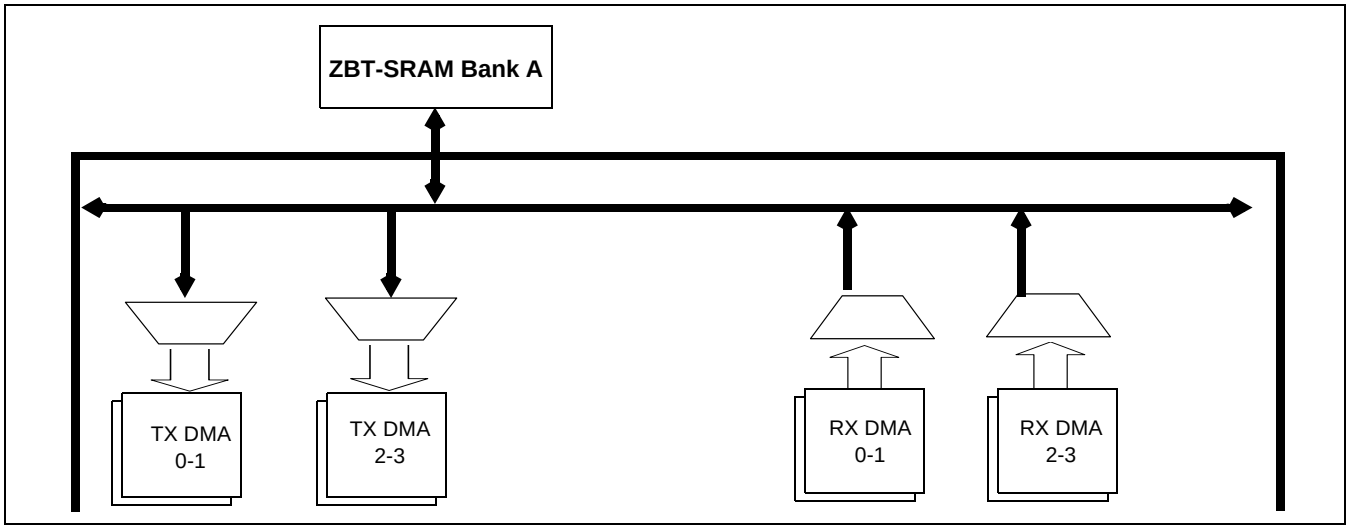


Figure 4 - MVTX2802AG SRAM Interface Block Diagram (DMAs for Gigabit Ports)

4.2 Detailed Memory Information

Because the memory bus is 64 bits wide, frames are broken into 8-byte granules, written to and read from each memory access. In the worst case, a 1-byte-long EOF granule gets written to memory Bank. This means that a 7-byte segment of memory bus is idle. The scenario results in a maximum 7 bytes of waste per frame, which is always acceptable because the interframe gap is 20 bytes.

The CPU management port gets treated like any other port, reading and writing to memory bank.

5.0 Search Engine

5.1 Search Engine Overview

The MVTX2802AG search engine is optimized for high throughput searching, with enhanced features to support:

- Up to 64K MAC addresses
- Up to 4K VLAN
- Up to 64K IP Multicast groups
- 4 groups of port trunking
- Traffic classification into 8 transmission priorities, and 2 drop precedence levels
- Packet filtering
- Security
- IP Multicast
- Per port, per VLAN Spanning Tree

5.2 Basic Flow

Shortly after a frame enters the MVTX2802AG and is written to the Frame Data Buffer (FDB), the frame engine generates a Switch Request, which is sent to the search engine. The switch request consists of the first 64 bytes of the frame, which contain all the necessary information for the search engine to perform its task. When the search engine is done, it writes to the Switch Response Queue, and the frame engine uses the information provided in that queue for scheduling and forwarding.

In performing its task, the search engine extracts and compresses the useful information from the 64-byte switch request. Among the information extracted are the source and destination MAC addresses, the transmission and discard priorities, whether the frame is unicast or multicast, and VLAN ID. Requests are sent to the external SRAM Switch Database to locate the associated entries in the external MCT table.

When all the information has been collected from external SRAM, the search engine has to compare the MAC address on the current entry with the MAC address for which it is searching. If it is not a match, the process is repeated on the internal MCT Table. All MCT entries other than the first of each linked list are maintained internal to the chip. If the desired MAC address is still not found, then the result is either learning (source MAC address unknown) or flooding (destination MAC address unknown).

In addition, VLAN information is used to select the correct set of destination ports for the frame (for multicast), or to verify that the frame's destination port is associated with the VLAN (for unicast).

If the destination MAC address belongs to a port trunk, then the trunk number is retrieved instead of the port number. But on which port of the trunk will the frame be transmitted? This is easily computed using a hash of the source and destination MAC addresses.

When all the information is compiled, the switch response is generated, as stated earlier. The search engine also interacts with the CPU with regard to learning and aging.

5.3 Search, Learning, and Aging

5.3.1 MAC Search

The search block performs source MAC address and destination MAC address (or destination IP address for IP multicast) searching. As we indicated earlier, if a match is not found, then the next entry in the linked list must be examined, and so on until a match is found or the end of the list is reached.

In tag based VLAN mode, if the frame is unicast, and the destination port is not a member of the correct VLAN, then the frame is dropped; otherwise, the frame is forwarded. If the frame is multicast, this same table is used to indicate all the ports to which the frame will be forwarded. Moreover, if port trunking is enabled, this block selects the destination port (among those in the trunk group).

In port based VLAN mode, a bitmap is used to determine whether the frame should be forwarded to the outgoing port. The main difference in this mode is that the bitmap is not dynamic. Ports cannot enter and exit groups because of real-time learning made by a CPU.

The MAC search block is also responsible for updating the source MAC address timestamp and the VLAN port association timestamp, used for aging.

5.3.2 Learning

The learning module learns new MAC addresses and performs port change operations on the MCT database. The goal of learning is to update this database as the networking environment changes over time. When CPU reporting is enabled, learning and port change will be performed when the CPU request queue has room, and a memory slot is available, and a "Learn MAC Address" message is sent to the CPU. When CPU reporting is disabled, learning and port change will be performed based on memory slot availability only.

In tag based VLAN mode, if the source port is not a member of a classified VLAN, a "New VLAN Port" message is sent to the CPU. The CPU can decide whether or not the source port can be added to the VLAN.

5.3.3 Aging

Aging time is controlled by register 400h and 401h.

The aging module scans and ages MCT entries based on a programmable “age out” time interval. As we indicated earlier, the search module updates the source MAC address and VLAN port association timestamps for each frame it processes. When an entry is ready to be aged, the entry is removed from the table, and a “Delete MAC Address” message is sent to inform the CPU.

Supported entry types are dynamic, static, source filter, destination filter, IP multicast, source and destination filter, and secure MAC address. Only dynamic entries can be aged; whether an entry is static or dynamic is maintained in the “status” field of the MCT data structure.

5.3.4 Data Structure

The MCT data structure is used for searching for MAC addresses. The structure is maintained by hardware in the search engine. The CPU can make requests to add to, delete from, or search the MCT database. The database is essentially a hash table, with collisions resolved by chaining. The database is partially external, and partially internal, as described earlier: the first MCT entry of each linked list is always located in the external SRAM, and the subsequent MCT’s are located internally.

5.3.5 VLAN Port Association Table

31	30	29 – 27	26 – 0
Valid	Route	Reserved	Port 8 to 0 is VLAN status

Port 8	Port 7	Port 6	Port 5	Port 4	Port 3	Port 2	Port 1	Port 0
VLAN status	Reserved	Reserved	Reserved	Reserved	VLAN status	VLAN status	VLAN status	VLAN status

VLAN STATUS [2:0]

- 000:Not a valid entry
- 001:Blocking status, no RX and TX
- 010:Not a VLAN member, spanning tree learn status
- 011:VLAN member, spanning tree learn status
- 100:Not a VLAN member, spanning tree forward status
- 101:VLAN member and is subject to aging, spanning tree forward status (Don’t use)
- 110:VLAN member and is subject to aging, spanning tree forward status
- 111:VLAN member and is not subject to aging, spanning tree forward status

CPU can create static VLAN port by writing the static status to the VLAN- PORT status entry.

Dynamic VLAN and Port association can be created by writing “110” to the VLAN STATUS. Hardware will age and refresh the entry based on the VLAN – PORT activity. When the VLAN – PORT is ready to be aged out, a message is sent to CPU and CPU can remove the VLAN – PORT association by writing “000” to the VLAN STATUS. As a result, the VLAN and PORT are no long associated and the VLAN domain is shrunk

6.0 Frame Engine

6.1 Data Forwarding Summary

- Enters the device at the RxMAC, the RxDMA will move the data from the MAC RxFIFO to the FDB. Data is moved in 8-byte granules in conjunction with the scheme for the SRAM interface.
- A switch request is sent to the Search Engine. The Search Engine processes the switch request.
- A switch response is sent back to the Frame Engine and indicates whether the frame is unicast or multicast, and its destination port or ports. A VLAN table lookup is performed as well.
- A Transmission Scheduling Request is sent in the form of a signal notifying the TxQ manager. Upon receiving a Transmission Scheduling Request, the device will format an entry in the appropriate Transmission Scheduling Queue (TxSch Q) or Queues. There is 8 transmission queues per Gigabit port, one for each priority. Creation of a queue entry either involves linking a new job to the appropriate linked list if unicast, or adding an entry to a physical queue if multicast.
- When the port is ready to accept the next frame, the TxQ manager will get the head-of-line (HOL) entry of one of the TxSch Qs, according to the transmission scheduling algorithm (so as to ensure per-class quality of service). The unicast linked list and the multicast queue for the same port-class pair are treated as one logical queue.
- The TxDMA will pull frame data from the memory and forward it granule-by-granule to the MAC TxFIFO of the destination port.

6.2 Frame Engine Details

This section briefly describes the functions of each of the modules of the MVTX2802AG frame engine.

6.2.1 FCB Manager

The FCB manager allocates FCB handles to incoming frames, and releases FCB handles upon frame departure. The FCB manager is also responsible for enforcing buffer reservations and limits. The default values can be determined by referring to Chapter 8. In addition, the FCB manager is responsible for buffer aging, and for linking unicast forwarding jobs to their correct TxSch Q. The buffer aging can be enabled or disabled by the bootstrap pin and the aging time is defined in register FCBAT.

6.2.2 Rx Interface

The Rx interface is mainly responsible for communicating with the RxMAC. It keeps track of the start and end of frame and frame status (good or bad). Upon receiving an end of frame that is good, the Rx interface makes a switch request.

6.2.3 RxDMA

The RxDMA arbitrates among switch requests from each Rx interface. It also buffers the first 64 bytes of each frame for use by the search engine when the switch request has been made.

6.2.4 TxQ Manager

First, the TxQ manager checks the per-class queue status and global Reserved resource situation, and using this information, makes the frame dropping decision after receiving a switch response. If the decision is not to drop, the TxQ manager requests that the FCB manager link the unicast frame's FCB to the correct per-port-per-class TxQ. If multicast, the TxQ manager writes to the multicast queue for that port and class. The TxQ manager can also trigger source port flow control for the incoming frame's source if that port is flow control enabled. Second, the TxQ manager handles transmission scheduling; it schedules transmission among the queues representing different classes for a port. Once a frame has been scheduled, the TxQ manager reads the FCB information and writes to the correct port control module.

6.3 Port Control

The port control module calculates the SRAM read address for the frame currently being transmitted. It also writes start of frame information and an end of frame flag to the MAC TxFIFO. When transmission is done, the port control module requests that the buffer be released.

6.4 TxDMA

The TxDMA multiplexes data and address from port control, and arbitrates among buffer release requests from the port control modules.

7.0 Quality of Service and Flow Control

7.1 Model

Quality of service (QoS) is an all-encompassing term for which different people have different interpretations. In this chapter, by quality of service assurances, we mean the allocation of chip resources so as to meet the latency and bandwidth requirements associated with each traffic class. We do not presuppose anything about the offered traffic pattern. If the traffic load is light, then ensuring quality of service is straightforward. But if the traffic load is heavy, the MVTX2802AG must intelligently allocate resources so as to assure quality of service for high priority data.

We assume that the network manager knows his applications, such as voice, file transfer, or web browsing, and their relative importance. The manager can then subdivide the applications into classes and set up a service contract with each. The contract may consist of bandwidth or latency assurances per class. Sometimes it may even reflect an estimate of the traffic mix offered to the switch, though this is not required.

The table below shows examples of QoS applications with eight transmission priorities, including best effort traffic for which we provide no bandwidth or latency assurances.

Class	**Example Assured Bandwidth (user defined)**	**Low Drop Subclass (If class is oversubscribed, these packets are the last to be dropped.)**	**High Drop Subclass (If class is oversubscribed, these packets are the first to be dropped.)**
Highest transmission priorities, P7 **Latency < 200 µs**	300 Mbps	**Sample application:** control information	
Highest transmission priorities, P6 **Latency < 200 µs**	200 Mbps	**Sample applications:** phone calls; circuit emulation	**Sample application:** training video; other multimedia
Middle transmission priorities, P5 **Latency < 400 µs**	125 Mbps	**Sample application:** interactive activities	**Sample application:** non-critical interactive activities
Middle transmission priorities, P4 **Latency < 800 µs**	250 Mbps	**Sample application:** web business	
Low transmission priorities, P3 **Latency < 1600 µs**	80 Mbps	**Sample application:** file backups	
Low transmission priorities, P2 **Latency < 3200 µs**	45 Mbps	**Sample application:** email	**Sample application:** web research
Best effort, P1-P0	–	**Sample application:** casual web browsing	
TOTAL	1 Gbps		

Table 1 - Two-dimensional World Traffic

In our model, it is possible that a class of traffic may attempt to monopolize system resources by sending data at a rate in excess of the contractually assured bandwidth for that class. A well-behaved class offers traffic at a rate no greater than the agreed-upon rate. By contrast, a misbehaving class offers traffic that exceeds the agreed-upon rate. A misbehaving class is formed from an aggregation of misbehaving microflows. To achieve high link utilization, a misbehaving class is allowed to use any idle bandwidth. However, the quality of service (QoS) received by well-behaved classes must never suffer.

As Table 1 illustrates, each traffic class may have its own distinct properties and applications. As shown, classes may receive bandwidth assurances or latency bounds. In the example, P7, the highest transmission class, requires that all frames be transmitted within 0.2 ms, and receives 30% of the 1 Gbps of bandwidth at that port.

Best-effort (P1-P0) traffic forms a lower tier of service that only receives bandwidth when none of the other classes have any traffic to offer.

In addition, each transmission class has two subclasses, high-drop and low-drop. Well-behaved users should not lose packets. But poorly behaved users – users who send data at too high a rate – will encounter frame loss, and the first to be discarded will be high-drop. Of course, if this is insufficient to resolve the congestion, eventually some low-drop frames are dropped as well.

Table 1 shows that different types of applications may be placed in different boxes in the traffic table. For example, web search may fit into the category of high-loss, high-latency-tolerant traffic, whereas VoIP fits into the category of low-loss, low-latency traffic.

7.2 Four QoS Configurations

There are four basic pieces to QoS scheduling in the MVTX2802AG: strict priority (SP), delay bound, weighted fair queuing (WFQ), and best effort (BE). Using these four pieces, there are four different modes of operation, as shown in Table 2.

<table>
<tr><td></td><td>P7</td><td>P6</td><td>P5</td><td>P4</td><td>P3</td><td>P2</td><td>P1</td><td>P0</td></tr>
<tr><td>Op1 (default)</td><td colspan="6">Delay Bound</td><td colspan="2">BE</td></tr>
<tr><td>Op2</td><td colspan="2">SP</td><td colspan="4">Delay Bound</td><td colspan="2">BE</td></tr>
<tr><td>Op3</td><td colspan="2">SP</td><td colspan="6">WFQ</td></tr>
<tr><td>Op4</td><td colspan="8">WFQ</td></tr>
</table>

Table 2- Four QoS configurations per port.

The default configuration is six delay-bounded queues and two best-effort queues. The delay bounds per class are 0.16 ms for P7 and P6, 0.32 ms for P5, 0.64 ms for P4, 1.28 ms for P3, and 2.56 ms for P2. Best effort traffic is only served when there is no delay-bounded traffic to be served. P1 has strict priority over P0.

We have a second configuration in which there are two strict priority queues, four delay bounded queues, and two best effort queues. The delay bounds per class are 0.32 ms for P5, 0.64 ms for P4, 1.28 ms for P3, and 2.56 ms for P2. If the user is to choose this configuration, it is important that P7-P6 (SP) traffic be either policed or implicitly bounded (e.g. if the incoming SP traffic is very light and predictably patterned). Strict priority traffic, if not admission-controlled at a prior stage to the MVTX2802AG, can have an adverse effect on all other classes' performance. P7 and P6 are both SP classes, and P7 has strict priority over P6.

The third configuration contains two strict priority queues and six queues receiving a bandwidth partition via WFQ. As in the second configuration, strict priority traffic needs to be carefully controlled.

In the fourth configuration, all queues are served using a WFQ service discipline.

7.3 Delay Bound

In the absence of a sophisticated QoS server and signaling protocol, the MVTX2802AG may not be assured of the mix of incoming traffic ahead of time. To cope with this uncertainty, our delay assurance algorithm dynamically adjusts its scheduling and dropping criteria, guided by the queue occupancies and the due dates of their head-of-line (HOL) frames. As a result, we assure latency bounds for all admitted frames with high confidence, even in the presence of system-wide congestion. Our algorithm identifies misbehaving classes and intelligently discards frames at no detriment to well-behaved classes. Our algorithm also differentiates between high-drop and low-drop traffic with a weighted random early drop (WRED) approach. Random early dropping prevents congestion by randomly dropping a percentage of high-drop frames even before the chip's buffers are completely full, while still largely sparing low-drop frames. This allows high-drop frames to be discarded early, as a sacrifice for future low-drop frames. Finally, the delay bound algorithm also achieves bandwidth partitioning among classes.

7.4 Strict Priority and Best Effort

When strict priority is part of the scheduling algorithm, if a queue has even one frame to transmit, it goes first. Two of our four QoS configurations include strict priority queues. The goal is for strict priority classes to be used for IETF expedited forwarding (EF), where performance guarantees are required. As we have indicated, it is important that strict priority traffic be either policed or implicitly bounded, so as to keep from harming other traffic classes.

When best effort is part of the scheduling algorithm, a queue only receives bandwidth when none of the other classes have any traffic to offer. Two of our four QoS configurations include best effort queues. The goal is for best effort classes to be used for non-essential traffic, because we provide no assurances about best effort performance. However, in a typical network setting, much best effort traffic will indeed be transmitted, and with an adequate degree of expediency.

Because we do not provide any delay assurances for best effort traffic, we do not enforce latency by dropping best effort traffic. Furthermore, because we assume that strict priority traffic is carefully controlled before entering the MVTX2802AG, we do not enforce a fair bandwidth partition by dropping strict priority traffic. To summarize, dropping to enforce quality of service (i.e. bandwidth or delay) does not apply to strict priority or best effort queues. We only drop frames from best effort and strict priority queues when global buffer resources become scarce.

7.5 Weighted Fair Queuing

In some environments – for example, in an environment in which delay assurances are not required, but precise bandwidth partitioning on small time scales is essential - WFQ may be preferable to a delay-bounded scheduling discipline. The MVTX2802AG provides the user with a WFQ option with the understanding that delay assurances cannot be provided if the incoming traffic pattern is uncontrolled. The user sets eight WFQ "weights" such that all weights are whole numbers and sum to 64. This provides per-class bandwidth partitioning with error within 2%.

In WFQ mode, though we do not assure frame latency, the MVTX2802AG still retains a set of dropping rules that helps to prevent congestion and trigger higher level protocol end-to-end flow control.

As before, when strict priority is combined with WFQ, we do not have special dropping rules for the strict priority queues, because the input traffic pattern is assumed to be carefully controlled at a prior stage. However, we do indeed drop frames from SP queues for global buffer management purposes. In addition, queues P1 and P0 are treated as best effort from a dropping perspective, though they still are assured a percentage of bandwidth from a WFQ scheduling perspective. What this means is that these particular queues are only affected by dropping when the global buffer count becomes low.

7.6 Shaper

Although traffic shaping is not a primary function of the MVTX2802AG, the chip does implement a shaper for expedited forwarding (EF). Our goal in shaping is to control the peak and average rate of traffic exiting the MVTX2802AG. Shaping is limited to class P6 (the second highest priority). This means that class P6 will be the

class used for EF traffic. (By contrast, we assume class P7 will be used for control packets only.) If shaping is enabled for P6, then P6 traffic must be scheduled using strict priority. With reference to Table 2, only the middle two QoS configurations may be used.

Peak rate is set using a programmable whole number, no greater than 64 (register QOS-CREDIT_C6_Gn). For example, if the setting is 32, then the peak rate for shaped traffic is 32/64 × 1000 Mbps = 500 Mbps. Average rate is also a programmable whole number, no greater than 64, and no greater than the peak rate. For example, if the setting is 16, then the average rate for shaped traffic is 16/64 × 1000 Mbps = 250 Mbps. As a consequence of the above settings in our example, shaped traffic will exit the MVTX2802AG at a rate always less than 500 Mbps, and averaging no greater than 250 Mbps.

Also, when shaping is enabled, it is possible for a P6 queue to explode in length if fed by a greedy source. The reason is that a shaper is by definition not work-conserving; that is, it may hold back from sending a packet even if the line is idle. Though we do have global resource management, we do nothing to prevent this situation locally. We assume SP traffic is policed at a prior stage to the MVTX2802AG.

7.7 WRED Drop Threshold Management Support

To avoid congestion, the Weighted Random Early Detection (WRED) logic drops packets according to specified parameters. The following table summarizes the behaviour of the WRED logic.

	P7	P6	P5	P4	P3	P2	High Drop	Low Drop
Level 1 N ≥ 240	|P7| ≥ A KB	|P6| ≥ B KB	|P5| ≥ C KB	|P4| ≥ D KB	|P3| ≥ E KB	|P2| ≥ F KB	X%	0%
Level 2 N ≥ 280							Y%	Z%
Level 3 N ≥ 320							100%	100%

Table 3- WRED Dropping Scheme.

In the table, |Px| is the byte count in queue Px. The WRED logic has three drop levels, depending on the value of N, which is based on the number of bytes in the priority queues. If delay bound scheduling is used, N equals 16|P7| + 16|P6| + 8|P5| + 4|P4| + 2|P3| + |P2|. If WFQ scheduling is used, N equals |P7| + |P6| + |P5| + |P4| + |P3| + |P2|. Each drop level has defined high-drop and low-drop percentages, which indicate the percentage of high-drop and low-drop packets that will be dropped at that level. The X, Y, and Z percent parameters can be programmed using the registers RDRC0 and RDRC1. Parameters A-F are the byte count thresholds for each priority queue, and are also programmable. When using delay bound scheduling, the values selected for A-F also control the approximate bandwidth partition among the traffic classes; see application note.

7.8 Buffer Management

Because the number of frame data buffer (FDB) slots is a scarce resource, and because we want to ensure that one misbehaving source port or class cannot harm the performance of a well-behaved source port or class, we introduce the concept of buffer management into the MVTX2802AG. Our buffer management scheme is designed to divide the total buffer space into numerous reserved regions and one shared pool (see Figure 5).

As shown in the figure, the FDB pool is divided into several parts. A reserved region for temporary frames stores frames prior to receiving a switch response. Such a temporary region is necessary, because when the frame first enters the MVTX2802AG, its destination port and class are as yet unknown, and so the decision to drop or not needs to be temporarily postponed. This ensures that every frame can be received first before subjecting it to the frame drop discipline after classifying.

Six reserved sections, one for each of the highest six priority classes, ensure a programmable number of FDB slots per class. The lowest two classes do not receive any buffer reservation.

Another segment of the FDB reserves space for each of the 4 Gigabit ports and CPU port. These source port buffer reservations are programmable. These 9 reserved regions make sure that no well-behaved source port can be blocked by another misbehaving source port.

In addition, there is a shared pool, which can store any type of frame. The registers related to the Buffer Management logic are:

- PRG- Port Reservation for Gigabit Ports and CPU port
- SFCB- Share FCB Size
- C2RS- Class 2 Reserved Size
- C3RS- Class 3 Reserved Size
- C4RS- Class 4 Reserved Size
- C5RS- Class 5 Reserved Size
- C6RS- Class 6 Reserved Size
- C7RS- Class 7 Reserved Size

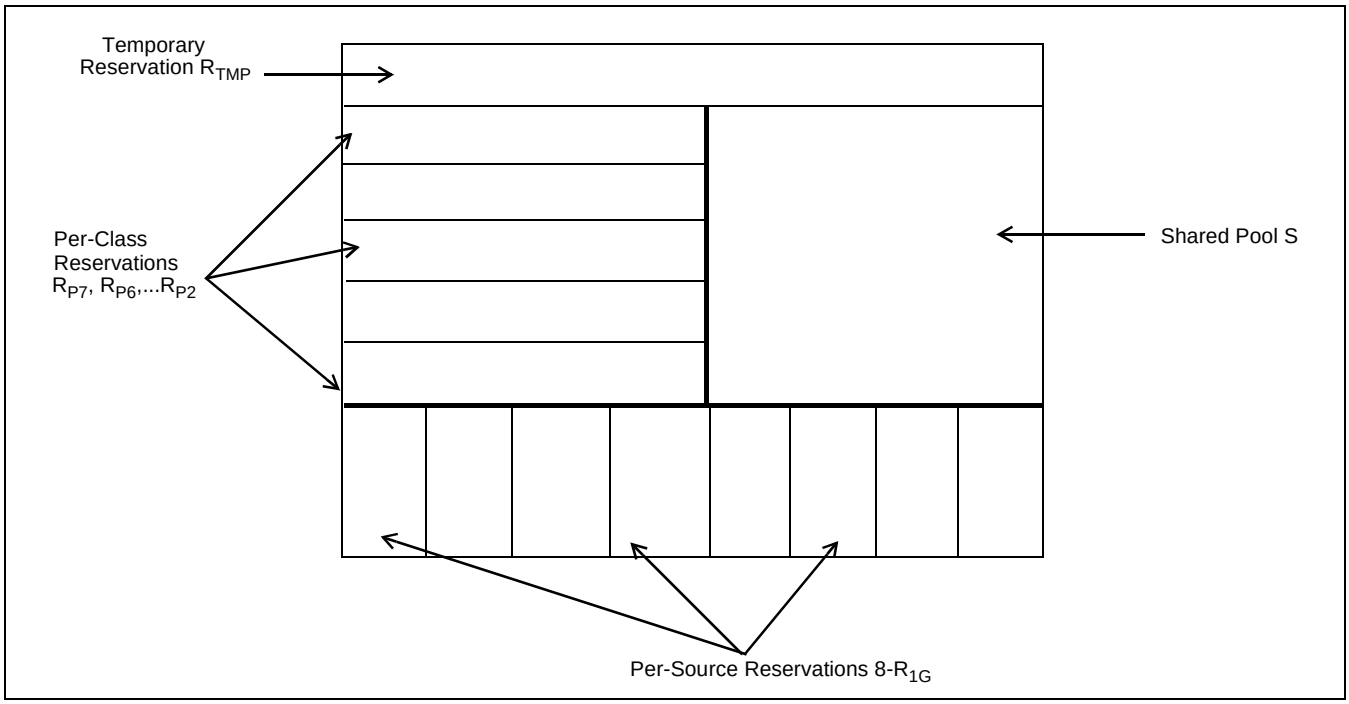


Figure 5 - Buffer Partition Scheme Used in the MVTX2802AG

7.8.1 Dropping When Buffers Are Scarce

Summarizing the two examples of local dropping discussed earlier in this chapter:

- If a queue is a delay-bounded queue, we have a multilevel WRED drop scheme, designed to control delay and partition bandwidth in case of congestion.
- If a queue is a WFQ-scheduled queue, we have a multilevel WRED drop scheme, designed to prevent congestion.

In addition to these reasons for dropping, the MVTX2802AG also drops frames when global buffer space becomes scarce. The function of buffer management is to ensure that such droppings cause as little blocking as possible.

7.9 Flow Control Basics

Because frame loss is unacceptable for some applications, the MVTX2802AG provides a flow control option. When flow control is enabled, scarcity of buffer space in the switch may trigger a flow control signal; this signal tells a source port, sending a packet to this switch, to temporarily hold off.

While flow control offers the clear benefit of no packet loss, it also introduces a problem for quality of service. When a source port receives an Ethernet flow control signal, all microflows originating at that port, well-behaved or not, are halted. A single packet destined for a congested output can block other packets destined for un-congested outputs. The resulting head-of-line blocking phenomenon means that quality of service cannot be assured with high confidence when flow control is enabled.

In the MVTX2802AG, each source port can independently have flow control enabled or disabled. For flow control enabled ports, by default all frames are treated as lowest priority during transmission scheduling. This is done so that those frames are not exposed to the WRED Dropping scheme. Frames from flow control enabled ports feed to only one queue at the destination, the queue of lowest priority. What this means is that if flow control is enabled for a given source port, then we can guarantee that no packets originating from that port will be lost, but at the possible expense of minimum bandwidth or maximum delay assurances. In addition, these “downgraded” frames may only use the shared pool or the per-source reserved pool in the FDB; frames from flow control enabled sources may not use reserved FDB slots for the highest six classes (P2-P7).

The MVTX2802AG does provide a system-wide option of permitting normal QoS scheduling (and buffer use) for frames originating from flow control enabled ports. When this programmable option is active, it is possible that some packets may be dropped, even though flow control is on. The reason is that intelligent packet dropping is a major component of the MVTX2802AG’s approach to ensuring bounded delay and minimum bandwidth for high priority flows.

7.9.1 Unicast Flow Control

For unicast frames, flow control is triggered by source port resource availability. Recall that the MVTX2802AG’s buffer management scheme allocates a reserved number of FDB slots for each source port. If a programmed number of a source port’s reserved FDB slots have been used, then flow control Xoff is triggered. Xon is triggered when a port is currently being flow controlled, and all of that port’s reserved FDB slots have been released.

Note that the MVTX2802AG’s per-source-port FDB reservations assure that a source port that sends a single frame to a congested destination will not be flow controlled.

7.9.2 Multicast Flow Control

In unmanaged mode, a global buffer counter triggers flow control for multicast frames. When the system exceeds a programmable threshold of multicast packets, Xoff is triggered. Xon is triggered when the system returns below this threshold. MCC register programs the threshold.

In managed mode, per-VLAN flow control is used for multicast frames. In this case, flow control is triggered by congestion at the destination. The MVTX2802AG checks each destination to which a multicast packet is headed. For each destination port, the occupancy of the lowest-priority transmission queue (measured in number of frames) is compared against a programmable congestion threshold. If congestion is detected at even one of the packet’s destinations, then Xoff is triggered.

In addition, each source port has an 4-bit port map recording which port or ports of the multicast frame’s fanout were congested at the time Xoff was triggered. All ports are continuously monitored for congestion, and a port is identified as uncongested when its queue occupancy falls below a fixed threshold. When all those ports that were originally marked as congested in the port map have become uncongested, then Xon is triggered, and the 4-bit vector is reset to zero.

The MVTX2802AG also provides the option of disabling multicast flow control.

Note: If port flow control is on, QoS performance will be affected.

7.10 Mapping to IETF Diffserv Classes

The mapping between priority classes discussed in this chapter and elsewhere is shown below.

MVTX2802AG	P7	P6	P5	P4	P3	P2	P1	P0
IETF	NM	EF	AF0	AF1	AF2	AF3	BE0	BE1

Table 4- Mapping between MVTX2802AG and IETF Diffserv Classes for Gigabit Ports

As the table illustrates, P7 is used solely for network management (NM) frames. P6 is used for expedited forwarding service (EF). Classes P2 through P5 correspond to an assured forwarding (AF) group of size 4. Finally, P0 and P1 are two best effort (BE) classes.

Features of the MVTX2802AG that correspond to the requirements of their associated IETF classes are summarized in the table below.

Network management (NM) and Expedited forwarding (EF)	• Global buffer reservation for NM and EF • Shaper for EF traffic • Option of strict priority scheduling • No dropping if admission controlled
Assured forwarding (AF)	• Four AF classes • Programmable bandwidth partition, with option of WFQ service • Option of delay-bounded service keeps delay under fixed levels even if not admission-controlled • Random early discard, with programmable levels • Global buffer reservation for each AF class
Best effort (BE)	• Two BE classes • Service only when other queues are idle means that QoS not adversely affected • Random early discard, with programmable levels • Traffic from flow control enabled ports automatically classified as BE

Table 5- MVTX2802AG Features Enabling IETF Diffserv Standards

8.0 Port Trunking

8.1 Features and Restrictions

A port group (i.e. trunk) can include up to 4 physical ports, but all of the ports in a group must be in the same MVTX2802AG.

In managed mode, there are four trunk groups total.

In unmanaged mode, the MVTX2802AG provides several pre-assigned trunk group options, containing as many as 4 ports per group, or alternatively, as many as 4 total groups.

Load distribution among the ports in a trunk for unicast is performed using hashing based on source MAC address and destination MAC address. The other options include source MAC address only, destination MAC address only. Load distribution for multicast is performed similarly.

If a VLAN includes any of the ports in a trunk group, all the ports in that trunk group should be in the same VLAN member map.

The MVTX2802AG also provides a safe fail-over mode for port trunking automatically. If one of the ports in the trunking group goes down, the MVTX2802AG will automatically redistribute the traffic over to the remaining ports in the trunk in unmanaged mode. In managed mode, the software can perform similar tasks.

8.2 Unicast Packet Forwarding

The search engine finds the destination MCT entry, and if the status field says that the destination address found belongs to a trunk, then the group number is retrieved instead of the port number. In addition, if the source address belongs to a trunk, then the source port's trunk membership register is checked to determine if the address has moved.

A hash key is used to determine the appropriate forwarding port, based on some combination of the source and destination MAC addresses for the current packet.

The search engine retrieves the VLAN member ports from the VLAN index table, which consists of 4K entries.

The search engine retrieves the VLAN member ports from the ingress port's VLAN map. Based on the destination MAC address, the search engine determines the egress port from the MCT database. If the egress port is member of a trunk group, the packet will be forward to only one port of the trunk group. The VLAN map is used to check whether the egress port is a member of the VLAN, based on the ingress port. If it is a member, the packet is forwarded otherwise it is discarded.

8.3 Multicast Packet Forwarding

For multicast packet forwarding, the device must determine the proper set of ports from which to transmit the packet based on the VLAN index and hash key.

Two functions are required in order to distribute multicast packets to the appropriate destination ports in a port trunking environment.

- Determining one forwarding port per group.
- For multicast packets, all but one port per group, the forwarding port, must be excluded.

8.4 Preventing Multicast Packets from Looping Back to the Source Trunk

The search engine needs to prevent a multicast packet from sending to a port that is in the same trunk group with the source port. This is because, when we select the primary forwarding port for each group, we do not take the source port into account. To prevent this, we simply apply one additional filter, so as to block that forwarding port for this multicast packet.

9.0 LED Interface

9.1 Introduction

The MVTX2802AG LED block provides two interfaces: a serial output channel, and a parallel time-division interface. The serial output channel provides port status information from the MVTX2802AG chip in a continuous serial stream. This means that a low cost external device must be used to decode the serial data and to drive an LED array for display.

By contrast, the parallel time-division interface supports a glueless LED module. Indeed, the parallel interface can directly drive low-current LEDs without any extra logic. The pin LED_PM is used to select serial or parallel mode.

For some LED signals, the interface also provides a blinking option. Blinking may be enabled for LED signals TxD, RxD, COL, and FC (to be described later). The pin LED_BLINK is used to enable blinking, and the blinking frequency is around 160 ms.

9.2 Serial Mode

In serial mode, the following pins are utilized:

- LED_SYNCO – a sync pulse that defines the boundary between status frames
- LED_CLKO – the clock signal
- LED_DO – a continuous serial stream of data for all status LEDs that repeats once every frame time

In each cycle (one frame of status information, or one sync pulse), 16×8 bits of data are transmitted on the LED_DO signal. The sequence of transmission of data bits is as shown in the figure below:

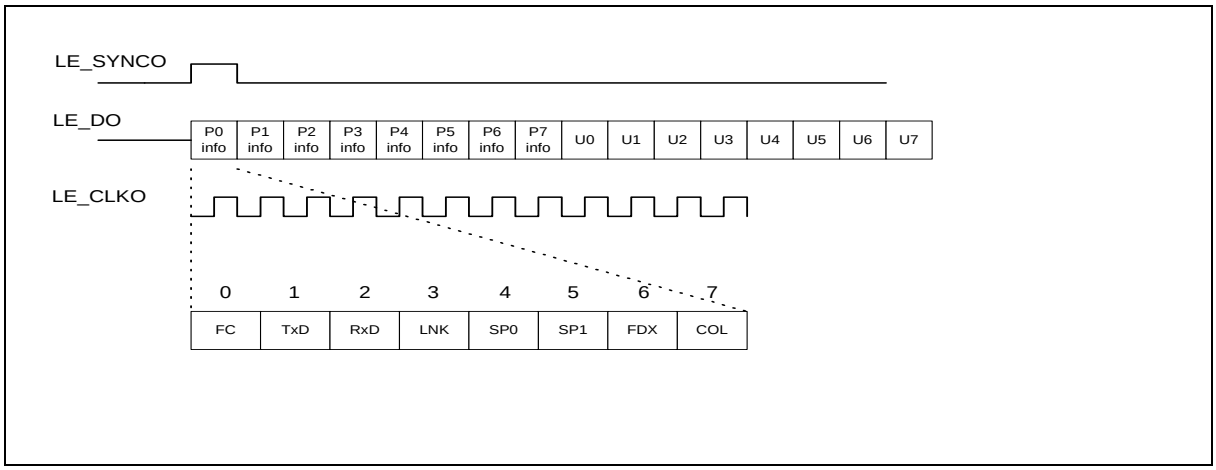


Figure 6 - Timing diagram for serial mode in LED interface

The status bits shown in here are flow control (FC), transmitting data (TxD), receiving data (RxD), link up (LNK), speed (SP0 and SP1), full duplex (FDX), and collision (COL). Note that SP[1:0] is defined as 10 for 1 Gbps, 01 for 100 Mbps, and 00 for 10 Mbps.

Also note that U0-U7 represent user-defined sub-frames in which additional status information may be embedded. We will see later that the MVTX2802AG provides registers that can be written by the CPU to indicate this additional status information as it becomes available.

9.3 Parallel Mode

In parallel mode, the following pins are utilized:

- LED_PORT_SEL[9:0] – indicates which of the 4 Gigabit port status bytes or 2 user-defined status bytes is being read out
- LED_BYTEOUT_[7:0] – provides 8 bits for 8 different port status indicators. Note that these bits are active low.

By default, the system is in parallel mode. In parallel mode, the 10 status bytes are scanned in a continuous loop, with one byte read out per clock cycle, and the appropriate port select bit asserted.

9.4 LED Control Registers

An LED Control Register can be used for programming the LED clock rate, sample hold time, and pattern in parallel mode.

In addition, the MVTX2802AG provides 8 registers called LEDUSER[7:0] for user-defined status bytes. During operation, the CPU can write values to these registers, which will be read out to the LED interface output (serial or parallel). Only LEDUSER[1:0] are used in parallel mode. The content of the LEDUSER registers will be sent out by the LED serial shift logic, or in parallel mode, a byte at a time.

Because in parallel mode there are only two user-defined registers, LEDUSER[7:2] is shared with LEDSIG[7:2].

For LEDSIG[j], where j = 2, 3, …, 6, the corresponding register is used for programming the LED pin LED_BYTEOUT_[j]. The format is as follows:

7			4	3			0
COL	FDX	SP1	SP0	COL	FDX	SP1	SP0

Bits [3:0]	Signal polarity:	0: do not invert polarity (high true) 1: invert polarity
Bits [7:4]	Signal select:	0: do not select 1: select the corresponding bit

For j = 2, 3, …, 5, the value of LED_BYTEOUT_[j] equals the logical AND of all selected bits. For j = 6, the value is equal to the logical OR. Therefore, the programmable LEDSIG[5:2] registers allow any conjunctive formula including any of the 4 status bits (COL, FDX, SP1, SP0) or their negations to be sent to the LED_BYTEOUT_[5:2] pins. Similarly, the programmable LEDSIG[6] register allows any disjunctive formula including any of the 4 status bits or their negations to be sent to pin LED_BYTEOUT_[6].

LEDSIG[7] is used for programming both LED_BYTEOUT_[1] and LED_BYTEOUT_[0]. As we will see, it has other functions as well. The format is as follows:

7			4	3			0
GP	RxD	TxD	FC	P6	RxD	TxD	FC

Bits [7]
- Global output polarity: this bit controls the output polarity of all LED_BYTEOUT_ and LED_PORT_SEL pins. **(Default 0)**
  - 0: do not invert polarity (LED_BYTEOUT_[7:0] are high activated; LED_PORT_SEL[9:0] are low activated)
  - 1: invert polarity (LED_BYTEOUT_[7:0] are low activated; LED_PORT_SEL[9:0] are high activated)

Bits [6:4]
- Signal select:
  - 0: do not select
  - 1: select the corresponding bit
- The value of LED_BYTEOUT_[1] equals the logical OR of all selected bits. **(Default 110)**

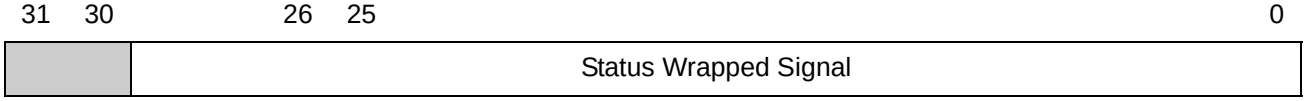
Bit [3]
- Polarity control of LED_BYTEOUT_[6] **(Default 0)**
  - 0: do not invert
  - 1: invert

Bits [2:0]
- Signal select:
  - 0: do not select
  - 1: select the corresponding bit
- The value of LED_BYTEOUT_[0] equals the logical OR of all selected bits. (**Default 001)**

10.0 Hardware Statistics Counter

10.1 9.1Hardware Statistics Counters List

MVTX2802AG hardware provides a full set of statistics counters for each Ethernet port. The CPU accesses these counters through the CPU interface. All hardware counters are rollover counters. When a counter rolls over, the CPU is interrupted, so that long-term statistics may be kept. The MAC detects all statistics, except for the delay exceed discard counter (detected by buffer manager) and the filtering counter (detected by queue manager). The following is the wrapped signal sent to the CPU through the command block.



B[0]	0-d	Bytes Sent (D)
B[1]	1-L	Unicast Frame Sent
B[2]	1-U	Frame Send Fail
B[3]	2-I	Flow Control Frames Sent
B[4]	2-u	Non-Unicast Frames Sent
B[5]	3-d	Bytes Received (Good and Bad) (D)
B[6]	4-d	Frames Received (Good and Bad) (D)
B[7]	5-d	Total Bytes Received (D)

B[8]	6-L	Total Frames Received
B[9]	6-U	Flow Control Frames Received
B[10]	7-l	Multicast Frames Received
B[11]	7-u	Broadcast Frames Received
B[12]	8-L	Frames with Length of 64 Bytes
B[13]	8-U	Jabber Frames
B[14]	9-L	Frames with Length Between 65-127 Bytes
B[15]	9-U	Oversize Frames
B[16]	A-l	Frames with Length Between 128-255 Bytes
B[17]	A-u	Frames with Length Between 256-511 Bytes
B[18]	B-l	Frames with Length Between 512-1023 Bytes
B[19]	B-u	Frames with Length Between 1024-1528 Bytes
B[20]	C-l	Fragments
B[21]	C-U1	Alignment Error
B[22]	C-U	Undersize Frames
B[23]	D-l	CRC
B[24]	D-u	Short Event
B[25]	E-l	Collision
B[26]	E-u	Drop
B[27]	F-l	Filtering Counter
B[28]	F-U1	Delay Exceed Discard Counter
B[29]	F-U	Late Collision
B[30]		Link Status Change
B[31]		Current link status

Notation: X-Y

X: Address in the contain memory

Y: Size and bits for the counter

d: D Word counter

L: 24 bits counter bit[23:0]

U: 8 bits counter bit[31:24]

U1: 8 bits counter bit[23:16]

l: 16 bits counter bit[15:0]

u: 16 bits counter bit[31:16]

10.2 IEEE 802.3 HUB Management (RFC 1213)

10.2.1 Event Counters

10.2.1.1 READABLEOCTET

Counts number of bytes (i.e. octets) contained in **good valid frames** received.

Frame size: ≥ 64 bytes, ≤ 1522 bytes if VLAN Tagged;
1518 bytes if not VLAN Tagged

No FCS (i.e. checksum) error

No collisions

10.2.1.2 READABLEFRAME

Counts number of **good valid frames** received.

Frame size: ≥ 64 bytes, ≤ 1522 bytes if VLAN Tagged;
1518 bytes if not VLAN Tagged

No FCS error

No collisions

10.2.1.3 FCSERRORS

Counts number of **valid frames** received with bad FCS.

Frame size: ≥ 64 bytes, ≤ 1522 bytes if VLAN Tagged;
1518 bytes if not VLAN Tagged

No framing error

No collisions

10.2.1.4 ALIGNMENTERRORS

Counts number of **valid frames** received with bad alignment (not byte-aligned).

Frame size: ≥ 64 bytes, ≤ 1522 bytes if VLAN Tagged;
1518 bytes if not VLAN Tagged

No framing error

No collisions

10.2.1.5 FRAMETOOLONGS

Counts number of frames received with size exceeding the maximum allowable frame size.

Frame size:	≥ 64 bytes,	> 1522 bytes if VLAN Tagged; 1518 bytes if not VLAN Tagged
FCS error:	don't care	
Framing error:	don't care	
No collisions		

10.2.1.6 SHORTEVENTS

Counts number of frames received with size less than the length of a short event.

Frame size:	≥ 64 bytes,	< 10 bytes
FCS error:	don't care	
Framing error:	don't care	
No collisions		

10.2.1.7 RUNTS

Counts number of frames received with size under 64 bytes, but greater than the length of a short event.

Frame size:	≥ 10 bytes,	< 64 bytes
FCS error:	don't care	
Framing error:	don't care	
No collisions		

10.2.1.8 Collisions

Counts number of collision events.

Frame size:	any size

10.2.1.9 LATEEVENTS

Counts number of collision events that occurred late (after LateEventThreshold = 64 bytes).

Frame size: any size

Events are also counted by collision counter

10.2.1.10 VERYLONGEVENTS

Counts number of frames received with size larger than Jabber Lockup Protection Timer (TW3).

Frame size: > Jabber

10.2.1.11 DATARATEMISATCHES

For repeaters or HUB application only.

10.2.1.12 AUTOPARTITIONS

For repeaters or HUB application only.

10.2.1.13 TOTALERRORS

Sum of the following errors:

FCS errors

Alignment errors

Frame too long

Short events

Late events

Very long events

10.3 IEEE – 802.1 Bridge Management (RFC 1286)

10.3.1 Event Counters

10.3.1.1 INFRAMES

Counts number of frames received by this port or segment.

Note: this counter only counts a frame received by this port if and only if it is for a protocol being processed by the local bridge function.

10.3.1.2 OUTFRAMES

Counts number of frames transmitted by this port.

Note: this counter only counts a frame transmitted by this port if and only if it is for a protocol being processed by the local bridge function.

10.3.1.3 INDISCARDS

Counts number of valid frames received which were discarded (i.e., filtered) by the forwarding process.

10.3.1.4 DELAYEXCEEDEDDISCARDS

Counts number of frames discarded due to excessive transmit delay through the bridge.

10.3.1.5 MTUEXCEEDEDDISCARDS

Counts number of frames discarded due to excessive size.

10.4 RMON – Ethernet Statistic Group (RFC 1757)

10.4.1 Event Counters

10.4.1.1 DROP EVENTS

Counts number of times a packet is dropped, because of lack of available resources. DOES NOT include all packet dropping -- for example, random early drop for quality of service support.

10.4.1.2 OCTETS

Counts the total number of octets (i.e. bytes) in **any frames** received.

10.4.1.3 BROADCASTPKTS

Counts the number of **good frames** received and forwarded with broadcast address.

Does not include non-broadcast multicast frames.

10.4.1.4 MULTICASTPKTS

Counts the number of **good frames** received and forwarded with multicast address.

Does not include broadcast frames.

10.4.1.5 CRCALIGNERRORS

Frame size: $\geq$ 64 bytes, < 1522 bytes if VLAN tag (1518 if no VLAN)

No collisions:

Counts number of frames received with FCS or alignment errors

10.4.1.6 UNDERSIZEPKTS

Counts number of frames received with size less than 64 bytes.

Frame size: < 64 bytes,

No FCS error

No framing error

No collisions

10.4.1.7 OVERSIZEPKTS

Counts number of frames received with size exceeding the maximum allowable frame size.

Frame size:	>1522 bytes if VLAN tag (1518 bytes if no VLAN)
FCS error	don't care
Framing error	don't care
No collisions	

10.4.1.8 FRAGMENTS

Counts number of frames received with size less than 64 bytes and with bad FCS.

Frame size:	< 64 bytes
Framing error	don't care
No collisions	

10.4.1.9 JABBERS

Counts number of frames received with size exceeding maximum frame size and with bad FCS.

Frame size:	> 1522 bytes if VLAN tag (1518 bytes if no VLAN)
Framing error	don't care
No collisions	

10.4.1.10COLLISIONS

Counts number of collision events detected.

Only a best estimate since collisions can only be detected while in transmit mode, but not while in receive mode.

Frame size:	any size

10.4.1.11Packet Count for Different Size Groups

Six different size groups – one counter for each:

Pkts64Octets	for any packet with size = 64 bytes
Pkts65to127Octets	for any packet with size from 65 bytes to 127 bytes

Pkts128to255Octets for any packet with size from 128 bytes to 255 bytes

Pkts256to511Octets for any packet with size from 256 bytes to 511 bytes

Pkts512to1023Octets for any packet with size from 512 bytes to 1023 bytes

Pkts1024to1518Octets for any packet with size from 1024 bytes to 1518 bytes

counts both good and bad packets.

Miscellaneous Counters

In addition to the statistics groups defined in previous sections, the MVTX2802AG has other statistics counters for its own purposes. We have two counters for flow control – one counting the number of flow control frames received, and another counting the number of flow control frames sent. We also have two counters, one for unicast frames sent, and one for non-unicast frames sent. A broadcast or multicast frame qualifies as non-unicast. Furthermore, we have a counter called “frame send fail.” This keeps track of FIFO under-runs, late collisions, and collisions that have occurred 16 times.

11.0 Register Definition

11.1 MVTX2802AG Register Description

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
0. ETHERNET Port Control Registers – Substitute [N] with Port number (0..3)						
ECR1P”N”	Port Control Register 1 for Port N (N=0-3)	000 + 2N	R/W	000+2N	c0	
ECR2P”N”	Port Control Register 2 for Port N (N=0-3)	001 + 2N	R/W	001+2N	00	
ECRMISC1	Port Control Misc1	010	R/W	010	c0	
ECRMISC2	Port Control Misc 2	011	R/W	011	00	
GGCONTROL0	Extra Gigabit Port Control –port 0,1	012	R/W	N/A	00	
GGCONTROL1	Extra Gigabit Port Control –port 2,3	013	R/W	N/A	00	
ACTIVELINK	Active Link status port 3:0	016	R/W	N/A	00	
1. VLAN Control Registers – Substitute [N] with Port number (0..3, 8)						
AVTCL	VLAN Type Code Register Low	100	R/W	012	00	
AVTCH	VLAN Type Code Register High	101	R/W	013	81	
PVMAP”N”_0	Port “N” Configuration Register 0 (N=0-3, 8)	102 + 4N	R/W	014+4N	ff	
PVMAP”N”_1	Port “N” Configuration Register 1 (N=0-3, 8)	103 + 4N	R/W	015+4N	ef	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
PVMODE	VLAN Operating Mode	126	R/W	038	00	
2. TRUNK Control Registers						
TRUNK0	Trunk group 0 Member	200	R/W	NA	00	
TRUNK1	Trunk group 1 Member	201	R/W	NA	00	
TRUNK2	Trunk group 2 Member	202	R/W	NA	00	
TRUNK3	Trunk group 3 Member	203	R/W	NA	00	
SINGLE_RING	Single ring port map	204	R/W	NA		
TRUNK_RING	Trunk ring port map	205	R/W	NA		

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
TRUNK_HASH_MODE	Trunk hash mode	206	R/W	NA	00	
TRUNK0_MODE	Trunk Group 0 Mode	207	R/W	039	00	
TRUNK0_HASH0	Trunk Group 0 Hash 0, 1, 2 Destination Port	208	R/W	NA	08	
TRUNK0_HASH1	Trunk Group 0 Hash 2, 3, 4, 5 Destination Port	209	R/W	NA	82	
TRUNK0_HASH2	Trunk Group 0 Hash 5, 6, 7 Destination Port	20A	R/W	NA	20	
TRUNK0_HASH3	Trunk Group 0 Hash 8, 9, 10 Destination Port	20B	R/W	NA	08	
TRUNK0_HASH4	Trunk Group 0 Hash 10, 11, 12, 13 Destination Port	20C	R/W	NA	82	
TRUNK0_HASH5	Trunk Group 0 Hash 13, 14, 15 Destination Port	20D	R/W	NA	20	
TRUNK1_HASH0	Trunk Group 1 Hash 0, 1, 2 Destination Port	20F	R/W	NA	08	
TRUNK1_HASH1	Trunk Group 1 Hash 2, 3, 4, 5 Destination Port	210	R/W	NA	82	
TRUNK1_HASH2	Trunk Group 1 Hash 5, 6, 7 Destination Port	211	R/W	NA	20	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
TRUNK1_HASH3	Trunk Group 1 Hash 8, 9, 10 Destination Port	212	R/W	NA	08	
TRUNK1_HASH4	Trunk Group 1 Hash 10, 11, 12, 13 Destination	213	R/W	NA	82	
TRUNK1_HASH5	Trunk Group 1 Hash 13, 14, 15 Destination	214	R/W	NA	20	
TRUNK2_HASH0	Trunk Group 2 Hash 0, 1, 2 Destination Port	215	R/W	NA	2c	
TRUNK2_HASH1	Trunk Group 2 Hash 2, 3, 4, 5 Destination Port	216	R/W	NA	cb	
TRUNK2_HASH2	Trunk Group 2 Hash 5, 6, 7 Destination Port	217	R/W	NA	b2	
TRUNK2_HASH3	Trunk Group 2 Hash 8, 9, 10 Destination Port	218	R/W	NA	2c	
TRUNK2_HASH4	Trunk Group 2 Hash 10, 11, 12, 13 Destination Port	219	R/W	NA	cb	
TRUNK2_HASH5	Trunk Group 2 Hash 13, 14, 15 Destination Port	21A	R/W	NA	b2	
TRUNK3_HASH0	Trunk Group 3 Hash 0, 1, 2 Destination Port	21B	R/W	NA	2c	
TRUNK3_HASH1	Trunk Group 3 Hash 2, 3, 4, 5 Destination Port	21C	R/W	NA	cb	
TRUNK3_HASH2	Trunk Group 3 Hash 5, 6, 7 Destination Port	21D	R/W	NA	b2	
TRUNK3_HASH3	Trunk Group 3 Hash 8, 9, 10 Destination Port	21E	R/W	NA	2c	
TRUNK3_HASH4	Trunk Group 3 Hash 10, 11, 12, 13 Destination Port	21F	R/W	NA	Bc	
TRUNK3_HASH5	Trunk Group 3 Hash 13, 14, 15 Destination Port	220	R/W	NA	b2	
Multicast_HASH00	Multicast hash result 0 mask bit[7:0]	221	R/W	NA	ff	
Multicast_HASH01	Multicast hash result 1 mask bit[7:0]	222	R/W	NA	ff	
Multicast_HASH02	Multicast hash result 2 mask bit[7:0]	223	R/W	NA	ff	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
Multicast_HASH03	Multicast hash result 3 mask bit[7:0]	224	R/W	NA	ff	
Multicast_HASH04	Multicast hash result 4 mask bit[7:0]	225	R/W	NA	ff	
Multicast_HASH05	Multicast hash result 5 mask bit[7:0]	226	R/W	NA	ff	
Multicast_HASH06	Multicast hash result 6 mask bit[7:0]	227	R/W	NA	ff	
Multicast_HASH07	Multicast hash result 7 mask bit[7:0]	228	R/W	NA	ff	
Multicast_HASH08	Multicast hash result 8 mask bit[7:0]	229	R/W	NA	ff	
Multicast_HASH09	Multicast hash result 9 mask bit[7:0]	22A	R/W	NA	fff	
Multicast_HASH10	Multicast hash result 10 mask bit[7:0]	22B	R/W	NA	ff	
Multicast_HASH11	Multicast hash result 11 mask bit[7:0]	22C	R/W	NA	ff	
Multicast_HASH12	Multicast hash result 12 mask bit[7:0]	22D	R/W	NA	ff	
Multicast_HASH13	Multicast hash result 13 mask bit[7:0]	22E	R/W	NA	ff	
Multicast_HASH14	Multicast hash result 14 mask bit[7:0]	22F	R/W	NA	ff	
Multicast_HASH15	Multicast hash result 15 mask bit[7:0]	230	R/W	NA	ff	
Multicast_HASHML	Multicast hash bit[8] for result 7-0	231	R/W	NA	ff	
Multicast HASHMH	Multicast hash bit[8] for result 15-8	232	R/W	NA	ff	
3. CPU Port Configuration						
MAC0	CPU MAC Address byte 0	300	R/W	NA	00	
MAC1	CPU MAC Address byte 1	301	R/W	NA	00	
MAC2	CPU MAC Address byte 2	302	R/W	NA	00	
MAC3	CPU MAC Address byte 3	303	R/W	NA	00	
MAC4	CPU MAC Address byte 4	304	R/W	NA	00	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
MAC5	CPU MAC Address byte 5	305	R/W	NA	00	
INT_MASK0	Interrupt Mask 0	306	R/W	NA	ff	
INT_MASK1	Interrupt Mask 1	307	R/W	NA	ff	
INT_MASK2	Interrupt Mask 2	308	R/W	NA	ff	
INT_MASK3	Interrupt Mask 3	309	R/W	NA	ff	
INT_STATUS0	Status of Masked Interrupt Register0	30A	RO	NA		
INT_STATUS1	Status of Masked Interrupt Register1	30B	RO	NA		
INTP_MASK”N”	Interrupt Mask for MAC Port 2n, 2n+1 (n=0-1)	30C-30F	R/W	NA	ff	
RQS	Receive Queue Select	310	R/W	NA	00	
RQSS	Receive Queue Status	311	RO	NA		
TX_AGE	Transmission Queue Aging Time	312	R/W	03B	08	
4. Search Engine Configurations						
AGETIME_LOW	MAC Address Aging Time Low	400	R/W	03C	2c	
AGETIME_HIGH	MAC Address Aging Time High	401	R/W	03D	00	
V_AGETIME	VLAN to Port Aging Time	402	R/W	NA	ff	
SE_OPMODE	Search Engine operation mode	403	R/W	NA	00	
Register	**Description**	**CPU Addr (Hex)**	**R/W**	**I2C Addr (Hex)**	**Default**	**Notes**
SCAN	Scan Control Register	404	R/W	NA	00	
5. Buffer Control and QOS Control						
FCBAT	FCB Aging Timer	500	R/W	03E	ff	
QOSC	QOS Control	501	R/W	03F	00	
FCR	Flooding Control Register	502	R/W	040	08	
AVPML	VLAN Priority Map Low	503	R/W	041	88	
AVPMM	VLAN Priority Map Middle	504	R/W	042	c6	
AVPMH	VLAN Priority Map High	505	R/W	043	fa	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
TOSPML	TOS Priority Map Low	506	R/W	044	88	
TOSPMM	TOS Priority Map Middle	507	R/W	045	c6	
TOSPMH	TOS Priority Map High	508	R/W	046	fa	
AVDM	VLAN Discard Map	509	R/W	047	00	
TOSDML	TOS Discard Map	50A	R/W	048	00	
BMRC	Broadcast/Multicast Rate Control	50B	R/W	049	00	
UCC	Unicast Congestion Control	50C	R/W	04A	07	
MCC	Multicast Congestion Control	50D	R/W	04B	48	
PR100	Port Reservation for 10/100 Ports	50E	R/W	04C	00	
PRG	Port Reservation for Giga Ports	50F	R/W	04D	26	
SFCB	Share FCB Size	510	R/W	04E	37	
C2RS	Class 2 Reserved Size	511	R/W	04F	00	
C3RS	Class 3 Reserved Size	512	R/W	050	00	
C4RS	Class 4 Reserved Size	513	R/W	051	00	
C5RS	Class 5 Reserved Size	514	R/W	052	00	
C6RS	Class 6 Reserved Size	515	R/W	053	00	
C7RS	Class 7 Reserved Size	516	R/W	054	00	
QOSC”N”	QOS Control (N=0 – 2F)	517–546	R/W	055-084		
QOSC”N”	QOS Control (N=30 – 82)	547-599	R/W	NA		
RDRC0	WRED Rate Control 0	59A	R/W	085	8e	
RDRC1	WRED Rate Control 1	59B	R/W	086	68	
6. MISC Configuration Registers						
MII_OP0	MII Register Option 0	600	R/W	0B1	00	
MII_OP1	MII Register Option 1	601	R/W	0B2	00	
FEN	Feature Registers	602	R/W	0B3	10	
MIIC0	MII Command Register 0	603	R/W	N/A	00	

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
MIIC1	MII Command Register 1	604	R/W	N/A	00	
MIIC2	MII Command Register 2	605	R/W	N/A	00	
MIIC3	MII Command Register 3	606	R/W	N/A	00	
MIID0	MII Data Register 0	607	RO	N/A	00	
MIID1	MII Data Register 1	608	RO	N/A	00	
LED	LED Control Register	609	R/W	0B4	38	
CHECKSUM	EEPROM Checksum Register	60B	R/W	0C5	00	
LEDUSER0	LED User Define Register 0	60C	R/W	0BB	00	
LEDUSER1	LED User Define Register 1	60D	R/W	0BC	00	
LEDUSER2	LED User Define Reg. 2/LED_byte pin 2	60E	R/W	0BD	80	
LEDUSER3	LED User Define Reg. 3/LED_byte pin 3	60F	R/W	0BE	33	
LEDUSER4	LED User Define Reg. 4/LED_byte pin 4	610	R/W	0BF	32	
LEDUSER5	LED User Define Reg. 5/LED_byte pin 5	611	R/W	0C0	20	
LEDUSER6	LED User Define Reg. 6/LED_byte pin 6	612	R/W	0C1	40	
LEDUSER7	LED User Define Reg. 7/LED_byte pin 1 & 0	613	R/W	0C2	61	
MIINP0	MII NEXT PAGE DATA REGISTER0	614	R/W	0C3	00	
MIINP1	MII NEXT PAGE DATA REGISTER1	615	R/W	0C4	00	
E. Test Group Control						
DTSRL	Test Register Low	E00	R/W	N/A	00	
DTSRM	Test Register Medium	E01	R/W	N/A	01	
DTSRH	Test Register High	E02	R/W	N/A	00	
TDRB0	TEST MUX read back register [7:0]	E03	RO	N/A		

Register	Description	CPU Addr (Hex)	R/W	I^2C Addr (Hex)	Default	Notes
TDRB1	TEST MUX read back register [15:8]	E04	RO	N/A		
DTCR	Test Counter Register	E05	R/W	N/A	00	
MASK0	MASK Timeout 0	E06	R/W	0B6	00	
MASK1	MASK Timeout 1	E07	R/W	0B7	00	
MASK2	MASK Timeout 2	E08	R/W	0B8	00	
MASK3	MASK Timeout 3	E09	R/W	0B9	00	
MASK4	MASK Timeout 4	E0A	R/W	0BA	00	
F. Device Configuration Register						
GCR	Global Control Register	F00	R/W	N/A	00	
DCR	Device Status and Signature Register	F01	RO	N/A		
DCR01	Gigabit Port0 Port1 Status Register	F02	RO	NA		
DCR23	Gigabit Port2 Port3 Status Register	F03	RO	NA		
DPST	Device Port Status Register	F06	R/W	N/A	00	
DTST	Data read back register	F07	RO	N/A		
PLLCR	PLL Control Register	F08	R/W	N/A		
LCLKCR	LCLK Control Register	F09	R/W	N/A		
BCLKCR	BCLK Control Register	F0A	R/W	N/A		
BSTRRB0	BOOT STRAP read back register 0	F0B	RO	N/A		
BSTRRB1	BOOT STRAP read back register 1	F0C	RO	N/A		
BSTRRB2	BOOT STRAP read back register 2	F0D	RO	N/A		
BSTRRB3	BOOT STRAP read back register 3	F0E	RO	N/A		
BSTRRB4	BOOT STRAP read back register 4	F0F	RO	N/A		
BSTRRB5	BOOT STRAP read back register 5	F10	RO	N/A		
DA	DA Register	FFF	RO	N/A	DA	

Note 1. se = Search Engine
2. fe = Frame Engine
3. pgs = Port Group01, 23, 45, and 67
4. mc = MAC Control
5. tm = timer

11.2 Directly Accessed Registers

11.2.1 INDEX_REG0

- Address bits [7:0] for indirectly accessed register addresses
- Address = 0 (write only)

11.2.2 INDEX_REG1 (only needed for CPU 8-bit bus mode)

- Address bits [15:8] for indirectly accessed register addresses
- Address = 1 (write only)

11.2.3 DATA_FRAME_REG

- Data of indirectly accessed registers. (8 bits)
- Address = 2 (read/write)

11.2.4 CONTROL_FRAME_REG

- CPU transmit/receive switch frames. (8/16 bits)
- Address = 3 (read/write)
- Format: (see processor interface application note for more information)
  - Send frame from CPU: (In sequence)

    Frame Data (size should be in multiple of 8-byte)

    8-byte of Frame status (Frame size, Destination port #, Frame O.K. status)
  - CPU Received frame: (In sequence)

    8-byte of Frame status (Frame size, Source port #, VLAN tag)

    Frame Data

11.2.5 COMMAND&STATUS

- CPU interface commands (write) and status
- Address = 4 (read/write)
  - When the CPU **reads** this register:
    - Bit [0]: Transmit Control Command 1 Ready; Must read true before CPU writes new Control Command 1.
    - Bit [1]: Receive Control Command 1 Ready; Must read true before CPU reads a new Control Command 1.
    - Bit [2]: Receive Control Command 2 Ready; Must read true before CPU reads a new Control Command 2.
    - Bit [3]: Receive CPU Frame Ready; Must read true before receiving a CPU frame and at every 8-byte boundary within a CPU frame.
    - Bit [4]: Transmit CPU Frame Ready; Must read true before transmitting a CPU frame and at every 16-byte boundary within a CPU frame.
    - Bit [5]: End of Receive CPU Frame to indicate that the last 8 bytes need to be read.
    - Bit [15:6]: Reserved.

- When the CPU **writes** to this register:
  - Bit [0]: End of Transmit Control Command indicator; Set after CPU writes a Control Command Frame into Rx buffer.
  - Bit [1]: End of Receive Control Command 1 indicator; Set after CPU reads out a Control Command 1 Frame from Tx buffer 1.
  - Bit [2]: End of Receive Control Command 2 indicator; Set after CPU reads out a Control Command 2 Frame from Tx buffer 2.
  - Bit [3]: End of Receive CPU Frame indicator. Set after CPU reads out a CPU frame or to flush out the rest of CPU frame.
  - Bit [4]: End of Transmit CPU Frame indicator. Set before writing the last byte of CPU frame.
  - Bit [7:5]: Reserved and always write 0's.
  - Bit [15:8]: Reserved and write 0's in 16-bit mode.

11.2.6 Interrupt Register

- Interrupt sources (8 bits)
- Address = 5 (read only)
- When CPU **reads** this register

Bit [0]: • CPU frame interrupt

Bit [1]: • Control Frame 1 interrupt. Control Frame receive buffer1 has data for CPU to read

Bit [2]: • Control Frame 2 interrupt. Control Frame receive buffer2 has data for CPU to read

Bit [3] • From any of the gigabit port interrupt

Bit [7:4]: • Reserve

Note: This register is not self-cleared. After reading CPU has to clear the bit writing 0 to it

11.2.7 Control Frame Buffer1 Access Register

- Address = 6 (read/write)
- When CPU writes to this register, data is written to the Control Command Frame Receive Buffer
- When CPU reads this register, data is read from the Control Command Frame Transmit Buffer1

11.2.8 Control Frame Buffer2 Access Register

- Address = 7 (read only)
- When CPU reads this register, data is read from the Control Command Frame transmit Buffer 2

Indirectly Accessed Registers

11.3 Group 0 Address

11.3.1 MAC Ports Group

11.3.1.1 ECR1PN: PORT N CONTROL REGISTER

- I^2C Address h00+2n; CPU Address:h000+2n (n=0 to 3)
- Accessed by CPU, serial interface and I^2C (R/W)

7 6	5 4 3 2 1 0
Sp State	A-FC	Port Mode

Bit [4:0]
- Port Mode **(Default 2'b00)**

Bit [4:3]
- 00 - Automatic Enable Auto-Negotiation – This enables hardware state machine for auto-negotiation.
- 01 - Limited Disable auto-Negotiation – This disables hardware auto-negotiation. Hardware only Polls MII for link status. Use bit [2:0] for config.
- 10 - Link Down - Force link down (disable the port). Does not talk to PHY.
- 11 - Link Up – Does not talk to PHY. User ERC1 [2:0] for config.

Bit [2]
- 1 – 10Mbps **(Default 1'b0)**
- 0 – 100Mbps

Bit 2 is used only when the port is in MII mode.

Bit [1]
- 1 – Half Duplex (Do not use) **(Default 1'b0)**
- 0 – Full Duplex

Bit [0]
- 1 – Flow Control Off **(Default 1'b0)**
- 0 – Flow Control On

- When flow control is on:
- In full duplex mode, the MAC transmitter sends Flow Control Frames when necessary. The MAC receiver interprets and processes incoming flow control frames. The Flow Control Frame Received counter is incremented whenever a flow control frame is received.

- When flow control is off:
- In full duplex mode, the MAC transmitter does not send flow control frames. The MAC receiver does not interpret or process the flow control frames. The Flow Control Frame Receiver counter is not incremented.

Bit [5]
- Asymmetric Flow Control Enable.
  - 0 – Disable asymmetric flow control
  - 1 – Enable asymmetric flow control
- When this bit is set, and flow control is on (bit[0] = 0), don't send out a flow control frame. But MAC receiver interprets and process flow control frames. (**Default is 0)**

Bit [7:6]
- SS - Spanning tree state (802.1D spanning tree protocol). **(Default 2'b11)**
  - 00 – Blocking: Frame is dropped
  - 01 - Listening: Frame is dropped
  - 10 - Learning: Frame is dropped. Source MAC address is learned.
  - 11 - Forwarding: Frame is forwarded. Source MAC address is learned.

11.3.1.2 ECR2PN: PORT N CONTROL REGISTER

- I^2C Address: 01+2n; CPU Address:h001+2n (n=0 to 3)
- Accessed by CPU and serial interface (R/W)

7 6	5 3	2	1	0
Security En		DisL	Ftf	Futf

Bit[0]:
- Filter untagged frame **(Default 0)**
  - 0: Disable
  - 1: Enable – All untagged frames from this port are discarded or follow security option when security is enable

Bit[1]:
- Filter Tag frame **(Default 0)**
  - 0: Disable
  - 1: Enable - All tagged frames from this port are discarded or follow security option when security is enable

Bit[2]:
- Learning Disable **(Default 0)**
  - 0: Learning is enabled on this port
  - 1: Learning is disabled on this port

Bit [5:3:]
- Reserved

Bit[7:6]
- Security Enable (Default 00). The MVTX2802AG checks the incoming data for one of the following conditions:

  1. If the source MAC address of the incoming packet is in the MAC table and is defined as secure address but the ingress port is not the same as the port associated with the MAC address in the MAC table.

  A MAC address is defined as secure when its entry at MAC table has static status and bit 0 is set to 1. MAC address bit 0 (the first bit transmitted) indicates whether the address is unicast or multicast. As source addresses are always unicast bit 0 is not used (always 0). MVTX2802 uses this bit to define secure MAC addresses.

  2. If the port is set as learning disable and the source MAC address of the incoming packet is not defined in the MAC address table.

  3. If the port is configured to filter untagged frames and an untagged frame arrives or if the port is configured to filter tagged frames and a tagged frame arrives.

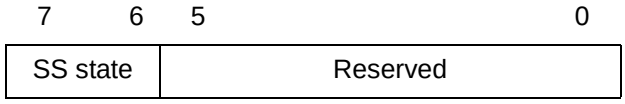
If one of these three conditions occurs, the packet will be handled according to one of the following specified options:

- CPU installed
  - 00 – Disable port security
  - 01 – Discard violating packets
  - 10 – Send packet to CPU and destination port
  - 11 – Send packet to CPU only

- CPU not installed
  - 00 – Disable port security
  - 01 – Enable port security. Port will be disabled when security violation is detected
  - 10 – N/A
  - 11 – N/A

11.3.1.3 ECRMISC1 – CPU Port Control Register MISC1

- I^2C Address h10, CPU Address:h010
- Access by CPU, serial interface and I^2C (R/W)

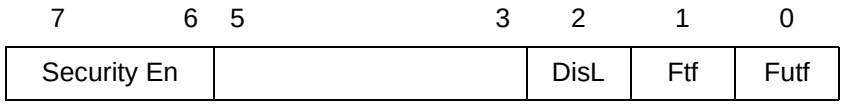


Bit [5:0] • Reserved

Bit [7:6] • SS - Spanning tree state (802.1D spanning tree protocol). **(Default 2'b11)**
- 00 – Blocking: Frame is dropped
- 01 - Listening: Frame is dropped
- 10 - Learning: Frame is dropped. Source MAC address is learned.
- 11 - Forwarding: Frame is forwarded. Source MAC address is learned.

11.3.1.4 ECRMISC2 – CPU PORT CONTROL REGISTER MISC2

- I^2C Address h11, CPU Address:h011
- Access by CPU, serial interface and I^2C (R/W)



Bit [0] • Filter untagged frame **(Default 0)**
- 0: Disable
- 1: Enable – All untagged frames from the CPU are discarded or follow security option when security is enable Security does not make much sense for CPU!

Bit[1] • Filter Tagged frame **(Default 0)**
- 0: Disable
- 1: Enable – All tagged frames from the CPU are discarded or follow security option when security is enable Security does not make much sense for CPU!

Bit[2] • Learning Disable **(Default 0)**
- 1 – Learning is disabled on this port
- 0 – Learning is enabled on this port

Bit [5:3] • Reserved **(Default 0)**

Bit[7:6] • Security Enable **(Default 2'b00)**
- CPU installed
  - 00 – Disable port security
  - 01 – Discard violation packet
  - 10 – Send packet to CPU and port
  - 11 – Send packet to CPU only

11.3.1.5 GGCONTROL 0– EXTRA GIGA PORT CONTROL

- CPU Address:h012
- Accessed by CPU and serial interface (R/W)

7 6	5	4	3 2	1	0
	MII1	Rst1		MII0	Rst0

Bit[0]: • Reset GIGA port 0 **Default is 0**
- 0: Normal operation
- 1: Reset Gigabit port 0. Example: used when a new Phy is connected (Hot swap)

Bit[1]: • GIGA port 0 use MII interface (10/100M) **Default is 0**
- 0: Gigabit port operation at 1000M mode
- 1: Gigabit port operation at 10/100M mode (MII)

Bit[3:2]: • Reserved - Must be '0'

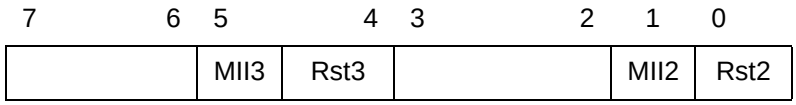
Bit[4]: • Reset GIGA port 1 **Default is 0**
- 0: Normal operation
- 1: Reset Gigabit port 1. Example: used when a new Phy is connected (Hot swap)

Bit[5]: • GIGA port 1 use MII interface (10/100M) **Default is 0**
- 0: Gigabit port operation at 1000M mode
- 1: Gigabit port operation at 10/100M mode (MII)

Bit[7:6]: • Reserved - Must be '0'

11.3.1.6 GGCONTROL 1– EXTRA GIGA PORT CONTROL

- CPU Address:h013
- Accessed by CPU and serial interface (R/W)



Bit[0]:
- Reset GIGA port 2 **Default is 0**
  - 0: Normal operation
  - 1: Reset Gigabit port 2. Example: used when a new Phy is connected (Hot swap)

Bit[1]:
- GIGA port 2 use MII interface (10/100M) **Default is 0**
  - 0: Gigabit port operation at 1000M mode
  - 1: Gigabit port operation at 10/100M mode (MII)

Bit[3:2]:
- Reserved - must be '0'

Bit[4]:
- Reset GIGA port 3 **Default is 0**
  - 0: Normal operation
  - 1: Reset Gigabit port 3. Example: used when a new Phy is connected (Hot swap)

Bit[5]:
- GIGA port 3 use MII interface (10/100M) **Default is 0**
  - 0: Gigabit port operation at 1000M mode
  - 1: Gigabit port operation at 10/100M mode (MII)

Bit[7:6]:
- Reserved - Must be '0'

11.4 Group 1 Address

11.4.1 VLAN Group

11.4.1.1 AVTCL – VLAN TYPE CODE REGISTER LOW

- I^2C Address h12; CPU Address:h100
- Accessed by CPU, serial interface and I^2C (R/W)

Bit[7:0]: VLANType_LOW: Lower 8 bits of the VLAN type code (Default 00)

11.4.1.2 AVTCH – VLAN TYPE CODE REGISTER HIGH

- I^2C Address h13; CPU Address:h101
- Accessed by CPU, serial interface and I^2C (R/W)
- Bit [7:0] VLANType_HIGH: Upper 8 bits of the VLAN type code (Default is 81)

11.4.1.3 PVMAP00_0 – PORT 00 CONFIGURATION REGISTER 0

- I^2C Address h14, CPU Address:h102)
- Accessed by CPU, serial interface and I^2C (R/W)

In Port Based VLAN Mode

This register indicates the legal egress ports. Example: A “1” on bit 3 means that packets arriving on port 0 can be sent to port 3. A “0” on bit 7 means that any packet destined to port 3 will be discarded.

Bit[3:0]:
- VLAN Mask for ports 3 to 0 (Default F)
  - 0 – Disable
  - 1 - Enable

In Tag Based VLAN Mode

This is the default VLAN tag. It works with configuration register PVMAP00_1 [7:5] [3:0] to form the default VLAN tag. If the received packed is untagged, it receives the default VLAN tag. If the packet has a VLAN ID of 0, then PVID is used to replace the packet’s VLAN.

Bit[3:0]: PVID [3:0] (Default is F)

11.4.1.4 PVMAP00_1 – PORT 00 CONFIGURATION REGISTER 1

- I^2C Address h15, CPU Address:h103
- Accessed by CPU, serial interface and I^2C (R/W)

In Port Based VLAN Mode

Bit[7:0]: VLAN Mask for port 8 – CPU port (Default is FF)

In Tag Based VLAN Mode

7 5	4	3 0
Unitag Port Priority	Ultrust	PVID

Bit[3:0]:
- PVID [11:8] (Default is F)

Bit [4]: • Untrusted Port. (Default is 0)

This register is used to change the VLAN priority field of a packet to a predetermined priority.

- 1: VLAN priority field is changed to Bit[7:5] at ingress port
- 0: Keep VLAN priority field

Bit [7:5]: • Untag Port Priority (Default 7)

11.4.1.5 PVMAP00_3 – PORT 00 CONFIGURATION REGISTER 3

- I^2C Address h17, CPU Address:h105)
- Accessed by CPU, serial interface and I^2C (R/W)

In Port Based Mode

7	6	5 3	2	1 0
FP en	Drop	Default TX priority	FNT	Reserved

Bit [1:0]: • Reserved (Default 0)

Bit [2]: • Force untagged out (Default 0)

- 0 Disable
- 1 Force untag output

All packets transmitted from this port are untagged. This register is used when this port is connected to legacy equipment that does not support VLAN tagging.

Bit [5:3]: • Fixed Transmit priority. Used when bit[7] = 1 **(Default 0)**

- 000 Transmit Priority Level 0 (Lowest)
- 001 Transmit Priority Level 1
- 010 Transmit Priority Level 2
- 011 Transmit Priority Level 3
- 100 Transmit Priority Level 4
- 101 Transmit Priority Level 5
- 110 Transmit Priority Level 6
- 111 Transmit Priority Level 7 (Highest)

Bit [6]: • Fixed Discard priority **(Default 0)**

- 0 – Discard Priority Level 0 (Lowest)
- 1 – Discard Priority Level 7(Highest)

Bit [7]: • Enable Fix Priority **(Default 0)**

- 0 Disable fix priority. All frames are analyzed. Transmit Priority and Drop Priority are based on VLAN Tag or TOS.
- 1 Transmit Priority and Discard Priority are based on values programmed in bit [6:3]

In Tag based VLAN Mode

Bit [1]:
- Ingress filter enable **(Default 1)**
  - 0 Disable – Ingress filter. Packets with VLAN not belonging to source port are forwarded if destination port belongs to the VLAN. Symmetric VLAN.
  - 1 Enable – Packets are discarded when source port is not a VLAN member. Asymmetric VLAN.

Bit [2]:
- Force untagged out **(Default 1).**
  - 0 Disable
  - 1 Force untagged output.

All packets transmitted from this port are untagged. This register is used when this port is connected to legacy equipment that does not support VLAN tagging.

Bit [5:3]:
- Fixed Transmit priority **(Default 0)** Used When Bit [7] = 1
  - 000 Transmit Priority Level 0 (Lowest)
  - 001 Transmit Priority Level 1
  - 010 Transmit Priority Level 2
  - 011 Transmit Priority Level 3
  - 100 Transmit Priority Level 4
  - 101 Transmit Priority Level 5
  - 110 Transmit Priority Level 6
  - 111 Transmit Priority Level 7 (Highest)

Bit [6]:
- Fixed Discard priority **(Default 0)** Used When Bit [7] = 1
  - 0 - Discard Priority Level 0 (Lowest)
  - 1 Discard Priority Level 1 (Highest)

Bit [7]:
- Enable Fix Priority **(Default 0)**
  - 0 Disable fix priority. All frames are analyzed. Transmit Priority and Drop Priority are based on VLAN Tag or TOS.
  - 1 Transmit Priority and Discard Priority are based on values programmed in bit [6:3]

11.5 Port VLAN Map

PVMAP00_0,1,3 I^2C Address h14,15,17; CPU Address:h102,103,105)

PVMAP01_0,1,3 I^2C Address h18,19,1B; CPU Address:h106,107,109)

PVMAP02_0,1,3 I^2C Address h1C,1D,1F; CPU Address:h10A, 10B,10D)

PVMAP03_0,1,3 I^2C Address h20,21,23; CPU Address:h10E, 10F,111)

PVMAP08_0,1,3 I^2C Address h34,35,37; CPU Address:h122, 123, 125) (CPU port)

11.5.1 PVMODE

- I^2C Address: h038, CPU Address:h126
- Accessed by CPU, serial interface (R/W)

7	6	5	4	3 ... 1	0
RO	MP	BPDU	DM	Reserved	Vmod

Bit [0]:
- VLAN Mode (vlan_enable) (Default = 0)
  - 1: Tag Based VLAN Mode
  - 0: Port Based VLAN Mode

Bit [4]:
- Disable MAC address 0
  - 0: MAC address 0 is not leaned.
  - 1: MAC address 0 is leaned.

Bit [5]:
- Force BPDU as multicast frame (**Default 0**)
  - 1: Enable.
  - 0: Disable. BPDU packet is forwarded to CPU.

Bit [6]:
- MAC/PORT
  - 0: Single MAC address per system
  - 1: Single MAC address per port

Bit [7]:
- Routing option (force frame as switched frame)
  - 1: Routing Frame to CPU is independent of ingress port spanning tree state
  - 0: Routing Frame to CPU is dependent of ingress port spanning tree state

11.6 Group 2 Address

11.6.1 Port Trunking Group

11.6.1.1 TRUNK0 – Trunk group 0 Member (Managed Mode Only)

- CPU Address:h200
- Accessed by CPU, serial interface (R/W)
- Bit [3:0] Port3-0 bit map of trunk 0. (**Default 00**)

TRUNK0 provides a bitmap for trunk0 membership. Example: To trunk ports 0 and 2 in trunk group 0, bits 0 and 2 of TRUNK0 must be set to 1. All others must be cleared to “0” to indicate that they are not members of the trunk 0.

11.6.1.2 TRUNK1 – Trunk group 1 Member (Managed Mode Only)

- CPU Address:h201
- Accessed by CPU, serial interface (R/W)
- Bit [3:0] Port3-0 bit map of trunk 1. (**Default 00**)

11.6.1.3 TRUNK2– TRUNK GROUP 2 MEMBER (MANAGED MODE ONLY)

- CPU Address:h202
- Accessed by CPU, serial interface (R/W)
- Bit [3:0] Port3-0 bit map of trunk 2. (**Default 00**)

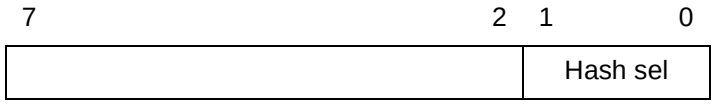
11.6.1.4 TRUNK3– TRUNK GROUP 3 MEMBER (MANAGED MODE ONLY)

- CPU Address:h203
- Accessed by CPU, serial interface (R/W)
- Bit [3:0] Port3-0 bit map of trunk 3. (**Default 00**)

11.6.1.5 TRUNK_HASH_MODE – TRUNK HASH MODE

- CPU Address:h206
- Accessed by CPU, serial interface (R/W)

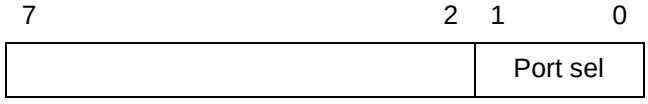
Hash Select. The hash selected is valid for Trunk 0, 1, 2 and 3.



Bit [1:0]:
- (Default 2'b00)
  - 00 – Use Source and Destination Mac address for hashing.
  - 01 – Use Source Mac Address for hashing.
  - 10 – Use Destination Mac Address for hashing.
  - 11 – Not Used.

11.6.1.6 TRUNK0_MODE – TRUNK GROUP 0 MODE (UNMANAGED MODE)

- I^2C Address: h039, CPU Address:h207
- Accessed by serial interface and I^2C (R/W)
- Port Selection in unmanaged mode. Trunk group 0 and trunk group 1 are enable accordingly to bits [1:0] when input pin P_d[9] = 0 (external pull down).



Bit [1:0]:
- Port member selection for Trunk 0 and 1 in unmanaged mode (Default 2'b00)
  - 00 – Only trunk group 0 is enable. Port 0 and 1 are used for trunk group0
  - 01 – Only trunk group 0 is enable. Port 0,1 and 2 are used for trunk group0
  - 10 – Only trunk group 0 is enable. Port 0,1,2 and 3 are used for trunk group0
  - 11 – Trunk group 0 and 1 are enable. Port 0, 1 are used for trunk group0, and port 2 and 3 are used for trunk group1

TRUNK HASH

- Trunk group 0 achieve load balance by TRUNK0_HASH0 to 5. (only in managed mode)
- Trunk group 1 achieve load balance by TRUNK1_HASH0 to 5. (only in managed mode)
- Trunk group 2 achieve load balance by TRUNK2_HASH0 to 5. (only in managed mode)
- Trunk group 3 achieve load balance by TRUNK3_HASH0 to 5. (only in managed mode)

11.6.1.7 TRUNK0_HASH0 – TRUNK GROUP 0 HASH RESULT 0,1,2 DESTINATION PORT NUMBER

- CPU Address:h208
- Accessed by CPU, serial interface (R/W)

Bit [2:0]: • Hash result 0 destination port number[2:0] **(Default 000)**

Bit[5:3] • Hash result 1 destination port number[2:0] **(Default 001)**

Bit[7:6] • Hash result 2 destination port number[1:0] **(Default 00)**

11.6.1.8 TRUNK0_HASH1 – TRUNK GROUP 0 HASH RESULT 2,3,4,5 DESTINATION PORT NUMBER

- CPU Address:h209
- Accessed by CPU, serial interface (R/W)

Bit [0]: • Hash result 2 destination port number[2] **(Default 0)**

Bit[3:1] • Hash result 3 destination port number[2:0] **(Default 001)**

Bit[6:4] • Hash result 4 destination port number[2:0] **(Default 000)**

Bit[7] • Hash result 5 destination port number[0] **(Default 1)**

11.6.1.9 TRUNK0_HASH2 – TRUNK GROUP 0 HASH RESULT 5,6,7 DESTINATION PORT NUMBER

- CPU Address:h20A
- Accessed by CPU, serial interface (R/W)

Bit [1:0]: • Hash result 5 destination port number[2:1] **(Default 00)**

Bit[4:2] • Hash result 6 destination port number[2:0] **(Default 000)**

Bit[7:5] • Hash result 7 destination port number[2:0] **(Default 001)**

11.6.1.10TRUNK0_HASH3 – TRUNK GROUP 0 HASH RESULT 8,9,10 DESTINATION PORT NUMBER

- CPU Address:h20B
- Accessed by CPU, serial interface (R/W)

Bit [2:0]:	•	Hash result 8 destination port number[2:0] **(Default 000)**
Bit[5:3]	•	Hash result 9 destination port number[2:0] **(Default 001)**
Bit[7:6]	•	Hash result 10 destination port number[1:0] **(Default 00)**

11.6.1.11TRUNK0_HASH4 – TRUNK GROUP 0 HASH RESULT 10,11,12,13 DESTINATION PORT NUMBER

- CPU Address:h20C
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 10 destination port number[2] **(Default 0)**
Bit[3:1]	•	Hash result 11 destination port number[2:0] **(Default 001)**
Bit[6:4]	•	Hash result 12 destination port number[2:0] **(Default (000)**
Bit[7]	•	Hash result 13 destination port number[2:0] **(Default (1)**

11.6.1.12TRUNK0_HASH5 – TRUNK GROUP 0 HASH RESULT 13,14,15 DESTINATION PORT NUMBER

- CPU Address:h20D
- Accessed by CPU, serial interface (R/W)

Bit [1:0]:	•	Hash result 13 destination port number[2:1] **(Default 00)**
Bit[4:2]	•	Hash result 14 destination port number[2:0] **(Default 000)**
Bit[7:5]	•	Hash result 15 destination port number[2:0] **(Default 001)**

11.6.1.13TRUNK1_HASH0 – TRUNK GROUP 1 HASH RESULT 0, 1, 2 DESTINATION PORT NUMBER

- CPU Address:h20F
- Accessed by CPU, serial interface (R/W)

Bit [2:0]:	•	Hash result 0 destination port number[2:0] **(Default 000)**
Bit[5:3]	•	Hash result 1 destination port number[2:0] **(Default 001)**
Bit[7:6]	•	Hash result 2 destination port number[1:0] **(Default 00)**

11.6.1.14TRUNK1_HASH1 – TRUNK GROUP 1 HASH RESULT 2, 3, 4, 5 DESTINATION PORT NUMBER

- CPU Address:h210
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 2 destination port number[2] **(Default 0)**
Bit[3:1]	•	Hash result 3 destination port number[2:0] **(Default 001)**
Bit[6:4]	•	Hash result 4 destination port number[2:0] **(Default 000)**
Bit[7]	•	Hash result 5 destination port number[0] **(Default 1)**

11.6.1.15TRUNK1_HASH2 – TRUNK GROUP 1 HASH RESULT 5, 6, 7 DESTINATION PORT NUMBER

- CPU Address:h211
- Accessed by CPU, serial interface (R/W)

Bit [1:0]:	•	Hash result 5 destination port number[2:1] **(Default 00)**
Bit[4:2]	•	Hash result 6 destination port number[2:0] **(Default 000)**
Bit[7:5]	•	Hash result 7 destination port number[2:0] **(Default 001)**

11.6.1.16TRUNK1_HASH3 – TRUNK GROUP 1 HASH RESULT 8, 9, 10 DESTINATION PORT NUMBER

- CPU Address:h212
- Accessed by CPU, serial interface (R/W)

Bit [2:0]:	•	Hash result 8 destination port number[2:0] **(Default 000)**
Bit[5:3]	•	Hash result 9 destination port number[2:0] **(Default 001)**
Bit[7:6]	•	Hash result 10 destination port number[1:0] **(Default 00)**

11.6.1.17TRUNK1_HASH4– TRUNK GROUP 1 HASH RESULT 11, 12, 13 DESTINATION PORT NUMBER

- CPU Address:h213
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 10 destination port number[2] **(Default 0)**
Bit[3:1]	•	Hash result 11 destination port number[2:0] **(Default 001)**
Bit[6:4]	•	Hash result 12 destination port number[2:0] **(Default (000)**
Bit[7]	•	Hash result 13 destination port number[0] **(Default (1)**

11.6.1.18TRUNK1_HASH5 – TRUNK GROUP 1 HASH RESULT 13, 14, 15 DESTINATION PORT NUMBER

- CPU Address:h214
- Accessed by CPU, serial interface (R/W)

Bit [1:0]:	•	Hash result 13 destination port number[2:1] **(Default 00)**
Bit[4:2]	•	Hash result 14 destination port number[2:0] **(Default 000)**
Bit[7:5]	•	Hash result 15 destination port number[2:0] **(Default 001)**

11.6.1.19TRUNK2_HASH0 – TRUNK GROUP 2 HASH RESULT 0, 1, 2 DESTINATION PORT NUMBER

- CPU Address:h215
- Accessed by CPU, serial interface (R/W)

Bit [2:0]:	•	Hash result 0 destination port number[2:0] (Default 100)
Bit[5:3]	•	Hash result 1 destination port number[2:0] **(Default 101)**
Bit[7:6]	•	ash result 2 destination port number[1:0] **(Default 00)**

11.6.1.20TRUNK2_HASH1 – TRUNK GROUP 2 HASH RESULT 2, 3, 4, 5 DESTINATION PORT NUMBER

- CPU Address:h216
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 2 destination port number[2] **(Default 1)**
Bit[3:1]	•	Hash result 3 destination port number[2:0] **(Default 101)**
Bit[6:4]	•	Hash result 4 destination port number[2:0] **(Default 100)**
Bit[7]	•	Hash result 5 destination port number[0] **(Default 1)**

11.6.1.21TRUNK2_HASH2 – TRUNK GROUP 2 HASH RESULT 5, 6, 7 DESTINATION PORT NUMBER

- CPU Address:h217
- Accessed by CPU, serial interface (R/W)

Bit [1:0]:	•	Hash result 5 destination port number[2:1] **(Default 10)**
Bit[4:2]	•	Hash result 6 destination port number[2:0] **(Default 100)**
Bit[7:5]	•	Hash result 7 destination port number[2:0] **(Default 101)**

11.6.1.22TRUNK2_HASH3 – TRUNK GROUP 2 HASH RESULT 8, 9, 10 DESTINATION PORT NUMBER

- CPU Address:h218
- Accessed by CPU, serial interface (R/W)

Bit [2:0]: • Hash result 8 destination port number[2:0] **(Default 000)**

Bit[5:3] • Hash result 9 destination port number[2:0] **(Default 001)**

Bit[7:6] • Hash result 10 destination port number[1:0] **(Default 00)**

11.6.1.23TRUNK2_HASH4 – TRUNK GROUP 2 HASH RESULT 10, 11, 12, 13 DESTINATION PORT NUMBER

- CPU Address:h219
- Accessed by CPU, serial interface (R/W)

Bit [0]: • Hash result 10 destination port number[2] **(Default 1)**

Bit[3:1] • Hash result 11 destination port number[2:0] **(Default 101)**

Bit[6:4] • Hash result 12 destination port number[2:0] **(Default 1000)**

Bit[7] • Hash result 13 destination port number[2:0] **(Default (1)**

11.6.1.24TRUNK2_HASH5 – TRUNK GROUP 2 HASH RESULT 13, 14, 15 DESTINATION PORT NUMBER

- CPU Address:h21A
- Accessed by CPU, serial interface (R/W)

Bit [1:0]: • Hash result 13 destination port number[2:1] **(Default 10)**

Bit[4:2] • Hash result 14 destination port number[2:0] **(Default 100)**

Bit[7:5] • Hash result 15 destination port number[2:0] **(Default 101)**

11.6.1.25TRUNK3_HASH0 – TRUNK GROUP 3 HASH RESULT 0, 1, 2 DESTINATION PORT NUMBER

- CPU Address:h21B
- Accessed by CPU, serial interface (R/W)

Bit [2:0]: • Hash result 0 destination port number[2:0] (Default 100)

Bit[5:3] • Hash result 1 destination port number[2:0] **(Default 101)**

Bit[7:6] • Hash result 2 destination port number[1:0] **(Default 00)**

11.6.1.26TRUNK3_HASH1 – TRUNK GROUP 3 HASH RESULT 2, 3, 4, 5 DESTINATION PORT NUMBER

- CPU Address:h21C
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 2 destination port number[2] **(Default 1)**
Bit[3:1]	•	Hash result 3 destination port number[2:0] **(Default 101)**
Bit[6:4]	•	Hash result 4 destination port number[2:0] **(Default 100)**
Bit[7]	•	Hash result 5 destination port number[0] **(Default 1)**

11.6.1.27TRUNK3_HASH2 – TRUNK GROUP 3 HASH RESULT 5, 6, 7 DESTINATION PORT NUMBER

- CPU Address:h21D
- Accessed by CPU, serial interface (R/W)

Bit [1:0]:	•	Hash result 5 destination port number[2:1] **(Default 10)**
Bit[4:2]	•	Hash result 6 destination port number[2:0] **(Default 100)**
Bit[7:5]	•	Hash result 7 destination port number[2:0] **(Default 101)**

11.6.1.28TRUNK3_HASH3 – TRUNK GROUP 3 HASH RESULT 8, 9, 10 DESTINATION PORT NUMBER

- CPU Address:h21E
- Accessed by CPU, serial interface (R/W)

Bit [2:0]:	•	Hash result 8 destination port number[2:0] **(Default 100)**
Bit[5:3]	•	Hash result 9 destination port number[2:0] **(Default 101)**
Bit[7:6]	•	Hash result 10 destination port number[1:0] **(Default 00)**

11.6.1.29TRUNK3_HASH4 – TRUNK GROUP 3 HASH RESULT 10, 11, 12, 13 DESTINATION PORT NUMBER

- CPU Address:h21F
- Accessed by CPU, serial interface (R/W)

Bit [0]:	•	Hash result 10 destination port number[2] **(Default 1)**
Bit[3:1]	•	Hash result 11 destination port number[2:0] **(Default 101)**
Bit[6:4]	•	Hash result 12 destination port number[2:0] **(Default (100)**
Bit[7]	•	Hash result 13 destination port number[2:0] **(Default (1)**

11.6.1.30TRUNK3_HASH5 – TRUNK GROUP 3 HASH RESULT 13, 14, 15 DESTINATION PORT NUMBER

- CPU Address:h220
- Accessed by CPU, serial interface (R/W)

Bit [1:0]: • Hash result 13 destination port number[2:1] **(Default 10)**

Bit[4:2] • Hash result 14 destination port number[2:0] **(Default 100)**

Bit[7:5] • Hash result 15 destination port number[2:0] **(Default 101)**

11.6.2 Multicast Hash Registers

Multicast Hash registers are used to distribute multicast traffic. 16 + 2 registers are used to form a 16-entry array; each entry has 9 bits, with each bit representing one port. Any port not belonging to a trunk group should be programmed with 1. Ports belonging to the same trunk group should only have a single port set to “1” per entry. The port set to “1” is picked to transmit the multicast frame when the hash value is met.

Bit	8	7	6	5	4	3	2	1	0
Hash Result = 0									
Hash Result = 1									
Hash Result = 2									
...									
Hash Result = 13									
Hash Result = 14									
Hash Result = 15									
	CPU Port	nu	nu	nu	nu	Port 3	Port 2	Port 1	Port 0

11.6.2.1 MULTICAST_HASH00 – MULTICAST HASH RESULT0 MASK BYTE [7:0]

- CPU Address:h221
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.2 MULTICAST_HASH01 – MULTICAST HASH RESULT1 MASK BYTE [7:0]

- CPU Address:h222
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.3 MULTICAST_HASH02 – MULTICAST HASH RESULT2 MASK BYTE [7:0]

- CPU Address:h223
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.4 MULTICAST_HASH03 – MULTICAST HASH RESULT3 MASK BYTE [7:0]

- CPU Address:h224
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.5 MULTICAST_HASH04 – MULTICAST HASH RESULT4 MASK BYTE [7:0]

- CPU Address:h225
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.6 MULTICAST_HASH05 – MULTICAST HASH RESULT5 MASK BYTE [7:0]

- CPU Address:h226
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.7 MULTICAST_HASH06 – MULTICAST HASH RESULT6 MASK BYTE [7:0]

- CPU Address:h227
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.8 MULTICAST_HASH07 – MULTICAST HASH RESULT7 MASK BYTE [7:0]

- CPU Address:h228
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.9 MULTICAST_HASH08 – MULTICAST HASH RESULT8 MASK BYTE [7:0]

- CPU Address:h229
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.10 MULTICAST_HASH09 – MULTICAST HASH RESULT9 MASK BYTE [7:0]

- CPU Address:h22A
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.11 MULTICAST_HASH10 – MULTICAST HASH RESULT10 MASK BYTE [7:0]

- CPU Address:h22B
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.12 MULTICAST_HASH11 – MULTICAST HASH RESULT11 MASK BYTE [7:0]

- CPU Address:h22C
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.13 MULTICAST_HASH12 – MULTICAST HASH RESULT12 MASK BYTE [7:0]

- CPU Address:h22D
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.14 Multicast_HASH13 – Multicast hash result13 mask byte [7:0]

- CPU Address:h22E
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.15 MULTICAST_HASH14 – MULTICAST HASH RESULT14 MASK BYTE [7:0]

- CPU Address:h22F
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.16 MULTICAST_HASH15 – MULTICAST HASH RESULT15 MASK BYTE [7:0]

- CPU Address:h230
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.17 MULTICAST_HASHML – MULTICAST HASH BIT[8] FOR RESULT7-0

- CPU Address:h231
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.6.2.18 MULTICAST_HASHML – MULTICAST HASH BIT[8] FOR RESULT 15-8

- CPU Address:h232
- Accessed by CPU, serial interface (R/W)
- Bit [7:0] (Default FF)

11.7 Group 3 Address

11.7.1 CPU Port Configuration Group

- MAC5 to MAC0 registers form the CPU address. When a packet with destination address equal to MAC5[5:0] arrives, it is forwarded to the CPU.

(MC bit)

MAC5	MAC4	MAC3	MAC2	MAC1	MAC0

11.7.1.1 MAC0 – CPU MAC ADDRESS BYTE 0

- CPU Address:h300
- Accessed by CPU
- Bit [7:0] Byte 0 of the CPU MAC address. (Default 8’00)

11.7.1.2 MAC1 – CPU MAC ADDRESS BYTE 1

- CPU Address:h301
- Accessed by CPU
- Bit [7:0] Byte 1 of the CPU MAC address. (Default 8’00)

11.7.1.3 MAC2 – CPU MAC ADDRESS BYTE 2

- CPU Address:h302
- Accessed by CPU
- Bit [7:0] Byte 2 of the CPU MAC address. (Default 8’00)

11.7.1.4 MAC3 – CPU Mac address byte 3

- CPU Address:h303
- Accessed by CPU
- Bit [7:0] Byte 3 of the CPU MAC address. (Default 8'00)

11.7.1.5 MAC4 – CPU Mac address byte 4

- CPU Address:h304
- Accessed by CPU
- Bit [7:0] Byte 4 of the CPU MAC address. (Default 8'00)

11.7.1.6 MAC5 – CPU MAC ADDRESS BYTE 5

- CPU Address:h305
- Accessed by CPU
- Bit [7:0] Byte 5 of the CPU MAC address. (Default 8'00). These registers form the CPU MAC address

11.7.1.7 INT_MASK0 – INTERRUPT MASK 0

- CPU Address:h306
- Accessed by CPU, serial interface (R/W)
- Mask off the interrupt source

The CPU can dynamically mask the interruption when it is busy and doesn't want to be interrupted

Bit [0]: • CPU frame interrupt. CPU frame buffer has data for CPU to read (Default 1'b1)

Bit [1]: • Control Command Frame 1 interrupt. Control Command Frame buffer1 has data for CPU to read (Default 1'b1)

Bit [2]: • Control Command Frame 2 interrupt. Control Command Frame buffer2 has data for CPU to read (Default 1'b1)

Bit [7:3]: • Reserved

1 – Mask the interrupt
0 – Unmask the interrupt (Enable interrupt)

11.7.1.8 INT_MASK1 – INTERRUPT MASK 1

- CPU Address:h307
- Accessed by CPU, serial interface (R/W)

Mark off the interrupt source

Bit [0]: • From Gigabit port 0 interrupt (Default 1'b1)

Bit [1]: • From Gigabit port 1 interrupt (Default 1'b1)

Bit [2]: • From Gigabit port 2 interrupt (Default 1'b1)

Bit [3]: • From Gigabit port 3 interrupt (Default 1'b1)

Bit [4]: • From Gigabit port 4 interrupt (Default 1'b1)

Bit [5]: • From Gigabit port 5 interrupt (Default 1'b1)

Bit [6]: • From Gigabit port 6 interrupt (Default 1'b1)

Bit [7]: • From Gigabit port 7 interrupt (Default 1'b1)

- 1 – Mask the interrupt
- 0 – Unmask the interrupt (Enable interrupt)

11.7.1.9 INT_STATUS0 – MASKED INTERRUPT STATUS REGISTER0

- CPU Address:h30A
- Access by CPU, serial interface (RO)
- Indicate the source of the masked interrupt.

Bit [0]: • CPU frame interrupt.

Bit [1] • Control Command Frame 1 interrupt.

Bit [2] • Control Command Frame 2 interrupt.

Bit [3] • From any of the Gigabit port interrupt.

Bit [7:4] • Reserved.

11.7.1.10INT_STATUS1 – MASKED INTERRUPT STATUS REGISTER1

- (CPU Address:h30B)
- Access by CPU, serial interface (RO)
- Indicate the source of the masked interrupt.

Bit [0]: • From Gigabit port 0 interrupt

Bit [1]: • From Gigabit port 1 interrupt

Bit [2]: • From Gigabit port 2 interrupt

Bit [3]: • From Gigabit port 3 interrupt

Bit [4]: Nu

Bit [5]: Nu

Bit [6]: Nu

Bit [7]: Nu

11.7.1.11INTP_MASK0 – INTERRUPT MASK FOR MAC PORT 0,1

- CPU Address:h30C
- Accessed by CPU, serial interface (R/W)

The CPU can dynamically mask the interruption when it is busy and doesn't want to be interrupted

7	6	5	4	3	2	1	0
		P1				P0	

1 – Mask the Interrupt

0 – Unmask the Interrupt (Enable interrupt)

Bit[0]: Port 0 statistic counter Wrap around interrupt mask. An interrupt is generated when a statistic counter gets to its maximum value and wraps around. Refer to hardware statistic counter for interrupt sources. **(Default 1'b1)**

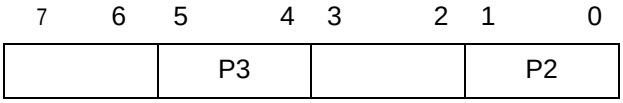
Bit [1]: Port 0 Link change mask. **(Default 1'b1)**

Bit [4]: Port 1 statistic counter Wrap around interrupt mask. **(Default 1'b1)**

Bit [5]: Port 1 Link change mask. **(Default 1'b1)**

11.7.1.12INTP_MASK1 – INTERRUPT MASK FOR MAC PORT 2,3

- CPU Address:h30D
- Accessed by CPU, serial interface (R/W)



Bit [0]: • Port 2 WAS mask (Default 1'b1)

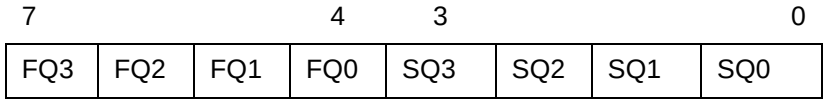
Bit [1]: • Port 2 link change mask (Default 1'b1)

Bit [4]: • Port 3 WAS mask (Default 1'b1)

Bit [5]: • Port 3 link change mask (Default 1'b1)

11.7.2 RQS – Receive Queue Select

- CPU Address:h310
- Accessed by CPU, serial interface (RW)
- This register selects which receive queue is enable to send data to the CPU.



Bit[0]: Select Queue 0. If set to one, this queue may be scheduled to CPU port. If set to zero, this queue will be blocked. If multiple queues are selected, a strict priority will be applied. Q3> Q2> Q1> Q0. Same applies to bits [3:1]. See QoS application note for more information.

Bit[1]: Select Queue 1

Bit[2]: Select Queue 2

Bit[3]: Select Queue 3

Note: Strip priority applies between different selected queues (Q3>Q2>Q1>Q0)

Bit[4]: Enable flush Queue 0

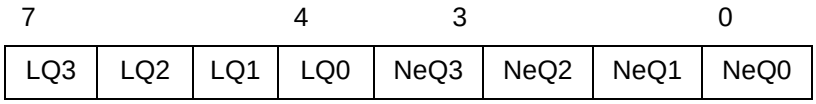
Bit[5]: Enable flush Queue 1

Bit[6]: Enable flush Queue 2
Bit[7]: Enable flush Queue 3

When flush (drop frames) is enable, it starts when queue is too long or entry is too old. A queue is too long when it reaches WRED thresholds. Queue 0 is not subject to early drop. Packets in queue 0 are dropped only when the queue is too old. An entry is too old when it is older than the time programmed in the register TX_AGE [5:0]. CPU can dynamically program this register reading register RQSS [7:4].

11.7.3 RQSS – Receive Queue Status

- CPU Address:h311
- Accessed by CPU, serial interface (RO)



CPU queue status:

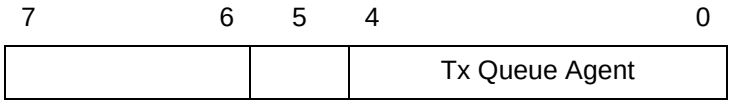
Bit[3:0]: Queue 3 to 0 not empty

Bit[4]: Head of line entry for Queue 3 to 0 is valid for too long. CPU queue 0 has no WRED threshold

Bit[7:5]: Head of line entry for Queue 3 to 0 is valid for too long or Queue length is longer than WRED threshold

11.7.4 TX_AGE – Tx Queue Aging timer

- I^2C Address: h03B;CPU Address:h312
- Accessed by CPU, serial interface (RO)



Bit[4:0]: Unit of 100ms (Default 8)Disable transmission queue aging if value is zero.

Bit[5]Must be set to ‘0’

Bit[7:6]: Reserved

11.8 Group 4 Address

11.8.1 Search Engine Group

11.8.1.1 AGETIME_LOW – MAC ADDRESS AGING TIME LOW

- I^2C Address h03C; CPU Address:h400
- Accessed by CPU, serial interface and I^2C (R/W)
- Bit [7:0] Low byte of the MAC address aging timer. (Default 2c)
- The 2800 removes the MAC address from the data base and sends a Delete MAC Address Control Command to the CPU. Mac address aging is enable/disable by boot strap T_d[9].

11.8.1.2 AGETIME_HIGH –MAC address aging time High

- I^2C Address h03D; CPU Address h401
- Accessed by CPU, serial interface and I^2C (R/W)
- Bit [7:0]: High byte of the MAC address aging timer. (Default 00)

Aging time is based on the following equation:

{AGETIME_HIGH, AGETIME_LOW} X (# of MAC entries X100µsec)

Note: the number of entries= 66K when T_d[5] is pull down (SRAM memory size = 512K) and 34K when T_d[5] is pull up (SRAM memory size = 256K).

11.8.1.3 V_AGETIME – VLAN TO PORT AGING TIME

- CPU Address h402
- Accessed by CPU (R/W)
- Bit [7:0] - 2msec/unit. (Default FF)

11.8.1.4 SE_OPMODE – SEARCH ENGINE OPERATION MODE

- CPU Address:h403
- Accessed by CPU (R/W)

7	6	5	4	3	2	1	0
SL	DMS	ARP	DRA	DA	DRD	DRN	FL

Bit [0]:
- 1 – Enable fast learning mode. In this mode, the hardware learns all the new MAC addresses at highest rate, and reports to the CPU while the hardware scans the MAC database. When the CPU report queue is full, the MAC address is learned and marked as “Not reported”. When the hardware scans the database and finds a MAC address marked as “Not Reported” it tries to report it to the CPU. The scan rate must be set. SCAN Control register sets the scan rate.**(Default 0)**
- 0 – Search Engine learns a new MAC address and sends a message to the CPU report queue. If queue is full, the learning is temporarily halted.

Bit [1]:
- 1 – Disable report new VLAN port association**(Default 0)**
- 0 – Report new VLAN port association

Bit [2]:
- Report control
  - 1 – Disable report MAC address deletion **(Default 0)**
  - 0 – Report MAC address deletion (MAC address is deleted from MCT after aging time)

Bit [3]:
- Delete Control
  - 1 – Disable aging logic from removing MAC during aging **(Default 0)**
  - 0 – MAC address entry is removed when it is old enough to be aged.
  - However, a report is still sent to the CPU in both cases, when bit[2] = 0

Bit [4]:
- 1 – Disable report aging VLAN port association **(Default 0)**
- 0 – Enable Report aging VLAN. VLAN is not removed by hardware. The CPU needs to remove the VLAN –port association.

Bit [5]:
- 1 - Report ARP packet to CPU **(Default 0)**

Bit [6]:
- Disable MCT speedup aging **(Default 0)**
  - 1 – Disable speedup aging when MCT resource is low.
  - 0 – Enable speedup aging when MCT resource is low.

Bit [7]:
- Slow Learning **(Default 0)**
  - 1– Enable slow learning. Learning is temporary disabled when search demand is high
  - 0 – Learning is performed independent of search demand

11.8.1.5 SCAN – SCAN CONTROL REGISTER

- CPU Address h404
- Accessed by CPU (R/W)

7	6 ... 0
R	Ratio

SCAN is used when fast learning is enabled (SE_OP MODE bit 0). It is used for setting up the report rate for newly learned MAC addresses to the CPU.

Bit [6:0]:
- Ratio between database scanning and aging round (Default 00)

Bit [7]:
- Reverse the ratio between scanning round and aging round **(Default 0)**

Examples:

R= 0, Ratio = 0: All aging rounds are used for aging

R= 0, Ratio = 1: Aging and scanning in every other aging round

R= 1, Ratio = 7: In eight rounds, one is used for scanning and seven is used for aging

R= 0, Ratio = 7: In eight rounds, one is used for aging and seven is used for scanning

11.9 Group 5 Address

11.9.1 Buffer Control/QOS Group

11.9.1.1 FCBAT – FCB AGING TIMER

- I^2C Address h03E; CPU Address:h500

7 ... 0
FCBAT

Bit [7:0]:
- FCB Aging time. Unit of 1ms. (**Default FF**)
- FCBAT define the aging time out interval of FCB handle

11.9.1.2 QOSC – QOS CONTROL

- I^2C Address h03F; CPU Address:h501
- Accessed by CPU, serial interface and I^2C (R/W)

7	6	5	4	3 1	0
Tos-d	Tos-p	CPUQ	VF1c		fb

Bit [0]:
- QoS frame lost is OK. Priority will be available for flow control enabled source only when this bit is set **(Default 0)**

Bit [4]:
- Per VLAN Multicast Flow Control **(Default 0)**
  - 0 – Disable
  - 1 - Enable

Bit [5]:
- CPU multicast queues size
  - 0 = 16 entries
  - 1 = 160 entries

Bit [6]:
- Select TOS bits for Priority **(Default 0)**
  - 0 – Use TOS [4:2] bits to map the transmit priority
  - 1 – Use TOS [5:3] bits to map the transmit priority

Bit [7]:
- Select TOS bits for Drop **(Default 0)**
  - 0 – Use TOS [4:2] bits to map the drop priority
  - 1 – Use TOS [5:3] bits to map the drop priority

11.9.1.3 FCR – FLOODING CONTROL REGISTER

- I^2C Address h040; CPU Address:h502
- Accessed by CPU, serial interface and I^2C (R/W)

7	6 4	3 0
Tos	TimeBase	U2MR

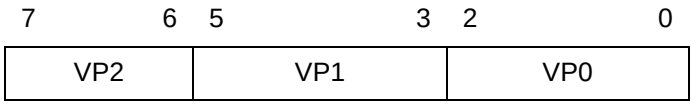
Bit [3:0]:
- U2MR: Unicast to Multicast Rate. Units in terms of time base defined in bits [6:4]. This is used to limit the amount of flooding traffic. The value in U2MR specifies how many packets are allowed to flood within the time specified by bit [6:4]. To disable this function, program U2MR to 0. (**Default = 4'h8**)

Bit [6:4]:
- TimeBase: (Default = 000)
  - 000 = 10us
  - 001 = 20us
  - 010 = 40us
  - 011 = 80us
  - 100 = 160us
  - 101 = 320us
  - 110 = 640us
  - **111 = 10us**, same as 000.

Bit [7]:
- Select VLAN tag or TOS field (IP packets) to be preferentially picked to map transmit priority and drop priority (**Default = 0**).
  - 0 – Select VLAN tag priority field over TOS field
  - 1 – Select TOS field over VLAN tag priority field

11.9.1.4 AVPML – VLAN Priority Map

- I^2C Address h041; CPU Address:h503
- Accessed by CPU, serial interface and I^2C (R/W)



Registers AVPML, AVPMM, and AVPMH allow the eight VLAN priorities to map into eight internal level transmit priorities. Under the internal transmit priority, “seven” is the highest priority where as “zero” is the lowest. This feature allows the user the flexibility of redefining the VLAN priority field. For example, programming a value of 7 into bit 2:0 of the AVPML register would map packet VLAN priority) into internal transmit priority 7. The new priority is used only inside the 2802. When the packet goes out it carries the original priority.

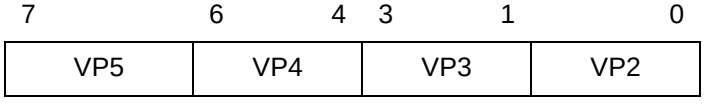
Bit [2:0]: Mapped priority of 0 **(Default 000)**

Bit [5:3]: Mapped priority of 1 **(Default 001)**

Bit [7:6]: Mapped priority of 2 **(Default 10)**

11.9.1.5 AVPMM – VLAN Priority Map

- I^2C Address h042, CPU Address:h504
- Accessed by CPU, serial interface and I^2C (R/W)



Map VLAN priority into eight level transmit priorities:

Bit [0]: Mapped priority of 2 **(Default 0)**

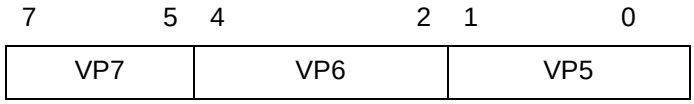
Bit [3:1]: Mapped priority of 3 **(Default 011)**

Bit [6:4]: Mapped priority of 4 **(Default 100)**

Bit [7]: Mapped priority of 5 **(Default 1)**

11.9.1.6 AVPMH – VLAN Priority Map

- I^2C Address h043, CPU Address:h505
- Accessed by CPU, serial interface and I^2C (R/W)



Map VLAN priority into eight level transmit priorities:

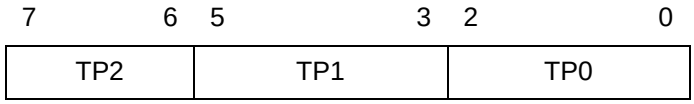
Bit [1:0]: Mapped priority of 5 **(Default 10)**

Bit [4:2]: Mapped priority of 6 **(Default 110)**

Bit [7:5]: Mapped priority of 7 **(Default 111)**

11.9.1.7 TOSPML – TOS PRIORITY MAP

- I^2C Address h044, CPU Address:h506
- Accessed by CPU, serial interface and I^2C (R/W)



Map TOS field in IP packet into four level transmit priorities

Bit [2:0]: Mapped priority when TOS is 0 **(Default 000)**

Bit [5:3]: Mapped priority when TOS is 1 **(Default 001)**

Bit [7:6]: Mapped priority when TOS is 2 **(Default 10)**

11.9.1.8 TOSPMM – TOS PRIORITY MAP

- I^2C Address h045, CPU Address:h507
- Accessed by CPU, serial interface and I^2C (R/W)

7	6 4	3 1	0
TP5	TP4	TP3	TP2

Map TOS field in IP packet into four level transmit priorities

Bit [0]: Mapped priority when TOS is 2 **(Default 0)**

Bit [3:1]: Mapped priority when TOS is 3 **(Default 011)**

Bit [6:4]: Mapped priority when TOS is 4 **(Default 100)**

Bit [7]: Mapped priority when TOS is 5 **(Default 1)**

11.9.1.9 TOSPMH – TOS PRIORITY MAP

- I^2C Address h046, CPU Address:h508
- Accessed by CPU, serial interface and I^2C (R/W)

7 5	4 2	1 0
TP7	TP6	TP5

Map TOS field in IP packet into four level transmit priorities:

- Bit [1:0]: • Mapped priority when TOS is 5 **(Default 01)**
- Bit [4:2]: • Mapped priority when TOS is 6 **(Default 110)**
- Bit [7:5]: • Mapped priority when TOS is 7 **(Default 111)**

11.9.1.10 AVDM – VLAN DISCARD MAP

- I^2C Address h047, CPU Address:h509
- Accessed by CPU, serial interface and I^2C (R/W)

7	6	5	4	3	2	1	0
FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0

Map VLAN priority into frame discard when low priority buffer usage is above threshold. Frames with high discard (drop) priority will be discarded (dropped) before frames with low drop priority.

- 0 – Low discard priority
- 1 – High discard priority

Bit [0]: Frame discard priority for frames with VLAN transmit priority 0 **(Default 0)**

Bit [1]: Frame discard priority for frames with VLAN transmit priority 1 **(Default 0)**

Bit [2]: Frame discard priority for frames with VLAN transmit priority 2 **(Default 0)**

Bit [3]: Frame discard priority for frames with VLAN transmit priority 3 **(Default 0)**

Bit [4]: Frame discard priority for frames with VLAN transmit priority 4 **(Default 0)**

Bit [5]: Frame discard priority for frames with VLAN transmit priority 5 **(Default 0)**

Bit [6]: Frame discard priority for frames with VLAN transmit priority 6 **(Default 0)**

Bit [7]: Frame discard priority for frames with VLAN transmit priority 7 **(Default 0)**

11.9.1.11TOSDML – TOS DISCARD MAP

- I^2C Address h048, CPU Address:h50A
- Accessed by CPU, serial interface and I^2C (R/W)

7	6	5	4	3	2	1	0
FDT7	FDT6	FDT5	FDT4	FDT3	FDT2	FDT1	FDT0

Map TOS into frame discard when low priority buffer usage is above threshold

Bit [0]: Frame discard priority for frames with TOS transmit priority 0 **(Default 0)**

Bit [1]: Frame discard priority for frames with TOS transmit priority 1 **(Default 0)**

Bit [2]: Frame discard priority for frames with TOS transmit priority 2 **(Default 0)**

Bit [3]: Frame discard priority for frames with TOS transmit priority 3 **(Default 0)**

Bit [4]: Frame discard priority for frames with TOS transmit priority 4 **(Default 0)**

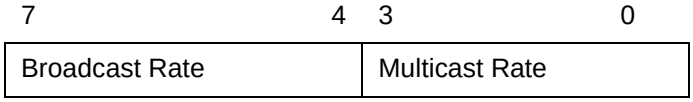
Bit [5]: Frame discard priority for frames with TOS transmit priority 5 **(Default 0)**

Bit [6]: Frame discard priority for frames with TOS transmit priority 6 **(Default 0)**

Bit [7]: Frame discard priority for frames with TOS transmit priority 7 **(Default 0)**

11.9.2 BMRC - Broadcast/Multicast Rate Control

- I^2C Address h049, CPU Address:h50B
- Accessed by CPU, serial interface and I^2C (R/W)



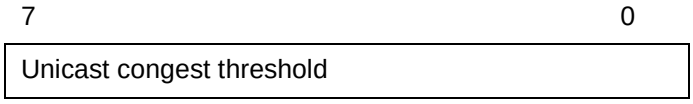
This broadcast and multicast rate defines for each port the number of incoming packet allowed to be forwarded within a specified time. Once the packet rate is reached, packets will be dropped. To turn off the rate limit, program the field to 0.

Bit [3:0]: Multicast Rate Control Number of multicast packets allowed within the time defined in bits 6 to 4 of the Flooding Control Register (FCR). **(Default 0)**.

Bit [7:4]: Broadcast Rate Control Number of broadcast packets allowed within the time defined in bits 6 to 4 of the Flooding Control Register (FCR). **(Default 0)**

11.9.3 UCC – Unicast Congestion Control

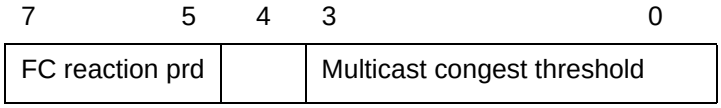
- I^2C Address h04A, CPU Address: h50C
- Accessed by CPU, serial interface and I^2C (R/W)



Bit [7:0]: Number of frame count. Used for best effort dropping at B% when destination port's best effort queue reaches UCC threshold and shared pool is all in use. Granularity 16 frame. **(Default: h07)**

11.9.4 MCC – Multicast Congestion Control

- I^2C Address h0B7, CPU Address: h50D
- Accessed by CPU, serial interface and I^2C (R/W)



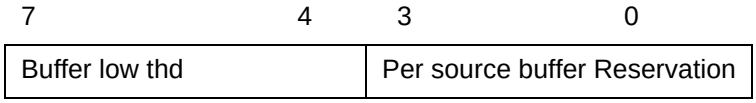
Bit [3:0]: In multiples of two. Used for triggering MC flow control when destination port's multicast best effort queue reaches MCC threshold. (Default 5'h08)

Bit [4]: Must be 0

Bit [7:5]: Flow control reaction period. ([7:5] *4 uSec)+3 uSec (Default 3'h2).

11.9.5 PRG – Port Reservation for Giga ports

- I^2C Address h0B9, CPU Address h50F
- Accessed by CPU, serial interface and I^2C (R/W)



Bit [3:0]: Per source buffer reservation. Define the space in the FDB reserved for each port. Expressed in multiples of 16 packets. For each packet 1536 bytes are reserved in the memory.
Default: 4'hA for 4MB memory
4'h6 for 2MB memory
4'h3 for 1MB memory

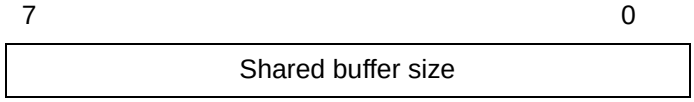
Bits [7:4]: Expressed in multiples of 16 packets. Threshold for dropping all best effort frames when destination port best effort queues reach UCC threshold and shared pool is all used and source port reservation is at or below the PRG[7:4] level. Also the threshold for initiating UC flow control.

Default: 4'h6 for 4MB memory
4'h2 for 2MB memory
4'h1 for 1MB memory

11.9.6 FCB Reservation

11.9.6.1 SFCB – SHARE FCB SIZE

- I^2C Address h04E), CPU Address h510
- Accessed by CPU, serial interface and I^2C (R/W)



Bits [7:0]: • Expressed in multiples of 8. Buffer reservation for shared pool.
(Default 4G & 4M = 8’d62)
(Default 4G & 2M = 8’d20)
(Default 4G & 1M = 8'd08
(Default 8G & 4M = 8’d150)
(Default 8G & 2M = 8’d55)
(Default 8G & 1M = 8'd25

11.9.6.2 C2RS – CLASS 2 RESERVED SIZE

- I^2C Address h04F, CPU Address h511
- Accessed by CPU, serial interface and I^2C (R/W)

Bits [7:0]: • Buffer reservation for class 2 (third lowest priority). Granularity 2. **(Default 8’h00)**

11.9.6.3 C3RS – CLASS 3 RESERVED SIZE

- I^2C Address h050, CPU Address h512
- Accessed by CPU, serial interface and I^2C (R/W)

Bits [7:0]: • Buffer reservation for class 3. Granularity 2. **(Default 8’h00)**

11.9.6.4 C4RS – CLASS 4 RESERVED SIZE

- I^2C Address h051, CPU Address h513
- Accessed by CPU, serial interface and I^2C (R/W)

Bits [7:0]: • Buffer reservation for class 4. Granularity 2. **(Default 8’h00)**

11.9.6.5 C5RS – Class 5 Reserved Size

- I^2C Address h052; CPU Address h514
- Accessed by CPU, serial interface and I^2C (R/W)

Bits [7:0]: • Buffer reservation for class 5. Granularity 2. **(Default 8'h00)**

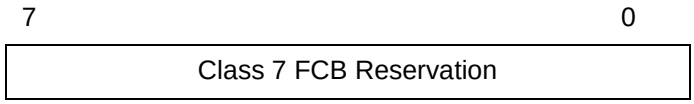
11.9.6.6 C6RS – CLASS 6 RESERVED SIZE

- I^2C Address h053; CPU Address h515
- Accessed by CPU, serial interface and I^2C (R/W)

Bits [7:0]: • Buffer reservation for class 6 (second highest priority). Granularity 2. **(Default 8'h00)**

11.9.6.7 C7RS – CLASS 7 RESERVED SIZE

- I^2C Address h054; CPU Address h516
- Accessed by CPU, serial interface and I^2C (R/W)



Bits [7:0]: • Buffer reservation for class 7 (highest priority). Granularity 2. **(Default 8'h00)**

11.9.7 Classes Byte Gigabit Port 0

- Accessed by CPU; serial interface and I^2C (R/W):

11.9.7.1 QOSC00 – BYTE_C2_G0

- I^2C Address h055, CPU Address h517

Bits [7:0]:
- Byte count threshold for C2 queue WRED **(Default 8'h28)**
- (1024byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.7.2 QOSC01 – BYTE_C3_G0

- I^2C Address h056, CPU Address h518

Bits [7:0]:
- Byte count threshold for C3 queue WRED **(Default 8'h28)**
- (512byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.7.3 QOSC02 – BYTE_C4_G0

- I^2C Address h057, CPU Address h519

Bits [7:0]:
- Byte count threshold for C4 queue WRED **(Default 8'h28)**
- (256byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.7.4 QOSC03 – BYTE_C5_G0

- I^2C Address h058, CPU Address h51A

Bits [7:0]:
- Byte count threshold for C5 queue WRED **(Default 8'h28)**
- (128byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.7.5 QOSC04 – BYTE_C6_G0

- I^2C Address h059, CPU Address h51B

Bits [7:0]:
- Byte count threshold for C6 queue WRED **(Default 8'h50)**
- (64byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.7.6 QOSC05 – BYTE_C7_G0

- I^2C Address h05A, CPU Address h51C

Bits [7:0]:
- Byte count threshold for C6 queue WRED **(Default 8'h50)**
- (64byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

QOSC00 through QOSC05 represent the values F-A in Table 3 for Gigabit port 0. They are per-queue byte thresholds for weighted random early drop (WRED). QOSC05 represents A, and QOSC00 represents F. See QoS application note for more information.

11.9.8 Classes Byte Gigabit Port 1

- Accessed by CPU; serial interface and I^2C (R/W):

11.9.8.1 QOSC06 – BYTE_C2_G1

- I^2C Address h05B, CPU Address 5h1D

Bits [7:0]:
- Byte count threshold for C2 queue WRED **(Default 8'h28)**
- (1024byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.8.2 QOSC07 – BYTE_C3_G1

- I^2C Address h05C, CPU Address h51E

Bits [7:0]
- Byte count threshold for C3 queue WRED **(Default 8'h28)**
- (512 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.8.3 QOSC08 – BYTE_C4_G1

- I^2C Address h05D, CPU Address h51F

Bits [7:0]:
- Byte count threshold for C4 queue WRED **(Default 8'h28)**
- (256 byte/unit when Delay Bound is used)
- (1024byte/unit when WFQ is used)

11.9.8.4 QOSC09 – BYTE_C5_G1

- I^2C Address h05E, CPU Address h520

Bits [7:0]:
- Byte count threshold for C5 queue WRED **(Default 8'h28)**
- (128 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.8.5 QOSC0A – BYTE_C6_G1

- I^2C Address h05F, CPU Address h521

Bits [7:0]:
- Byte count threshold for C6 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.8.6 QOSC0B – BYTE_C7_G1

- I^2C Address h060, CPU Address h522

Bits [7:0]:
- Byte count threshold for C7 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

QOSC06 through QOSC0B represent the values F-A in Table 3. They are per-queue byte thresholds for random early drop. QOSC0B represents A, and QOSC06 represents F. See QoS application note for more information

11.9.9 Classes Byte Gigabit Port 2

- Accessed by CPU; serial interface and I^2C (R/W):

11.9.9.1 QOSC0C – BYTE_C2_G2

- I^2C Address h061, CPU Address h523

Bits [7:0]:
- Byte count threshold for C2 queue WRED **(Default 8'h28)**
- (1024 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.9.2 QOSC0D – BYTE_C3_G2

- I^2C Address h062, CPU Address h524

Bits [7:0]:
- Byte count threshold for C3 queue WRED **(Default 8'h28)**
- (512 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.9.3 QOSC0E – BYTE_C4_G2

- I^2C Address h063, CPU Address h525

Bits [7:0]:
- Byte count threshold for C4 queue WRED **(Default 8'h28)**
- (256 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.9.4 QOSC0F – BYTE_C5_G2

- I^2C Address h064, CPU Address h526

Bits [7:0]:
- Byte count threshold for C5 queue WRED **(Default 8'h28)**
- (128 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.9.5 QOSC10 – BYTE_C6_G2

- I^2C Address h065, CPU Address h527

Bits [7:0]:
- Byte count threshold for C6 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.9.6 QOSC11 – BYTE_C7_G2

- I^2C Address h066, CPU Address h528

Bits [7:0]:
- Byte count threshold for C7 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

QOSC0C through QOSC11 represent the values F-A in Table 3. They are per-queue byte thresholds for random early drop. QOSC11 represents A, and QOSC0C represents F. See QoS application note for more information

11.9.10 Classes Byte Gigabit Port 3

- Accessed by CPU; serial interface and I^2C (R/W):

11.9.10.1QOSC12 – BYTE_C2_G3

- I^2C Address h067, CPU Address h529

Bits [7:0]:
- Byte count threshold for C2 queue WRED **(Default 8'h28)**
- (1024 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.10.2QOSC13 – BYTE_C3_G3

- I^2C Address h068, CPU Address h52A

Bits [7:0]:
- Byte count threshold for C3 queue WRED **(Default 8'h28)**
- (512 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.10.3QOSC14 – BYTE_C4_G3

- I^2C Address h069, CPU Address h52B

Bits [7:0]:
- Byte count threshold for C4 queue WRED **(Default 8'h28)**
- (256 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.10.4QOSC15 – BYTE_C5_G3

- I^2C Address h06A, CPU Address h52C

Bits [7:0]:
- Byte count threshold for C5 queue WRED **(Default 8'h28)**
- (128 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.10.5QOSC16 – BYTE_C6_G3

- I^2C Address h06B, CPU Address h52D

Bits [7:0]:
- Byte count threshold for C6 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

11.9.10.6QOSC17 – BYTE_C7_G3

- I^2C Address h06C, CPU Address h52E

Bits [7:0]:
- Byte count threshold for C7 queue WRED **(Default 8'h50)**
- (64 byte/unit when Delay Bound is used)
- (1024 byte/unit when WFQ is used)

QOSC12 through QOSC17 represent the values F-A in Table 3. They are per-queue byte thresholds for random early drop. QOSC17 represents A, and QOSC12 represents F. See QoS application note for more information

11.9.11 Classes Byte Limit CPU

- Accessed by CPU; serial interface and I^2C (R/W):

11.9.11.1QOSC30 – BYTE_C01

- CPU Address h547

Bits [7:0]:
- Byte count threshold for C1 queue (256byte/unit)

11.9.11.2QOSC31 – BYTE_C02

- CPU Address h548

Bits [7:0]:
- Byte count threshold for C2 queue (256byte/unit)

11.9.11.3QOSC32 – BYTE_C03

- CPU Address h549

Bits [7:0]: • Byte count threshold for C3 queue (256byte/unit)

QOSC30 through QOSC32 represent the values C-A for CPU port. The values A-C are per-queue byte thresholds for random early drop. QOSC32 represents A, and QOSC30 represents C. Queue 0 does not have weighted random drop. See QoS application note for more information.

11.9.12 Classes WFQ Credit - Port G0

- Accessed by CPU only

11.9.12.1QOSC33 – CREDIT_C0_G0

- CPU Address h54A

Bits [5:0]: • W0 - Credit register for WFQ. (Default 6'h04)

Bits [7:6]: • Priority type. Define one of the four QoS mode of operation for port 0 (Default 2'00)

See table below:

Queue	P7	P6	P5	P4	P3	P2	P1	P0
Option 1 Bit [7:6] = 2'B00	DELAY BOUND						BE	
Option 2 Bit [7:6] = 2'B01	SP		DELAY BOUND				BE	
Option 3 Bit [7:6] = 2'B10	SP		WFQ					
Option 4 Bit [7:6] = 2'B11	WFQ							
Credit for WFQ – Bit [5:0]	W7	W6	W5	W4	W3	W2	W1	W0

11.9.12.2QOSC34 – CREDIT_C1_G0

- CPU Address h54B

Bits [7]: • Flow control allow during WFQ scheme. **(Default 1'b1)**
  - 0 = Not support QoS when the Source port Flow control status is on.
  - 1= Always support QoS)

Bits [6]: • Flow control BE Queue only. **(Default 1'b1)**
  - 0= DO NOT send any frames if the XOFF is on.
  - 1= the P7-P2 frames can be sent even the XOFF is ON

Bits [5:0] • W1 - Credit register. **(Default 4'h04)**

Fc_allow	Fc_be_only	Lost_ok	
Egress- for dest fc_status		Ingress- for src fc status	
0	0	0	Go to BE Queue if (Src FC or Des FC on) otherwise Normal
0	0	1	Go to BE Queue if (Dest FC on) otherwise Normal
1	0	0	(WFQ only) Go to BE Queue if (Src FC on) otherwise BAD
1	0	1	(WFQ only) Always Normal
X	1	0	Go to BE Queue if (Src FC on)
X	1	1	Always Normal

11.9.12.3QOSC35 – CREDIT_C2_G0

- CPU Address h54C

Bits [5:0] • W2 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.12.4QOSC36 – CREDIT_C3_G0

- CPU Address h54D

Bits [5:0] • W3 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.12.5QOSC37 – CREDIT_C4_G0

- CPU Address h54E

Bits [5:0] • W4 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.12.6QOSC38 – CREDIT_C5_G0

- CPU Address h54F

Bits [5:0] • W5 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.12.7QOSC39– CREDIT_C6_G0

- CPU Address h550

Bits [5:0] • W6 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.12.8QOSC3A– CREDIT_C7_G0

- CPU Address h551

Bits [5:0] • W7 - Credit register. **(Default 5'h10)**

Bits [7:6]: • Reserved

QOSC33 through QOSC3Arepresents the set of WFQ parameters (see section 7.5) for Gigabit port 0. The granularity of the numbers is 1, and their sum must be 64. QOSC33 corresponds to W0, and QOSC3A corresponds to W7.

11.9.13 Classes WFQ Credit Port G1

- Access by CPU only

11.9.13.1QOSC3B – CREDIT_C0_G1

- CPU Address h552

Bits [5:0]: • W0 - Credit register for WFQ. (Default 6'h04)

Bits [7:6]: • Priority type. Define one of the four QoS mode of operation for port 1 (Default 2'00)

See table below:

Queue	P7	P6	P5	P4	P3	P2	P1	P0
Option 1 Bit [7:6] = 2'B00	DELAY BOUND						BE	
Option 2 Bit [7:6] = 2'B01	SP		DELAY BOUND				BE	
Option 3 Bit [7:6] = 2'B10	SP		WFQ					
Option 4 Bit [7:6] = 2'B11	WFQ							
Credit for WFQ – Bit [5:0]	W7	W6	W5	W4	W3	W2	W1	W0

11.9.13.2QOSC3C – CREDIT_C1_G1

- CPU Address h54B

Bits [7]:
- Flow control allow during WFQ scheme. **(Default 1'b1)**
  - 0 = Not support QoS when the Source port Flow control status is on.
  - 1= Always support QoS)

Bits [6]:
- Flow control BE Queue only. **(Default 1'b1)**
  - 0= DO NOT send any frames if the XOFF is on.
  - 1= the P7-P2 frames can be sent even the XOFF is ON

Bits [5:0]
- W1 - Credit register. **(Default 4'h04)**

Fc_allow	Fc_be_only	Lost_ok	
Egress- for dest fc_status		Ingress- for src fc status	
0	0	0	Go to BE Queue if (Src FC or Des FC on) otherwise Normal
0	0	1	Go to BE Queue if (Dest FC on) otherwise Normal
1	0	0	(WFQ only) Go to BE Queue if (Src FC on) otherwise BAD
1	0	1	(WFQ only) Always Normal
X	1	0	Go to BE Queue if (Src FC on)
X	1	1	Always Normal

11.9.13.3QOSC3D – CREDIT_C2_G1

- CPU Address h553

Bits [5:0]
- W2 - Credit register. **(Default 4'h04)**

Bits [7:6]:
- Reserved

11.9.13.4QOSC3E – CREDIT_C3_G1

- CPU Address h554

Bits [5:0]
- W3 - Credit register. **(Default 4'h04)**

Bits [7:6]:
- Reserved

11.9.13.5QOSC3F – CREDIT_C4_G1

- CPU Address h555

Bits [5:0] • W4 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.13.6QOSC40 – CREDIT_C5_G1

- CPU Address h556

Bits [5:0] • W5 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.13.7QOSC41– CREDIT_C6_G1

- CPU Address h557

Bits [5:0] • W6 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.13.8QOSC42– CREDIT_C7_G1

- CPU Address h558

Bits [5:0] • W7 - Credit register. **(Default 5'h10)**

Bits [7:6]: • Reserved

QOSC3B through QOSC42 represents the set of WFQ parameters (see section 7.5) for Gigabit port 1. The granularity of the numbers is 1, and their sum must be 64. QOSC3B corresponds to W0, and QOSC42 corresponds to W7

11.9.14 Classes WFQ Credit Port G2

- Access by CPU only

11.9.14.1QOSC43 – CREDIT_C0_G2

- CPU Address h55A

Bits [5:0]: • W0 - Credit register for WFQ. (Default 6'h04)

Bits [7:6]: • Priority type. Define one of the four QoS mode of operation for port 2 (Default 2'00)

See table below:

Queue	P7	P6	P5	P4	P3	P2	P1	P0
Option 1 Bit [7:6] = 2'B00	DELAY BOUND						BE	
Option 2 Bit [7:6] = 2'B01	SP		DELAY BOUND				BE	
Option 3 Bit [7:6] = 2'B10	SP		WFQ					
Option 4 Bit [7:6] = 2'B11	WFQ							
Credit for WFQ – Bit [5:0]	W7	W6	W5	W4	W3	W2	W1	W0

11.9.14.2QOSC44 – CREDIT_C1_G2

- CPU Address h55B

Bits [7]:
- Flow control allow during WFQ scheme. **(Default 1'b1)**
  - 0 = Not support QoS when the Source port Flow control status is on.
  - 1= Always support QoS)

Bits [6]:
- Flow control BE Queue only. **(Default 1'b1)**
  - 0= DO NOT send any frames if the XOFF is on.
  - 1= the P7-P2 frames can be sent even the XOFF is ON

Bits [5:0]
- W1 - Credit register. **(Default 4'h04)**

Fc_allow	Fc_be_only	Lost_ok	
Egress- for dest fc_status		Ingress- for src fc status	
0	0	0	Go to BE Queue if (Src FC or Des FC on) otherwise Normal
0	0	1	Go to BE Queue if (Dest FC on) otherwise Normal
1	0	0	(WFQ only) Go to BE Queue if (Src FC on) otherwise BAD
1	0	1	(WFQ only) Always Normal
X	1	0	Go to BE Queue if (Src FC on)
X	1	1	Always Normal

11.9.14.3QOSC45 – CREDIT_C2_G2

- CPU Address h55C

Bits [5:0] • W2 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.14.4QOSC46 – CREDIT_C3_G2

- CPU Address h55D

Bits [5:0] • W3 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.14.5QOSC47 – CREDIT_C4_G2

- CPU Address h55E

Bits [5:0] • W4 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.14.6QOSC48 – CREDIT_C5_G2

- CPU Address h55F

Bits [5:0] • W5 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.14.7QOSC49– CREDIT_C6_G2

- CPU Address h560

Bits [5:0] • W6 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.14.8QOSC4A– CREDIT_C7_G2

- CPU Address h561

Bits [5:0] • W7 - Credit register. **(Default 5'h10)**

Bits [7:6]: • Reserved

QOSC43 through QOSC4Arepresents the set of WFQ parameters (see section 7.5) for Gigabit port 2. The granularity of the numbers is 1, and their sum must be 64. QOSC43 corresponds to W0, and QOSC4A corresponds to W7.

11.9.15 Classes WFQ Credit Port G3

- Access by CPU only

11.9.15.1QOSC4B – CREDIT_C0_G3

- CPU Address h562

Bits [5:0]: • W0 - Credit register for WFQ. (Default 6'h04)

Bits [7:6]: • Priority type. Define one of the four QoS mode of operation for port 3 (Default 2'00)

See table below:

Queue	P7	P6	P5	P4	P3	P2	P1	P0
Option 1 Bit [7:6] = 2'B00	DELAY BOUND						BE	
Option 2 Bit [7:6] = 2'B01	SP		DELAY BOUND				BE	
Option 3 Bit [7:6] = 2'B10	SP		WFQ					
Option 4 Bit [7:6] = 2'B11	WFQ							
Credit for WFQ – Bit [5:0]	W7	W6	W5	W4	W3	W2	W1	W0

11.9.15.2QOSC4 – CREDIT_C1_G3

- CPU Address h563

Bits [7]: • Flow control allow during WFQ scheme. **(Default 1'b1)**
  - 0 = Not support QoS when the Source port Flow control status is on.
  - 1= Always support QoS)

Bits [6]: • Flow control BE Queue only. **(Default 1'b1)**
  - 0= DO NOT send any frames if the XOFF is on.
  - 1= the P7-P2 frames can be sent even the XOFF is ON

Bits [5:0] • W1 - Credit register. **(Default 4'h04)**

Fc_allow	Fc_be_only	Lost_ok	
Egress- for dest fc_status		Ingress- for src fc status	
0	0	0	Go to BE Queue if (Src FC or Des FC on) otherwise Normal
0	0	1	Go to BE Queue if (Dest FC on) otherwise Normal
1	0	0	(WFQ only) Go to BE Queue if (Src FC on) otherwise BAD
1	0	1	(WFQ only) Always Normal
X	1	0	Go to BE Queue if (Src FC on)
X	1	1	Always Normal

11.9.15.3QOSC4D – CREDIT_C2_G3

- CPU Address h564

Bits [5:0] • W2 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.15.4QOSC4E – CREDIT_C3_G3

- CPU Address h565

Bits [5:0] • W3 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.15.5QOSC4F – CREDIT_C4_G3

- CPU Address h566

Bits [5:0] • W4 - Credit register. **(Default 4'h04)**

Bits [7:6]: • Reserved

11.9.15.6QOSC50 – CREDIT_C5_G3

- CPU Address h567

Bits [5:0] • W5 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.15.7QOSC51– CREDIT_C6_G3

- CPU Address h568

Bits [5:0] • W6 - Credit register. **(Default 5'h8)**

Bits [7:6]: • Reserved

11.9.15.8QOSC52– CREDIT_C7_G3

- CPU Address h569

Bits [5:0] • W7 - Credit register. **(Default 5'h10)**

Bits [7:6]: • Reserved

QOSC4B through QOSC52 represents the set of WFQ parameters (see section 7.5) for Gigabit port 3. The granularity of the numbers is 1, and their sum must be 64. QOSC4B corresponds to W0, and QOSC52 corresponds to W7.

11.9.16 Class 6 Shaper Control Port G0

- Accessed by CPU only

11.9.16.1QOSC73 – TOKEN_RATE_G0

- CPU Address h58A

Bits [7:0] • Bytes allow to transmit every frame time (0.512usec) when regulated by Shaper logic. **(Default: 8'h08)**

11.9.16.2QOSC74 – TOKEN_LIMIT_G0

- CPU Address h58B

Bits [7:0] • Bytes allow to continue transmit out when regulated by Shaper logic. (16byte/unit) **(Default: 8'hC0)**

QOSC73 and QOSC74 correspond to parameters from section 7.6 on the shaper for EF traffic. QOSC73 is an integer less than 64 (average rate), with granularity 1. QOSC74 is the programmed maximum value of the counter (maximum burst size). This value is expressed in multiples of 16. QOSC73 and QOSC74 apply to Gigabit port 0. Register QOSC39-CREDIT_C6_G0 programs the peak rate. See QoS application note for more information.

11.9.17 Class 6 Shaper Control Port G1

- Accessed by CPU only

11.9.17.1QOSC75 – TOKEN_RATE_G1

- CPU Address h58C

Bits [7:0]	• Bytes allow to transmit every frame time (0.512usec) when regulated by Shaper logic. **(Default: 8'h08)**

11.9.17.2QOSC76 – TOKEN_LIMIT_G1

- CPU Address h58D

Bits [7:0]	• Bytes allow to continue transmit out when regulated by Shaper logic. (16byte/unit) **(Default: 8'hC0)**

QOSC75 and QOSC76 correspond to parameters from section 7.6 on the shaper for EF traffic. QOSC75 is an integer less than 64 (average rate), with granularity 1. QOSC76 is the programmed maximum value of the counter (maximum burst size). This value is expressed in multiples of 16. QOSC75 and QOSC76 apply to Gigabit port 0. Register QOSC41-CREDIT_C6_G1 programs the peak rate. See QoS application note for more information.

11.9.18 Class 6 Shaper Control Port G2

- Accessed by CPU only

11.9.18.11QOSC77 – TOKEN_RATE_G2

- CPU Address h58E

Bits [7:0]	• Bytes allow to transmit every frame time (0.512usec) when regulated by Shaper logic. **(Default: 8'h08)**

11.9.18.2QOSC78 – TOKEN_LIMIT_G2

- CPU Address h58F

Bits [7:0]	• Bytes allow to continue transmit out when regulated by Shaper logic. (16byte/unit) **(Default: 8'hC0)**

QOSC77 and QOSC78 correspond to parameters from section 7.6 on the shaper for EF traffic. QOSC77 is an integer less than 64 (average rate), with granularity 1. QOSC78 is the programmed maximum value of the counter (maximum burst size). This value is expressed in multiples of 16. QOSC77 and QOSC78 apply to Gigabit port 2. QOSC49-CREDIT_C6_G2 programs the peak rate. See QoS application note for more information.

11.9.19 Class 6 Shaper Control Port G3

- Accessed by CPU only

11.9.19.1QOSC79 – TOKEN_RATE_G3

- CPU Address h590

Bits [7:0] • Bytes allow to transmit every frame time (0.512usec) when regulated by Shaper logic. **(Default: 8'h08)**

11.9.19.2QOSC7A – TOKEN_LIMIT_G3

- CPU Address h591

Bits [7:0] • Bytes allow to continue transmit out when regulated by Shaper logic. (16byte/unit) **(Default: 8'hC0)**

QOSC79 and QOSC7A correspond to parameters from section 7.6 on the shaper for EF traffic. QOSC79 is an integer less than 64 (average rate), with granularity 1. QOSC7A is the programmed maximum value of the counter (maximum burst size). This value is expressed in multiples of 16. QOSC79 and QOSC7A apply to Gigabit port 3. QOSC51-CREDIT_C6_G3 programs the peak rate. See QoS application note for more information.

11.9.20 RDRC0 – WRED Rate Control 0

- I^2C Address h085, CPU Address h59A
- Accessed by CPU, Serial Interface and I^2C (R/W)

7 4	3 0
X Rate	Y Rate

Bits [7:4]: • Corresponds to the percentage X% in Chapter 7. Used for random early drop. Granularity 6.25%. **(Default: 4'h8)**

Bits[3:0]: • Corresponds to the percentage Y% in Chapter 7. Used for random early drop. Granularity 6.25%.**(Default: 4'hE)**

11.9.21 RDRC1 – WRED Rate Control 1

- I^2C Address h086, CPU Address h59B
- Accessed by CPU, Serial Interface and I^2C (R/W)

7 4	3 0
Z Rate	B Rate

Bits [7:4]: • Corresponds to the percentage Z% in Chapter 7. Used for random early drop. Granularity 6.25%.%. **(Default: 4'h6)**

Bits[3:0]: • Corresponds to the best effort frame drop percentage B%, when shared pool is all in use and destination port best effort queue reaches UCC. Used for random early drop. Granularity 6.25%.%. **(Default: 4'h8)**

11.10 Group 6 Address

11.10.1 MISC Group

11.10.1.1MII_OP0 – MII Register Option 0

- I^{2}C Address h0B1, CPU Address:h600
- Accessed by CPU, serial interface and I^{2}C (R/W)

7	6	5	4 ... 0
Hfc	1prst	NP	Vendor Spc. Reg Addr

Bit [7]:
- Half duplex flow control (Do not use half duplex mode)
  - 0 = Half duplex flow control always enable
  - 1 = Half duplex flow control by negotiation

Bit[6]:
- Link partner reset auto-negotiate disable

Bit [5]
- Next page enable
  - 1: enable
  - 0: disable

Bit[4:0]:
- Vendor specified link status register address (null value means don't use it) **(Default 00)**

11.10.1.2MII_OP1 – MII Register Option 1

- I^{2}C Address 0B2, CPU Address:h601
- Accessed by CPU, serial interface and I^{2}C (R/W)

7 ... 4	3 ... 0
Speed bit location	Duplex bit location

Bits[3:0]:
- Duplex bit location in vendor specified register

Bits [7:4]:
- Speed bit location in vendor specified register **(Default 00)**

11.10.1.3FEN – Feature Register

- I^{2}C Address h0B3, CPU Address:h602
- Accessed by CPU, serial interface and I^{2}C (R/W)

7							0
DML	MII	Rp	IP Mul	V-Sp	DS		SC

Bits [0]:
- Statistic Counter Enable **(Default 0)**
  - 0 – Disable
  - 1 – Enable
- When statistic counter is enable, an interrupt control frame is generated to the CPU, every time a counter wraps around. This feature requires an external CPU.

Bits[1]:
- Reserved

Bit [2]:
- Support DS EF Code. **(Default 0)**
  - 0 – Disable
  - 1 – Enable (all ports)
- When 101110 is detected in DS field (TOS[7:2]), the frame priority is set for 110 and drop is set for 0.

Bit [3]:
- Enable VLAN spanning tree support **(Default 0)**
  - 0 – Disable
  - 1 – Enable
- When VLAN spanning tree is enable the register ECR1Pn are not used to program the port spanning tree status. The port spanning tree status is programmed in the VLAN status field.

Bit [4]:
- Disable IP Multicast Support **(Default 1)**
  - 0 – Enable IP Multicast Support
  - 1 – Disable IP Multicast Support
- When enable, IGMP packets are identified by search engine and are passed to the CPU for processing. IP multicast packets are forwarded to the IP multicast group members according to the VLAN port mapping table.

Bit [5]:
- Enable report of new MAC and VLAN **(Default 0)**
  - 0 – Disable report to CPU
  - 1 – Enable report to CPU
- When disable: new VLAN port association report, new MAC address report and aging report are disable for all ports. When enable, register SE_OPEMODE is used to enable/disable selectively each function.

Bit [6]:
- 0: Enable MII Management State Machine **(Default 0)**
- 1: Disable MII Management State Machine

Bit [7]:
- 0: Enable using MCT Link List structure
- 1: Disable using MCT Link List structure

11.10.1.4MIIC0 – MII COMMAND REGISTER 0

- CPU Address:h603
- Accessed by CPU and serial interface only (R/W)
- Bit [7:0] MII Data [7:0]

Note: Before programming MII command: set FEN[6], check MIIC3, making sure no RDY, and no VALID; then program MII command.

11.10.1.5MIIC1 – MII COMMAND REGISTER 1

- CPU Address:h604
- Accessed by CPU and serial interface only (R/W)
- Bit [7:0] MII Data [15:8]

Note: Before programming MII command: set FEN[6], check MIIC3, making sure no RDY and no VALID; then program MII command.

11.10.1.6MIIC2 – MII COMMAND REGISTER 2

- CPU Address:h605
- Accessed by CPU and serial interface only (R/W)

7	6 5	4 0
	MII OP	Register address

Bits [4:0]: REG_AD – Register PHY Address

Bit [6:5] OP – Operation code “10” for read command and “01” for write command

Note: Before programming MII command: set FEN[6], check MIIC3, making sure no RDY and no VALID; then program MII command.

11.10.1.7 MIIC3 – MII COMMAND REGISTER 3

- CPU Address:h606
- Accessed by CPU and serial interface only (R/W)

7	6	5	4 0
Rdy	Valid		PHY address

Bits [4:0]: PHY_AD – 5 Bit PHY Address

Bit [6] VALID – Data Valid from PHY (Read Only)

Bit [7] RDY – Data is returned from PHY (Ready Only)

Note: Before programming MII command: set FEN[6], check MIIC3, making sure no RDY and no VALID; then program MII command.

11.10.1.8 MIID0 – MII DATA REGISTER 0

- CPU Address:h607
- Accessed by CPU and serial interface only (RO)
- Bit [7:0] MII Data [7:0]

11.10.1.9 MIID1 – MII DATA REGISTER 0

- CPU Address:h608
- Accessed by CPU and serial interface only (RO)
- Bit [7:0] MII Data [15:8]

11.10.1.10LED MODE – LED CONTROL

- I^2C Address:h0B4; CPU Address:h609
- Accessed by CPU, serial interface and I^2C (R/W)

7	6	5 4	3 2	1 0
lpbk		Out Pattern	Clock rate	Hold Time

Bit[1:0]
- Sample hold time **(Default 2'b00)**
  - 2'b00- 8 msec
  - 2'b01- 16 msec
  - 2'b10- 32 msec
  - 2'b11- 64 msec

Bit[3:2]

- LED clock speed (serial mode) **(Default 2'b10)**
  2'b00- sclk/128 2'b01- sclk/256
  2'b10- sclk/1024 2'b11- sclk/2048
- LED clock speed (parallel mode) **(Default 2'b10)**
  2'b00- sclk/1024 2'b01- sclk/4096
  2'b10- sclk/2048 2'b11- sclk/8192

Bit[5:4]

- LED indicator out pattern **(Default 2'b11)**
  2'b00- Normal output, LED signals go straight out, no logical combination
  2'b01- 4 bi-color LED mode
  2'b10- 3 bi-color LED mode
  2'b11- programmable mode

1. Normal mode:

LED_BYTEOUT_[7]:Collision (COL)
LED_BYTEOUT_[6]:Full duplex (FDX)
LED_BYTEOUT_[5]:Speed[1] (SP1)
LED_BYTEOUT_[4]:Speed[0] (SP0)
LED_BYTEOUT_[3]:Link (LNK)
LED_BYTEOUT_[2]:Rx (RXD)
LED_BYTEOUT_[1]:Tx (TXD)
LED_BYTEOUT_[0]:Flow Control (FC)

2. 4 bi-color LED mode

LED_BYTEOUT_[7]:COL
LED_BYTEOUT_[6]:1000FDX
LED_BYTEOUT_[5]:1000HDX
LED_BYTEOUT_[4]:100FDX
LED_BYTEOUT_[3]:100HDX
LED_BYTEOUT_[2]:10FDX
LED_BYTEOUT_[1]:10HDX
LED_BYTEOUT_[0]:ACT

Note: All output qualified by Link signal

3. 3 bi-color LED mode:

LED_BYTEOUT_[7]:COL
LED_BYTEOUT_[6]:LNK
LED_BYTEOUT_[5]:FC
LED_BYTEOUT_[4]:SPD1000
LED_BYTEOUT_[3]:SPD100
LED_BYTEOUT_[2]:FDX
LED_BYTEOUT_[1]:HDX
LED_BYTEOUT_[0]:ACT

Note: All output qualified by Link signal

4. Programmable mode:

LED_BYTEOUT_[7]:Link
LED_BYTEOUT_[6:0]:Defined by the LEDSIG6 ~ LEDSIG0 programmable registers.

Note: All output qualified by Link signal

Bit[6]: • Reserved. Must be '0'

Bit[7]: • Enable internal loop back. When this bit is set to '1' all ports work in internal loop back mode. For normal operation must be '0'.

11.10.2 CHECKSUM - EEPROM Checksum

- I^2C Address h0C5, CPU Address:h60B
- Accessed by CPU, serial interface and I^2C (R/W)

Bit [7:0]: (Default 00)

11.10.3 LED User

11.10.3.1 LEDUSER0

- I^2C Address h0BB, CPU Address:h60C
- Accessed by CPU, serial interface and I^2C (R/W)

Bit [7:0]: (Default 00)
Content will send out by LED serial logic

11.10.3.2 LEDUSER1

- I^2C Address h0BC, CPU Address:h60D
- Accessed by CPU, serial interface and I^2C (R/W)

Bit [7:0]: (Default 00)
Content will send out by LED serial logic

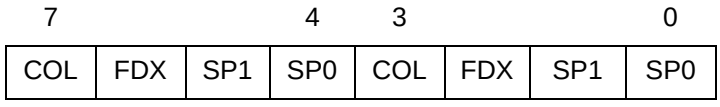
11.10.3.3 LEDUSER2/LEDSIG2

- I^2C Address h0BD, CPU Address:h60E
- Accessed by CPU, serial interface and I^2C (R/W)

In serial mode:

Bit [7:0]: (Default 00)
Content will be sent out by LED serial shift logic

In parallel mode: this register is used for programming the LED pin – led_byteout_[2]



Bit [3:0]: (Default 4'H0)
Signal polarity: 0: not invert polarity (high true)
1: invert polarity

Bit [7:4] (Default 4'H8)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[2] = AND (all selected bits)

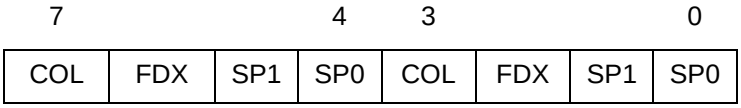
11.10.3.4 LEDUSER3/LEDSIG3

- I^2C Address:h0BE, CPU Address:h60F
- Access by CPU, serial interface (R/W)

In serial mode:

Bit [7:0]: (Default 8'H33)
Content will be sent out by LED serial shift logic.

In parallel mode: this register is used for programming the LED pin - led_byteout_[3]



Bit [3:0]: (Default 4'H3)
Signal polarity: 0: not invert polarity (high true)
1: invert polarity

Bit [7:4] (Default 4'H3)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[3] = AND (all selected bits)

11.10.3.5 LEDUSER4/LEDSIG4

- I^{2}C Address:h0BF, CPU Address:h610)
- Access by CPU, serial interface (R/W)

7							0
LED USER4							

Bit [7:0] (Default 8'H32)
Content will be sent out by LED serial shift logic.

In parallel mode: this register is used for programming the LED pin - led_byteout_[4]

7			4	3			0
COL	FDX	SP1	SP0	COL	FDX	SP1	SP0

Bit [3:0] (Default 4'H2)
Signal polarity: 0: not invert polarity (high true)
1: invert polarity

Bit [7:4] (Default 4'H3)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[4] = AND (all selected bits)

11.10.3.6 LEDUSER5/LEDSIG5

- I^{2}C Address:h0C0, CPU Address:h611
- Access by CPU, serial interface (R/W)

7							0
LED USER5							

Bit [7:0] (Default 8'H20)
Content will be sent out by LED serial shift logic.

In parallel mode: this register is used for programming the LED pin - led_byteout_[5]

7			4	3			0
COL	FDX	SP1	SP0	COL	FDX	SP1	SP0

Bit [3:0] (Default 4'H0)
Signal polarity: 0: not invert polarity (high true)
1: invert polarity

Bit [7:4] (Default 4'H2)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[5] = AND (all selected bits)

11.10.3.7 LEDUSER6/LEDSIG6

- I^2C Address:h0C1, CPU Address:h612
- Access by CPU, serial interface (R/W)

7							0
LED USER6							

Bit [7:0] (Default 8'H40)
Content will be sent out by LED serial shift logic.

In parallel mode: this register is used for programming the LED pin - led_byteout_[6]

7			4	3			0
COL	FDX	SP1	SP0	COL	FDX	SP1	SP0

Bit [3:0] (Default 4'B0000)
Signal polarity: 0: not invert polarity (high true)
1: invert polarity

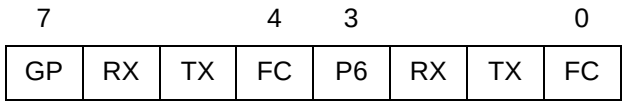
Bit [7:4] (Default 4'b0100)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[6] = AND (all selected bits), or the polarity of led_byteout_[6] is controlled by LEDSIG1_0[3]

11.10.3.8 LEDUSER7/LEDSIG1_0

- I^{2}C Address:h0C2, CPU Address:h613
- Access by CPU, serial interface (R/W)

Bit [7:0] (Default 8'H61)
Content will be sent out by LED serial shift logic.

In parallel mode: this register is used for programming the LED pin - led_byteout_[2]



Bit [7] (Default 1'B0)
- Global output polarity: this bit controls the output polarity of all led_byteout_ and led_port_sel pins.

0: no invert polarity - (led_byteout_[7:0] are high activated, led_port_sel[9:0] are low activated)
1: invert polarity - (led_byteout_[7:0] are low activated, led_port_sel[9:0] are high activated)

Bit [6:4] (Default 3'B110)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[6] = OR (all selected bits)

Bit[3] (Default 1'B0)
Polarity control of led_byteout_[6]
0: not invert
1: invert

Bit [2:0] (Default 3'b001)
Signal Select: 0: not select
1: select the corresponding bit
When bits get selected, the led_byteout_[0] = OR (all selected bits)

11.10.4 MIINP0 – MII Next Page Data Register 0

- I^2C Address:h0C3, CPU Address:h614
- Access by CPU and serial interface only (R/W)

Bit [7:0] MII next page Data [7:0]

11.10.5 MIINP1 – MII Next Page Data Register 1

- I^2C Address:h0C4, CPU Address:h615)
- Access by CPU and serial interface only (R/W)

Bit [7:0] MII next page Data [15:8]

11.11 Group F Address

11.11.1 CPU Access Group

11.11.1.1 GCR-GLOBAL CONTROL REGISTER

- CPU Address: hF00
- Accessed by CPU and serial interface. (R/W)

7	6	5	4	3	2	1	0
IP			Init	Reset	Bist	SR	SC

Bit [0]: Store configuration **(Default = 0)**
Write '1' followed by '0' to store configuration into external EEPROM

Bit[1]: Store configuration and reset **(Default = 0)**
Write '1' to store configuration into external EEPROM and reset chip

Bit[2]: Start BIST **(Default = 0)**
Write '1' followed by '0' to start the device's built-in self-test. The result is found in the DCR register.

Bit[3]: Soft Reset **(Default = 0)**
Write '1' to reset the chip

Bit[4]: Initialization Done **(Default = 0)**
This bit is meaningless when CPU is not installed. In managed mode, CPU write this bit with “1” to indicate initialization is completed and ready to forward packets.
- 1 – initialization is done
- 0 – initialization is not completed.

Bit[7] Interrupt Polarity **(Default = 0)**
- 1 - interrupt active high
- 0 - interrupt active low

11.11.1.2 DCR-Device Status and Signature Register

- CPU Address: hF01
- Accessed by CPU and serial interface. (RO)

7 6	5 4	3	2	1	0
Revision	Signature	RE	BinP	BR	BW

Bit [0]: 1 - Busy writing configuration to I^2C
0 – Not Busy writing configuration to I^2C

Bit[1]: 1 - Busy reading configuration from I^2C
0 – Not Busy reading configuration from I^2C

Bit[2]: 1 - BIST in progress
0 - BIST not running

Bit[3]: 1 - RAM Error
0 – RAM OK

Bit[5:4]: Device Signature
00 – 4 Ports Device, non-management mode
01 – 8 Ports Device, non-management mode
10 – 4 Ports Device, management mode possible (need to install CPU)
11 - 8 Ports Device, management mode possible (need to install CPU)

Bit [7:6]: Revision

11.11.1.3 DCR01-GIGA PORT STATUS

- CPU Address: hF02
- Accessed by CPU and serial interface. (RO)

7	6 4	3 2	1 0
CIC		GIGA1	GIGA0

Bit [1:0]: Giga port 0 strap option
- 00 – 100Mb MII mode
- 01 – Invalid
- 10 – GMII
- 11 – PCS

Bit[3:2] Giga port 1 strap option
- 00 – 100Mb MII mode
- 01 – Invalid
- 10 – GMII
- 11 – PCS

Bit [7] Chip initialization completed.
Note: DCR01[7], DCR23[7], DCR45[7] and DCR67[7] have the same function.

11.11.1.4 DCR23-GIGA PORT STATUS

- CPU Address: hF03
- Accessed by CPU and serial interface. (RO)

7	6 4	3 2	1 0
CIC		GIGA3	GIGA2

Bit [1:0]: Giga port 2 strap option
- 00 – 100Mb MII mode
- 01 – Invalid
- 10 – GMII
- 11 – PCS

Bit[3:2] Giga port 3 strap option
- 00 – 100Mb MII mode
- 01 – Invalid
- 10 – GMII
- 11 – PCS

Bit [7] Chip initialization completed

11.11.1.5 DPST – DEVICE PORT STATUS REGISTER

- CPU Address:hF06
- Accessed by CPU and serial interface (R/W)

Bit[2:0]: Read back index register. This is used for selecting what to read back from DTST. **(Default 00)**

- 3'B000 - Port 0 Operating mode and Negotiation status
- 3'B001 - Port 1 Operating mode and Negotiation status
- 3'B010 - Port 2 Operating mode and Negotiation status
- 3'B011 - Port 3 Operating mode and Negotiation status
- 3'B1XX - Reserved

11.11.2 DTST – Data Read Back Register 0

- CPU Address: hF07
- Accessed by CPU and serial interface (RO)

7	6	5	4	3	2	1	0
MD	InfoDet	SigDet	Giga	lnkdn	FE	Fdpx	Fc_en

This register provides various internal information as selected in DPST bit[2:0]

Bit[0]: Flow control enabled

Bit[1]: Full duplex port

Bit[2]: Fast ethernet port (if not giga)

Bit[3]: Link is down

Bit[4]: GIGA port

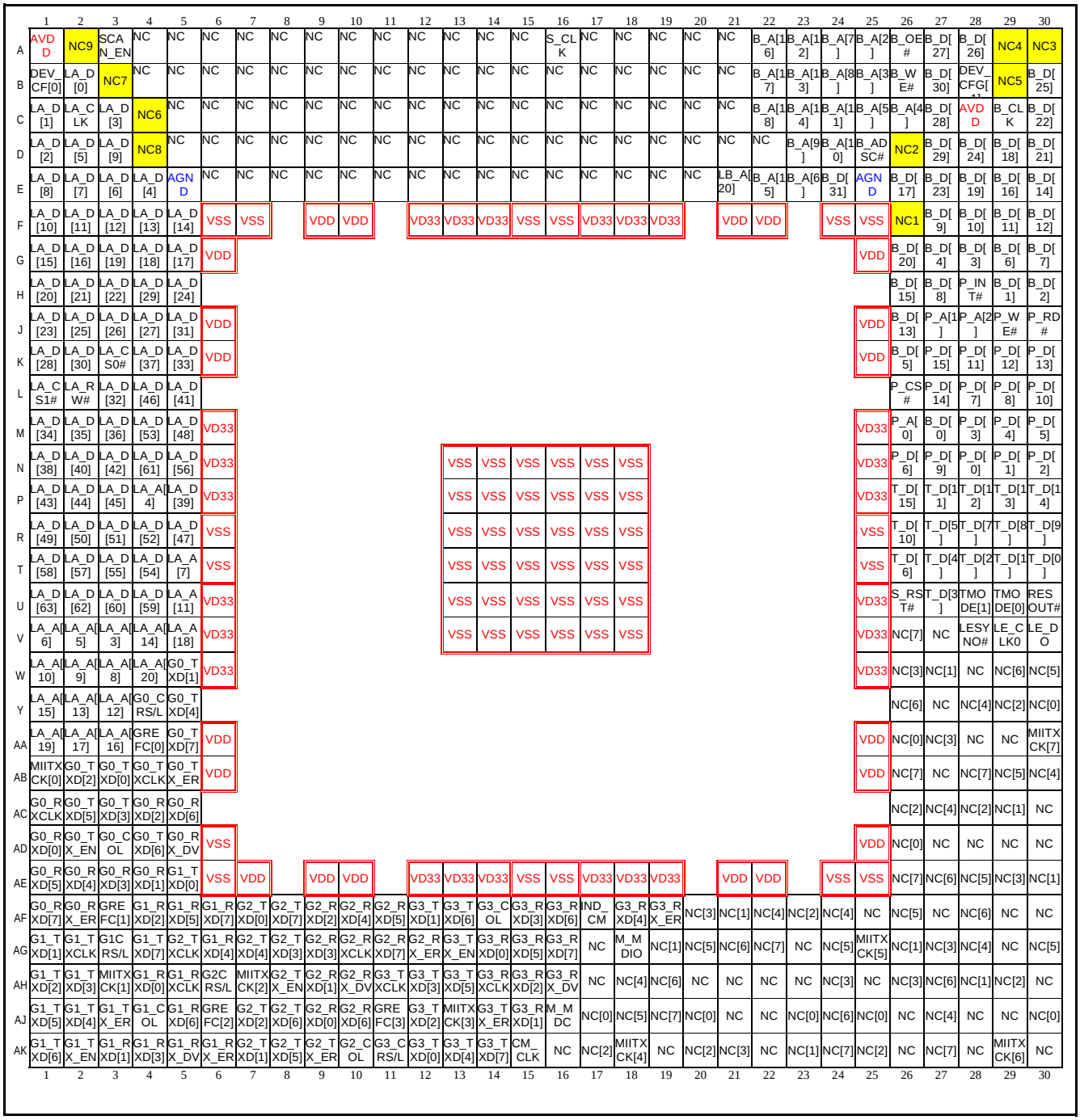
Bit[5]: Signal detect (when PCS interface mode)

Bit[6]: Pipe signal detected (pipe mode only)

Bit[7]: Module detected (for hot swap purpose)

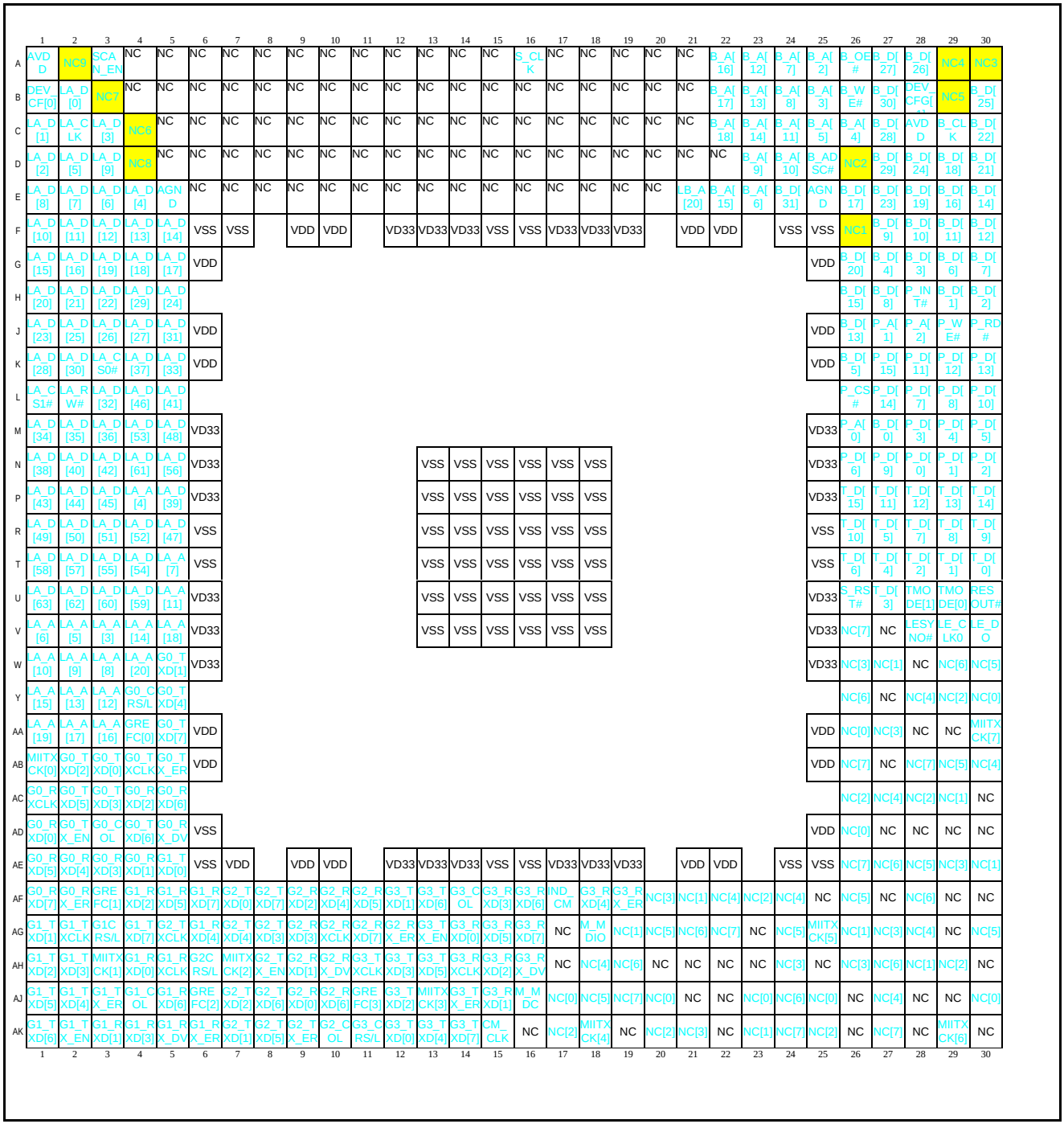
12.0 BGA and Ball Signal Description

12.1 BGA Views (Top-View)



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
A	AVDD	NC9	SCAN_EN	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	S_CLK	NC	NC	NC	NC	NC	B_A[16]	B_A[12]	B_A[7]	B_A[2]	B_OE#	B_D[27]	B_D[26]	NC4	NC3
B	DEV_CF[0]	LA_D[0]	NC7	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[17]	B_A[13]	B_A[8]	B_A[3]	B_WE#	B_D[30]	DEV_CFG[1]	NC5	B_D[25]
C	LA_D[1]	LA_CLK	LA_D[3]	NC6	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[18]	B_A[14]	B_A[1]	B_A[5]	B_A[4]	B_D[28]	AVDD	B_CLK	B_D[22]
D	LA_D[2]	LA_D[5]	LA_D[9]	NC8	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[9]	B_A[10]	B_ADSC#	NC2	B_D[29]	B_D[24]	B_D[18]	B_D[21]
E	LA_D[8]	LA_D[7]	LA_D[6]	LA_D[4]	AGND	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	LB_A[20]	B_A[15]	B_A[6]	B_D[31]	AGND	B_D[17]	B_D[23]	B_D[19]	B_D[16]	B_D[14]
F	LA_D[10]	LA_D[11]	LA_D[12]	LA_D[13]	LA_D[14]	VSS	VSS		VDD	VDD		VD33	VD33	VD33	VSS	VSS	VD33	VD33	VD33		VDD	VDD		VSS	VSS	NC1	B_D[9]	B_D[10]	B_D[11]	B_D[12]
G	LA_D[15]	LA_D[16]	LA_D[19]	LA_D[18]	LA_D[17]	VDD																			VDD	B_D[20]	B_D[4]	B_D[3]	B_D[6]	B_D[7]
H	LA_D[20]	LA_D[21]	LA_D[22]	LA_D[29]	LA_D[24]																					B_D[15]	B_D[8]	P_INT#	B_D[1]	B_D[2]
J	LA_D[23]	LA_D[25]	LA_D[26]	LA_D[27]	LA_D[31]	VDD																			VDD	B_D[13]	P_A[1]	P_A[2]	P_WE#	P_RD#
K	LA_D[28]	LA_D[30]	LA_CS0#	LA_D[37]	LA_D[33]	VDD																			VDD	B_D[5]	P_D[15]	P_D[11]	P_D[12]	P_D[13]
L	LA_CS1#	LA_RW#	LA_D[32]	LA_D[46]	LA_D[41]																					P_CS#	P_D[14]	P_D[7]	P_D[8]	P_D[10]
M	LA_D[34]	LA_D[35]	LA_D[36]	LA_D[53]	LA_D[48]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	P_A[0]	B_D[0]	P_D[3]	P_D[4]	P_D[5]
N	LA_D[38]	LA_D[40]	LA_D[42]	LA_D[61]	LA_D[56]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	P_D[6]	P_D[9]	P_D[0]	P_D[1]	P_D[2]
P	LA_D[43]	LA_D[44]	LA_D[45]	LA_A[4]	LA_D[39]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	T_D[15]	T_D[11]	T_D[12]	T_D[13]	T_D[14]
R	LA_D[49]	LA_D[50]	LA_D[51]	LA_D[52]	LA_D[47]	VSS							VSS	VSS	VSS	VSS	VSS	VSS							VSS	T_D[10]	T_D[5]	T_D[7]	T_D[8]	T_D[9]
T	LA_D[58]	LA_D[57]	LA_D[55]	LA_D[54]	LA_A[7]	VSS							VSS	VSS	VSS	VSS	VSS	VSS							VSS	T_D[6]	T_D[4]	T_D[2]	T_D[1]	T_D[0]
U	LA_D[63]	LA_D[62]	LA_D[60]	LA_D[59]	LA_A[11]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	S_RST#	T_D[3]	TMODE[1]	TMODE[0]	RESOUT#
V	LA_A[6]	LA_A[5]	LA_A[3]	LA_A[14]	LA_A[18]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	NC[7]	NC	LESYNO#	LE_CLK0	LE_DO
W	LA_A[10]	LA_A[9]	LA_A[8]	LA_A[20]	G0_TXD[1]	VD33																			VD33	NC[3]	NC[1]	NC	NC[6]	NC[5]
Y	LA_A[15]	LA_A[13]	LA_A[12]	G0_CRS/L	G0_TXD[4]																					NC[6]	NC	NC[4]	NC[2]	NC[0]
AA	LA_A[19]	LA_A[17]	LA_A[16]	GREFC[0]	G0_TXD[7]	VDD																			VDD	NC[0]	NC[3]	NC	NC	MIITXCK[7]
AB	MIITXCK[0]	G0_TXD[2]	G0_TXD[0]	G0_TXCLK	G0_TXER	VDD																			VDD	NC[7]	NC	NC[7]	NC[5]	NC[4]
AC	G0_RXCLK	G0_TXD[5]	G0_TXD[3]	G0_RXD[2]	G0_RXD[6]																					NC[2]	NC[4]	NC[2]	NC[1]	NC
AD	G0_RXD[0]	G0_TXEN	G0_COL	G0_TXD[6]	G0_RXDV	VSS																			VDD	NC[0]	NC	NC	NC	NC
AE	G0_RXD[5]	G0_RXD[4]	G0_RXD[3]	G0_RXD[1]	G1_TXD[0]	VSS	VDD		VDD	VDD		VD33	VD33	VD33	VSS	VSS	VD33	VD33	VD33		VDD	VDD		VSS	VSS	NC[7]	NC[6]	NC[5]	NC[3]	NC[1]
AF	G0_RXD[7]	G0_RXER	GREFC[1]	G1_RXD[2]	G1_RXD[5]	G1_RXD[7]	G2_TXD[0]	G2_TXD[7]	G2_RXD[2]	G2_RXD[4]	G2_RXD[5]	G3_TXD[1]	G3_TXD[6]	G3_COL	G3_RXD[3]	G3_RXD[6]	IND_CM	G3_RXD[4]	G3_RXER	NC[3]	NC[1]	NC[4]	NC[2]	NC[4]	NC	NC[5]	NC	NC[6]	NC	NC
AG	G1_TXD[1]	G1_TXCLK	G1C RS/L	G1_TXD[7]	G2_TXCLK	G1_RXD[4]	G2_TXD[4]	G2_TXD[3]	G2_RXD[3]	G2_RXCLK	G2_RXD[7]	G2_RXER	G3_TXEN	G3_RXD[0]	G3_RXD[5]	G3_RXD[7]	NC	M_MDIO	NC[1]	NC[5]	NC[6]	NC[7]	NC	NC[5]	MIITXCK[5]	NC[1]	NC[3]	NC[4]	NC	NC[5]
AH	G1_TXD[2]	G1_TXD[3]	MIITXCK[1]	G1_RXD[0]	G1_RXCLK	G2CRS/L	MIITXCK[2]	G2_TXEN	G2_RXD[1]	G2_RXDV	G3_TXCLK	G3_TXD[3]	G3_TXD[5]	G3_RXCLK	G3_RXD[2]	G3_RXDV	NC	NC[4]	NC[6]	NC	NC	NC	NC	NC[3]	NC	NC[3]	NC[6]	NC[1]	NC[2]	NC
AJ	G1_TXD[5]	G1_TXD[4]	G1_TXER	G1_COL	G1_RXD[6]	GREFC[2]	G2_TXD[2]	G2_TXD[6]	G2_RXD[0]	G2_RXD[6]	GREFC[3]	G3_TXD[2]	MIITXCK[3]	G3_TXER	G3_RXD[1]	M_MDC	NC[0]	NC[5]	NC[7]	NC[0]	NC	NC	NC[0]	NC[6]	NC[0]	NC	NC[4]	NC	NC	NC[0]
AK	G1_TXD[6]	G1_TXEN	G1_RXD[1]	G1_RXD[3]	G1_RXDV	G1_RXER	G2_TXD[1]	G2_TXD[5]	G2_TXER	G2_COL	G3_CRS/L	G3_TXD[0]	G3_TXD[4]	G3_TXD[7]	CM_CLK	NC	NC[2]	MIITXCK[4]	NC	NC[2]	NC[3]	NC	NC[1]	NC[7]	NC[2]	NC	NC[7]	NC	MIITXCK[6]	NC
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

12.2 Power and Ground Distribution



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
A	AVDD	NC9	SCAN_EN	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	S_CLK	NC	NC	NC	NC	NC	B_A[16]	B_A[12]	B_A[7]	B_A[2]	B_OE#	B_D[27]	B_D[26]	NC4	NC3
B	DEV_CF[0]	LA_D[0]	NC7	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[17]	B_A[13]	B_A[8]	B_A[3]	B_WE#	B_D[30]	DEV_CFG[	NC5	B_D[25]
C	LA_D[1]	LA_CLK	LA_D[3]	NC6	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[18]	B_A[14]	B_A[11]	B_A[5]	B_A[4]	B_D[28]	AVDD	B_CLK	B_D[22]
D	LA_D[2]	LA_D[5]	LA_D[9]	NC8	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	B_A[9]	B_A[10]	B_ADSC#	NC2	B_D[29]	B_D[24]	B_D[18]	B_D[21]
E	LA_D[8]	LA_D[7]	LA_D[6]	LA_D[4]	AGND	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	LB_A[20]	B_A[15]	B_A[6]	B_D[31]	AGND	B_D[17]	B_D[23]	B_D[19]	B_D[16]	B_D[14]
F	LA_D[10]	LA_D[11]	LA_D[12]	LA_D[13]	LA_D[14]	VSS	VSS		VDD	VDD		VD33	VD33	VD33	VSS	VSS	VD33	VD33	VD33		VDD	VDD		VSS	VSS	NC1	B_D[9]	B_D[10]	B_D[11]	B_D[12]
G	LA_D[15]	LA_D[16]	LA_D[19]	LA_D[18]	LA_D[17]	VDD																			VDD	B_D[20]	B_D[4]	B_D[3]	B_D[6]	B_D[7]
H	LA_D[20]	LA_D[21]	LA_D[22]	LA_D[29]	LA_D[24]																					B_D[15]	B_D[8]	P_INT#	B_D[1]	B_D[2]
J	LA_D[23]	LA_D[25]	LA_D[26]	LA_D[27]	LA_D[31]	VDD																			VDD	B_D[13]	P_A[1]	P_A[2]	P_WE#	P_RD#
K	LA_D[28]	LA_D[30]	LA_CS0#	LA_D[37]	LA_D[33]	VDD																			VDD	B_D[5]	P_D[15]	P_D[11]	P_D[12]	P_D[13]
L	LA_CS1#	LA_RW#	LA_D[32]	LA_D[46]	LA_D[41]																					P_CS#	P_D[14]	P_D[7]	P_D[8]	P_D[10]
M	LA_D[34]	LA_D[35]	LA_D[36]	LA_D[53]	LA_D[48]	VD33																			VD33	P_A[0]	B_D[0]	P_D[3]	P_D[4]	P_D[5]
N	LA_D[38]	LA_D[40]	LA_D[42]	LA_D[61]	LA_D[56]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	P_D[6]	P_D[9]	P_D[0]	P_D[1]	P_D[2]
P	LA_D[43]	LA_D[44]	LA_D[45]	LA_A[4]	LA_D[39]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	T_D[15]	T_D[11]	T_D[12]	T_D[13]	T_D[14]
R	LA_D[49]	LA_D[50]	LA_D[51]	LA_D[52]	LA_D[47]	VSS							VSS	VSS	VSS	VSS	VSS	VSS							VSS	T_D[10]	T_D[5]	T_D[7]	T_D[8]	T_D[9]
T	LA_D[58]	LA_D[57]	LA_D[55]	LA_D[54]	LA_A[7]	VSS							VSS	VSS	VSS	VSS	VSS	VSS							VSS	T_D[6]	T_D[4]	T_D[2]	T_D[1]	T_D[0]
U	LA_D[63]	LA_D[62]	LA_D[60]	LA_D[59]	LA_A[11]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	S_RST#	T_D[3]	TMODE[1]	TMODE[0]	RESOUT#
V	LA_A[6]	LA_A[5]	LA_A[3]	LA_A[14]	LA_A[18]	VD33							VSS	VSS	VSS	VSS	VSS	VSS							VD33	NC[7]	NC	LESYNO#	LE_CLK0	LE_DO
W	LA_A[10]	LA_A[9]	LA_A[8]	LA_A[20]	G0_TXD[1]	VD33																			VD33	NC[3]	NC[1]	NC	NC[6]	NC[5]
Y	LA_A[15]	LA_A[13]	LA_A[12]	G0_CRS/L	G0_TXD[4]																					NC[6]	NC	NC[4]	NC[2]	NC[0]
AA	LA_A[19]	LA_A[17]	LA_A[16]	GREFC[0]	G0_TXD[7]	VDD																			VDD	NC[0]	NC[3]	NC	NC	MIITXCK[7]
AB	MIITXCK[0]	G0_TXD[2]	G0_TXD[0]	G0_TXCLK	G0_TX_ER	VDD																			VDD	NC[7]	NC	NC[7]	NC[5]	NC[4]
AC	G0_RXCLK	G0_TXD[5]	G0_TXD[3]	G0_RXD[2]	G0_RXD[6]																					NC[2]	NC[4]	NC[2]	NC[1]	NC
AD	G0_RXD[0]	G0_TX_EN	G0_COL	G0_TXD[6]	G0_RX_DV	VSS																			VDD	NC[0]	NC	NC	NC	NC
AE	G0_RXD[5]	G0_RXD[4]	G0_RXD[3]	G0_RXD[1]	G1_TXD[0]	VSS	VDD		VDD	VDD		VD33	VD33	VD33	VSS	VSS	VD33	VD33	VD33		VDD	VDD		VSS	VSS	NC[7]	NC[6]	NC[5]	NC[3]	NC[1]
AF	G0_RXD[7]	G0_RX_ER	GREFC[1]	G1_RXD[2]	G1_RXD[5]	G1_RXD[7]	G2_TXD[0]	G2_TXD[7]	G2_RXD[2]	G2_RXD[4]	G2_RXD[5]	G3_TXD[1]	G3_TXD[6]	G3_COL	G3_RXD[3]	G3_RXD[6]	IND_CM	G3_RXD[4]	G3_RX_ER	NC[3]	NC[1]	NC[4]	NC[2]	NC[4]	NC	NC[5]	NC	NC[6]	NC	NC
AG	G1_TXD[1]	G1_TXCLK	G1CRS/L	G1_TXD[7]	G2_TXCLK	G1_RXD[4]	G2_TXD[4]	G2_TXD[3]	G2_RXD[3]	G2_RXCLK	G2_RXD[7]	G2_RX_ER	G3_TX_EN	G3_RXD[0]	G3_RXD[5]	G3_RXD[7]	NC	M_MDIO	NC[1]	NC[5]	NC[6]	NC[7]	NC	NC[5]	MIITXCK[5]	NC[1]	NC[3]	NC[4]	NC	NC[5]
AH	G1_TXD[2]	G1_TXD[3]	MIITXCK[1]	G1_RXD[0]	G1_RXCLK	G2CRS/L	MIITXCK[2]	G2_TX_EN	G2_RXD[1]	G2_RX_DV	G3_TXCLK	G3_TXD[3]	G3_TXD[5]	G3_RXCLK	G3_RXD[2]	G3_RX_DV	NC	NC[4]	NC[6]	NC	NC	NC	NC	NC[3]	NC	NC[3]	NC[6]	NC[1]	NC[2]	NC
AJ	G1_TXD[5]	G1_TXD[4]	G1_TX_ER	G1_COL	G1_RXD[6]	GREFC[2]	G2_TXD[2]	G2_TXD[6]	G2_RXD[0]	G2_RXD[6]	GREFC[3]	G3_TXD[2]	MIITXCK[3]	G3_TX_ER	G3_RXD[1]	M_MDC	NC[0]	NC[5]	NC[7]	NC[0]	NC	NC	NC	NC[0]	NC[6]	NC[0]	NC	NC[4]	NC	NC	NC[0]
AK	G1_TXD[6]	G1_TX_EN	G1_RXD[1]	G1_RXD[3]	G1_RX_DV	G1_RX_ER	G2_TXD[1]	G2_TXD[5]	G2_TX_ER	G2_COL	G3_CRS/L	G3_TXD[0]	G3_TXD[4]	G3_TXD[7]	CM_CLK	NC	NC[2]	MIITXCK[4]	NC	NC[2]	NC[3]	NC	NC[1]	NC[7]	NC[2]	NC	NC[7]	NC	MIITXCK[6]	NC
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

12.3 Ball- Signal Descriptions

All pins are CMOS type; all Input pins are 5 Volt tolerance, and all Output pins are 3.3 CMOS drive.

12.3.1 Ball Signal Description in Managed Mode

Ball No(s)	Symbol	I/O	Description
CPU Bus Interface			
K27, L27, K30, K29, K28, L30, N27, L29, L28, N26, M30, M29, M28, N30, N29, N28	P_DATA[15:0]	I/O-TS with pull up	Processor Bus Data Bit [15:0]
J28, J27, M26	P_A[2:0]	Input	Processor Bus Address Bit [2:0]
J29	P_WE#	Input with weak internal pull up	CPU Bus-Write Enable
J30	P_RD#	Input with weak internal pull up	CPU Bus-Read Enable
L26	P_CS#	Input with weak internal pull up	Chip Select
H28	P_INT#	Output	CPU Interrupt
Frame Buffer Interface			
U1, U2, N4, U3, U4, T1, T2, N5, T3, T4, M4, R4, R3, R2, R1, M5, R5, L4, P3, P2, P1, N3, L5, N2, P5, N1, K4, M3, M2, M1, K5, L3, J5, K2, H4, K1, J4, J3, J2, H5, J1, H3, H2, H1, G3, G4, G5, G2, G1, F5, F4, F3, F2, F1, D3, E1, E2, E3, D2., E4, C3, D1, C1, B2	LA_D[63:0]	I/O-TS with pull up	Frame Bank A– Data Bit [63:0]
AA1, V5, AA2, AA3, Y1, V4, Y2, Y3, U5, W1, W2, W3, T5, V1, V2, P4, V3	LA_A[19:3]	Output	Frame Bank A – Address Bit [19:3]
W4	LA_A[20]	Output with pull up	Frame Bank A – Address Bit [20]
C2	LA_CLK	Output	Frame Bank A Clock Input
K3	LA_CS0#	Output with pull up	Frame Bank A Low Portion Chip Selection
L1	LA_CS1#	Output with pull up	Frame Bank A High Portion Chip Selection
L2	LA_RW#	Output with pull up	Frame Bank A Read/Write

Ball No(s)	Symbol	I/O	Description
D18, B18, C18, A17, E17, B17, C17, E16, D17, B16, E15, C16, D16, D15, E14, C15, B15, E13, A15, D14, C14, D13, B14, A14, C13, E12, B13, A13, D12, C12, B12, A12, A11, E10, C10, B10, E9, A10, D11, D10, D8, D9, C9, B9, A9, C8, B8, A8, C7, E7, D7, B7, E8, A7, D6, C6, E6, B6, A6, A5, B5, C5, B4,A4	NC	I/O-TS with pullup.	No connect
D22, D20, E20, D21, A21, D19, B21, C21, A20, B20, E19, C20, A19, B19, E18, C19, A18	NC	Output	
Switch Database Interface			
E24,B27, D27, C27, A27, A28, B30, D28, E27, C30, D30, G26, E28, D29, E26, E29, H26, E30, J26, F30, F29, F28, F27, H27, G30, G29, K26, G27, G28, H30, H29, M27	B_D[31:0]	I/O-TS with pull up	Switch Database Domain - Data Bit [31:0]
C22, B22, A22, E22, C23, B23, A23, C24, D24, D23, B24, A24, E23, C25, C26, B25, A25	B_A[18:2]	Output	Switch Database Address (512K) - Address Bit [18:2]
C29	B_CLK	Output	Switch Database Clock Input
D25	B_ADSC#	Output with pull up	Switch Database Address Status Control
B26	B_WE#	Output with pull up	Switch Database Write Chip Select
A26	B_OE#	Output with pull up	Switch Database Read Chip Select
MII Management Interface			
AJ16	M_MDC	Output	MII Management Data Clock – (common for all MII Ports [3:0])
AG18	M_MDIO	I/O-TS with pull up	MII Management Data I/O – (common for all MII Ports –[3:0])) 2.5Mhz
GMII / MII Interface (193) Gigabit Ethernet Access Port			
AJ11, AJ6, AF3,AA4	GREF_CLK [3:0]	Input w/ pull up	Giga Reference Clock

Ball No(s)	Symbol	I/O	Description
AD29, AK30, AJ22, AG17,	NC		
AK15	CM_CLK	Input w/ pull up	Common Clock shared by port G[3:0]
AF17	IND/CM	Input w/ pull up	1: select GREF_CLK[3:0] as clock 0: select CM_CLK as clock for all ports
AJ13, AH7, AH3, AB1	MII TX CLK[3:0]	Input w/ pull up	
AA30, AK29, AG25, AK18,	NC		
AG16, AF16, AG15, AF18, AF15, AH15, AJ15, AG14 AG11, AJ10, AF11, AF10, AG9, AF9, AH9, AJ9 AF6, AJ5, AF5, AG6, AK4, AF4, AK3, AH4 AF1, AC5, AE1, AE2, AE3, AC4, AE4, AD1	G3_RXD[7:0] G2_RXD[7:0] G1_RXD[7:0] G0_RXD[7:0]	Input w/ pull up	G[3:0] port – Receive Data Bit [7:0]
V26, W29, W30, Y28, W26, Y29, W27, Y30 AB26, AE27, AE28, AC27, AE29, AC26, AE30, AD26 AK27, AH27, AF26, AJ27, AH26, AK25, AG26, AJ25 AG22, AG21, AG20, AF22, AK21, AK20, AF21, AJ20	NC		
AH16, AH10, AK5, AD5	G[3:0]_RX_DV	Input w/ pull down	G[3:0]port – Receive Data Valid
W28, AD30, AK28, AH22,	NC		
AF19, AG12, AK6, AF2	G[3:0]_RX_ER	Input w/ pull up	G[3:0]port – Receive Error
V27, AD27, AJ28, AH23,	NC		
AK11, AH6, AG3, Y4	G[3:0]_CRS/LINK	Input w/ pull down	G[3:0]port – Carrier Sense
AC30, AJ29, AG23, AK16,	NC		
AF14, AK10, AJ4, AD3	G[3:0]_COL	Input w/ pull up	G[3:0]port – Collision Detected
AA28, AF29, AJ26, AJ21,	NC		
AH21, AH14, AG10, AH5, AC1	G[3:0]_RXCLK	Input w/ pull up	G[3:0]port – Receive Clock
AA29, AF27, AK26,	NC		

Ball No(s)	Symbol	I/O	Description
AK14, AF13, AH13, AK13, AH12, AJ12, AF12, AK12 AF8, AJ8, AK8, AG7, AG8, AJ7, AK7, AF7 AG4, AK1, AJ1, AJ2, AH2, AH1, AG1, AE5 AA5, AD4, AC2, Y5, AC3, AB2, W5, AB3	G3_TXD[7:0] G2_TXD[7:0] G1_TXD[7:0] G0_TXD[7:0]	Output	G[3:0]port – Transmit Data Bit [7:0]
AB28, Y26, AB29, AB30, AA27, AC28, AC29, AA26 AE26, AF28, AG30, AG28, AG27, AH29, AH28, AJ30 AK24, AJ24, AG24, AF24, AH24, AF23, AK23, AJ23 AJ19, AH19, AJ18, AH18, AF20, AK17, AG19, AJ17	NC		
AG13, AH8, AK2, AD2	G[3:0]_TX_EN	Output w/ pull up	G[3:0]port – Transmit Data Enable
Y27, AG29, AH25, AK19,	NC		
AJ14, AK9, AJ3, AB5	G[3:0]_TX_ER	Output w/ pull up	G[3:0]port – Transmit Error
AB27, AF30, AF25, AH20,	NC		
AH11, AG5, AG2, AB4	G[3:0]_ TXCLK	Output	G[3:0]port – Gigabit Transmit Clock
AD28, AH30, AK22, AH17,	NC		
PMA Interface (193) Gigabit Ethernet Access Port (PCS)			
AJ11, AJ6, AF3,AA4	GREF_CLK [3:0]	Input w/ pull up	Gigabit Reference Clock
AD29, AK30, AJ22, AG17,	NC		
AK15	CM_CLK	Input w/ pull up	Common Clock shared by port G[3:0]
AF17	IND/CM	Input w/ pull up	1: select GREF_CLK[3:0] as clock 0: select CM_CLK as clock for all port

Ball No(s)	Symbol	I/O	Description
AG16, AF16, AG15, AF18, AF15, AH15, AJ15, AG14	G3_RXD[7:0]	Input w/ pull up	G[3:0]port – PMA Receive Data Bit [7:0]
AG11, AJ10, AF11, AF10, AG9, AF9, AH9, AJ9	G2_RXD[7:0]		
AF6, AJ5, AF5, AG6, AK4, AF4, AK3, AH4	G1_RXD[7:0]		
AF1, AC5, AE1, AE2, AE3, AC4, AE4, AD1	G0_RXD[7:0]		
V26, W29, W30, Y28, W26, Y29, W27, Y30 AB26, AE27, AE28, AC27, AE29, AC26, AE30, AD26 AK27, AH27, AF26, AJ27, AH26, AK25, AG26, AJ25 AG22, AG21, AG20, AF22, AK21, AK20, AF21, AJ20	NC		
AH16, AH10, AK5, AD5	GP[3:0]_RX_D[8]	Input w/ pull down	G[3:0]port – PMA Receive Data Bit [**8**]
W28, AD30, AK28, AH22,	NC		
AF19, AG12, AK6, AF2	GP[3:0]_RX_D[9]	Input w/ pull up	G[3:0]port – PMA Receive Data Bit [**9**]
V27, AD27, AJ28, AH23,	NC		
AF14, AK10, AJ4, AD3	GP[3:0]_ RXCLK 1	Input w/ pull up	G[3:0]port – PMA Receive Clock 1
AA28, AF29, AJ26, AJ21,	NC		
AH14, AG10, AH5, AC1	GP[3:0]_RXCLK0	Input w/ pull up	G[3:0]port – PMA Receive Clock 0
AA29, AF27, AK26, AH21,	NC		
AK14, AF13, AH13, AK13, AH12, AJ12, AF12, AK12	G3_TXD[7:0]	Output	G[3:0]port – PMA Transmit Data Bit [7:0]
AF8, AJ8, AK8, AG7, AG8, AJ7, AK7, AF7	G2_TXD[7:0]		
AG4, AK1, AJ1, AJ2, AH2, AH1, AG1, AE5	G1_TXD[7:0]		
AA5, AD4, AC2, Y5, AC3, AB2, W5, AB3	G0_TXD[7:0]		

Ball No(s)	Symbol	I/O	Description
AB28, Y26, AB29, AB30, AA27, AC28, AC29, AA26 AE26, AF28, AG30, AG28, AG27, AH29, AH28, AJ30 AK24, AJ24, AG24, AF24, AH24, AF23, AK23, AJ23 AJ19, AH19, AJ18, AH18, AF20, AK17, AG19, AJ17	NC		
AG13, AH8, AK2, AD2	GP[3:0]_TXD[8]	Output w/ pull up	G[3:0]port – PMA Transmit Data Bit [8]
Y27, AG29, AH25, AK19,	NC		
AJ14, AK9, AJ3, AB5	GP[3:0]_TXD[9]	Output w/ pull up	G[3:0]port – PMA Transmit Data Bit [**9**]
AB27, AF30, AF25, AH20,	NC		
AH11, AG5, AG2, AB4	G[3:0]_ TXCLK	Output	G[3:0]port – PMA Gigabit Transmit Clock
AD28, AH30, AK22, AH17,	NC		
Test Facility (3)			
U29	T_MODE0	I/O-TS with pull up	Test – Set upon Reset, and provides NAND Tree test output during test mode Use external Pull up for normal operation
U28	T_MODE1	I/O-TS with pull up	Test – Set upon Reset, and provides NAND Tree test output during test mode Use external Pull up for normal operation
A3	SCAN_EN	Input w/ pull down	Enable test mode For normal operation leave it open
LED Interface (serial and parallel)			
R28, T26, R27, T27, U27, T28, T29, T30	T_D[7:0]/ LE_PD[7:0]	Output	While resetting, T_D[7,0] are in input mode and are used as strapping pins. Internal pullup LE_PD - Parallel Led data [7:0]

Ball No(s)	Symbol	I/O	Description
P27, R26, R30, R29	T_D[11:8]/ LE_PT[3:0]	Output	While resetting, T_D[11:8] are in input mode and are used as strapping pins. Internal pullup LED_PR[3:0] – Parallel Led port sel [3:0]
P26, P30, P29, P28,	T_D[15:12]/ LE_PT[7:4]	Output	While resetting, T_D[15:12] are in input mode and are used as strapping pins. Internal pullup LED_PR[7:4] – No Meaning
V29	LE_CLK0/ LE_PT[8]	Output	LE_CLK0 - LED Serial Interface Output Clock LE_PT[8] – Parallel Led port sel [8]
V30	LED_BLINK/ LE_DO/ LE_PT[9]	Output	While resetting, LED-BLINK is in input mode and is used as strapping pin. 1: No Blink, 0: Blink. Internal pullup. LE_DO - LED Serial Data Output Stream LE_PT[9] – Parallel Led port sel [9]
V28	LED_PM/ LE_SYNCO#	Output w/ pull up	While resetting, LED_PM is in input mode and is used as strapping pin. Internal pull up. 1: Enable parallel interface, 0: enable serial interface. LE_SYNCO# - LED Output Data Stream Envelop
System Clock, Power, and Ground Pins			
A16	S_CLK	Input	System Clock at 133 MHz
U26	S_RST#	Input - ST	Reset Input
U30	RESOUT#	Output	Reset PHY
B1	DEV_CFG[0]	Input w/ pull down	Not used
B28	DEV_CFG[1]	Input w/ pull down	Not used
AE7, AE9, F10, F21, F22, F9, G25, G6, J25, J6, K25, K6, AA25, AA6, AB25, AB6, AD25, AE10, AE21, AE22	VDD	Power core	+2.5 Volt DC Supply

Ball No(s)	Symbol	I/O	Description
V14, V15, V16, V17, V18, F16, F24, F25, F6, F7, N13, N14, N15, N16, N17, N18, P13, P14, P15, P16, P17, P18, R13, R14, R15, R16, R17, R18, R25, R6, T13, T14, T15, T16, T17, T18, T25, T6, U13, U14, U15, U16, U17, U18, V13, AD6, AE15, AE16, AE24, AE25, AE6, F15	VSS	Ground	Ground
A1, C28	AVDD	Power	Analog +2.5 Volt DC Supply
E5, E25	AVSS	Ground	Analog Ground
AE12, AE13, AE14, AE17, AE18, AE19, F12, F13, F14, F17, F18, F19, M25, M6, N25, N6, P25, P6, U25, U6, V25, V6, W25, W6	VDD33	Power I/O	+3.3 Volt DC Supply
Bootstrap Pins (Default= pull up, 1= pull up 0= pull down)			
AD2	G0_TX_EN	Default: PCS	Giga0 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AB5	G0_TX_ER	Default: PCS	
AK2	G1_TX_EN	Default: PCS	Giga1 Mode: G1_TXEN G1_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AJ3	G1_TXER	Default: PCS	
AH8	G2_TX_EN	Default: PCS	Giga2 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AK9	G2_TX_ER	Default: PCS	

Ball No(s)	Symbol	I/O	Description
AG13	G3_TX_EN	Default: PCS	Giga3 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AJ14	G3_TX_ER	Default: PCS	
After reset T_d[15:0] are used by the LED interface			
T30	T_d[0]	1	Giga link active status 0 – active low 1 – active high
T29	T_d[1]	1	Power saving 0 – No power saving 1 – Power saving Stop MAC clock if no MAC activity.
T28	T_d[2]	Must be pulled-down	Reserved - Must be pulled-down
U27	T_d[3]	1	Hot plug port module detection enable 0 – module detection enable 1 – module detection disable
T27	T_d[4]	Must be pulled-down	Reserved - Must be pulled-down
R27	T_d[5]	1	SRAM memory size 0 – 512K SRAM 1 – 256K SRAM
T26	T_d[6]	1	CPU Port mode 0 – 8 bit cpu data bus 1 – 16 bit cpu data bus
R28	T_d[7]	1	FDB memory depth 1– one memory layer 0 – two memory layers
W4, E21	La_a[20], Lb_a[20]	11	FDB memory size 11 - 2M per bank = 4M total 10 - 1M per bank = 2M total 0x - 512K per bank = 1M total
R29	T_d[8]	1	EEPROM installed 0 – EEPROM is installed 1 – EEPROM is not installed
R30	T_d[9]	1	MCT Aging enable 0 – MCT aging disable 1 – MCT aging enable

Ball No(s)	Symbol	I/O	Description
R26	T_d[10]	1	FCB handle aging enable 0 – FCB handle aging disable 1 – FCB handle aging enable
P27	T_d[11]	1	Timeout reset enable 0 – timeout reset disable 1 – timeout reset enable Issue reset if any state machine did not go back to idle for 5sec.
P26	T_d[15]	1	External RAM test 0 – Perform the infinite loop of ZBT RAM BIST. Debug test only 1 – Regular operation.
P30	T_d[14]	1	CPU installed 0 - CPU installed. 1 - CPU is not installed.

After reset P_d[8:0] are used by the CPU bus interface			
N30, N29, N28	P_d[2:0]	111	ZBT RAM la_clk turning 3'b000 - control by reg. **LCLKCR**[2:0] 3'b001 - delay by method # 0 3'b010 - delay by method # 1 3'b011 - delay by method # 2 3'b100 - delay by method # 3 3'b101 - delay by method # 4 3'b110 - delay by method # 5 3'b111 - delay by method # 6 USE METHOD 6 FOR NORMAL OPERATION. External pull up not required
M30, M29, M28	P_d[5:3]	111	No Use
L29, L28, N26	P_d[8:6]	111	SBRAM b_clk turning 3'b000 - control by **BCLKCR**[2:0] 3'b001 - delay by method # 0 3'b010 - delay by method # 1 3'b011 - delay by method # 2 3'b100 - delay by method # 3 3'b101 - delay by method # 4 3'b110 - delay by method # 5 3'b111 - delay by method # 6 USE METHOD 6 FOR NORMAL OPERATION. External pull up not required

Note:

# =	Active low signal
Input =	Input signal
In-ST =	Input signal with Schmitt-Trigger
Output =	Output signal (Tri-State driver)
Out-OD=	Output signal with Open-Drain driver
I/O-TS =	Input & Output signal with Tri-State driver
I/O-OD =	Input & Output signal with Open-Drain driver

12.3.2 Ball – Signal Description in Unmanaged Mode

Ball No(s)	Symbol	I/O	Description
K27, L27, K30, K29, K28	P_DATA[15:11]	I/O-TS with pull up	Not used – leave unconnected
L30	P_DATA[10]	I/O – TS with pull up	Trunk enable in unmanaged mode External pull up or unconnected – disable trunk group 0 and 1 External pull down – enable trunk group 0 and 1 See register TRUNK0_MODE for port selection and trunk enable.
N27	P_DATA[9]	I/O – TS with pull up	Trunk enable in unmanaged mode External pull up or unconnected – disable trunk group 2 and 3 External pull down – enable trunk group 2 and 3 See register TRUNK1_MODE for port selection and trunk enable
L29, L28, N26, M30, M29, M28, N30, N29, N28	P_DATA[8:0]	I/O – TS with pull up	Bootstrap function – See bootstrap section
J28	P_A[2]	Input	Not used – leave unconnected
H28	P_INT#	Output with internal weak pullup	Not used – leave unconnected
I^2C Interface (0) Note: In unmanaged mode, Use I^2C and Serial control interface to configure the system			
J27	SCL	Output	I^2C Data Clock
M26	SDA	I/O-TS with pull up	I^2C Data I/O
Serial Control Interface			
J29	PS_STROBE	Input with weak internal pull up	Serial Strobe Pin
L26	PS_DO	Input with weak internal pull up	Serial Data Input
J30	PS_DI (AUTOFD)	Output with pull up	Serial Data Output (AutoFD)
Frame Buffer Interface			

Ball No(s)	Symbol	I/O	Description
U1, U2, N4, U3, U4,T1,T2, N5, T3, T4, M4, R4, R3, R2, R1, M5, R5, L4, P3, P2, P1, N3,L5, N2, P5, N1, K4, M3, M2, M1, K5, L3, J5, K2, H4, K1, J4, J3, J2, H5, J1, H3, H2, H1, G3, G4, G5, G2, G1, F5, F4, F3, F2, F1, D3, E1,E2,E3, D2., E4, C3, D1, C1, B2	LA_D[63:0]	I/O-TS with pull up	Frame Bank A– Data Bit [63:0]
AA1, V5, AA2, AA3, Y1, V4, Y2, Y3, U5, W1, W2, W3, T5, V1, V2, P4, V3	LA_A[19:3]	Output	Frame Bank A – Address Bit [19:3]
W4	LA_A[20]	Output with pull up	Frame Bank A – Address Bit [20]
C2	LA_CLK	Output	Frame Bank A Clock Input
K3	LA_CS0#	Output with pull up	Frame Bank A Low Portion Chip Selection
L1	LA_CS1#	Output with pull up	Frame Bank A High Portion Chip Selection
L2	LA_RW#	Output with pull up	Frame Bank A Read/Write
D18, B18, C18, A17, E17, B17, C17, E16, D17, B16, E15, C16, D16, D15, E14, C15, B15, E13, A15, D14, C14, D13, B14, A14, C13, E12, B13, A13, D12, C12, B12, A12, A11, E10, C10, B10, E9, A10, D11, D10, D8, D9, C9, B9, A9, C8, B8, A8, C7, E7, D7, B7, E8, A7, D6, C6, E6, B6, A6, A5, B5, C5, B4,A4	NC	I/O-TS with pull up.	No Use
D22, D20, E20, D21, A21, D19, B21, C21, A20, B20, E19, C20, A19, B19, E18, C19, A18	NC	Output	No Use
E21	LB_A[20]	Output with pull up	Bootstrap Pin
D5	NC	Output	No Use
B11	NC	Output with pull up	No Use
E11	NC	Output with pull up	No Use
C11	NC	Output with pull up	No Use
Switch Database Interface			

Ball No(s)	Symbol	I/O	Description
E24,B27, D27, C27, A27, A28, B30, D28, E27, C30, D30, G26, E28, D29, E26, E29, H26, E30, J26, F30, F29, F28, F27, H27, G30, G29, K26, G27, G28, H30, H29, M27	B_D[31:0]	Output with pull up	Switch Database Domain – Data Bit [31:0]
C22, B22, A22, E22, C23, B23, A23, C24, D24, D23, B24, A24, E23, C25, C26, B25, A25	B_A[18:2]	Output	Switch Database Address (512K) – Address Bit [18:2]
C29	B_CLK	Output	Switch Database Clock Input
D25	B_ADSC#	Output with pull up	Switch Database Address Status Control
B26	B_WE#	Output with pull up	Switch Database Write Chip Select
A26	B_OE#	Output with pull up	Switch Database Read Chip Select
MII Management Interface			
AJ16	M_MDC	Output	MII Management Data Clock – (common for all MII Ports [3:0])
AG18	M_MDIO	I/O-TS with pull up	MII Management Data I/O – (common for all MII Ports –[3:0])) 2.5Mhz
GMII / MII Interface (193) Gigabit Ethernet Access Port			
AJ11, AJ6, AF3,AA4	GREF_CLK [3:0]	Input w/ pull up	Gigabit Reference Clock
AD29, AK30, AJ22, AG17,	NC		
AK15	CM_CLK	Input w/ pull up	Common Clock shared by port G[3:0]
AF17	IND/CM	Input w/ pull up	1: select GREF_CLK[3:0] as clock 0: select CM_CLK as clock for all ports
AJ13, AH7, AH3, AB1	MII TX CLK[3:0]	Input w/ pull up	
AA30, AK29, AG25, AK18,	NC		

Ball No(s)	Symbol	I/O	Description
AG16, AF16, AG15, AF18, AF15, AH15, AJ15, AG14 AG11, AJ10, AF11, AF10, AG9, AF9, AH9, AJ9 AF6, AJ5, AF5, AG6, AK4, AF4, AK3, AH4 AF1, AC5, AE1, AE2, AE3, AC4, AE4, AD1	G3_RXD[7:0] G2_RXD[7:0] G1_RXD[7:0] G0_RXD[7:0]	Input w/ pull up	G[3:0] port – Receive Data Bit [7:0]
V26, W29, W30, Y28, W26, Y29, W27, Y30 AB26, AE27, AE28, AC27, AE29, AC26, AE30, AD26 AK27, AH27, AF26, AJ27, AH26, AK25, AG26, AJ25 AG22, AG21, AG20, AF22, AK21, AK20, AF21, AJ20	NC		
AH16, AH10, AK5, AD5	G[3:0]_RX_DV	Input w/ pull down	G[3:0]port – Receive Data Valid
W28, AD30, AK28, AH22,	NC		
AF19, AG12, AK6, AF2	G[3:0]_RX_ER	Input w/ pull up	G[3:0]port – Receive Error
V27, AD27, AJ28, AH23,	NC		
AK11, AH6, AG3, Y4	G[3:0]_CRS/LINK	Input w/ pull down	G[3:0]port – Carrier Sense
AC30, AJ29, AG23, AK16,	NC		
AF14, AK10, AJ4, AD3	G[3:0]_COL	Input w/ pull up	G[3:0]port – Collision Detected
AA28, AF29, AJ26, AJ21,	NC		
AH14, AG10, AH5, AC1	G[3:0]_RXCLK	Input w/ pull up	G[3:0]port – Receive Clock
AA29, AF27, AK26, AH21,	NC		
AK14, AF13, AH13, AK13, AH12, AJ12, AF12, AK12 AF8, AJ8, AK8, AG7, AG8, AJ7, AK7, AF7 AG4, AK1, AJ1, AJ2, AH2, AH1, AG1, AE5 AA5, AD4, AC2, Y5, AC3, AB2, W5, AB3	G3_TXD[7:0] G2_TXD[7:0] G1_TXD[7:0] G0_TXD[7:0]	Output	G[3:0]port – Transmit Data Bit [7:0]

Ball No(s)	Symbol	I/O	Description
AB28, Y26, AB29, AB30, AA27, AC28, AC29, AA26 AE26, AF28, AG30, AG28, AG27, AH29, AH28, AJ30 AK24, AJ24, AG24, AF24, AH24, AF23, AK23, AJ23 AJ19, AH19, AJ18, AH18, AF20, AK17, AG19, AJ17	NC		
AG13, AH8, AK2, AD2	G[3:0]_TX_EN	Output w/ pull up	G[3:0]port – Transmit Data Enable
Y27, AG29, AH25, AK19,	NC		
AJ14, AK9, AJ3, AB5	G[3:0]_TX_ER	Output w/ pull up	G[3:0]port – Transmit Error
AB27, AF30, AF25, AH20,	NC		
AH11, AG5, AG2, AB4	G[3:0]_ TXCLK	Output	G[3:0]port – Gigabit Transmit Clock
AD28, AH30, AK22, AH17,	NC		
PMA Interface (193) Gigabit Ethernet Access Port (PCS)			
AJ11, AJ6, AF3,AA4	GREF_CLK [3:0]	Input w/ pull up	Gigabit Reference Clock
AD29, AK30, AJ22, AG17,	NC		
AK15	CM_CLK	Input w/ pull up	Common Clock shared by port G[3:0]
AF17	IND/CM	Input w/ pull up	1: select GREF_CLK[3:0] as clock 0: select CM_CLK as clock for all port
AG16, AF16, AG15, AF18, AF15, AH15, AJ15, AG14 AG11, AJ10, AF11, AF10, AG9, AF9, AH9, AJ9 AF6, AJ5, AF5, AG6, AK4, AF4, AK3, AH4 AF1, AC5, AE1, AE2, AE3, AC4, AE4, AD1	G3_RXD[7:0] G2_RXD[7:0] G1_RXD[7:0] G0_RXD[7:0]	Input w/ pull up	G[3:0]port – PMA Receive Data Bit [**7:0**]

Ball No(s)	Symbol	I/O	Description
V26, W29, W30, Y28, W26, Y29, W27, Y30 AB26, AE27, AE28, AC27, AE29, AC26, AE30, AD26 AK27, AH27, AF26, AJ27, AH26, AK25, AG26, AJ25 AG22, AG21, AG20, AF22, AK21, AK20, AF21, AJ20	NC		
AH16, AH10, AK5, AD5	G[3:0]_RX_D[8]	Input w/ pull down	G[3:0]port – PMA Receive Data Bit [**8**]
W28, AD30, AK28, AH22,	NC		
AF19, AG12, AK6, AF2	G[3:0]_RX_D[9]	Input w/ pull up	G[3:0]port – PMA Receive Data Bit [9]
V27, AD27, AJ28, AH23,	NC		
AF14, AK10, AJ4, AD3	G[3:0]_RXCLK1	Input w/ pull up	G[3:0]port – PMA Receive Clock 1
AA28, AF29, AJ26, AJ21,	NC		
AH14, AG10, AH5, AC1	G[3:0]_RXCLK0	Input w/ pull up	G[3:0]port – PMA Receive Clock 0
AA29, AF27, AK26, AH21,	NC		
AK14, AF13, AH13, AK13, AH12, AJ12, AF12, AK12 AF8, AJ8, AK8, AG7, AG8, AJ7, AK7, AF7 AG4, AK1, AJ1, AJ2, AH2, AH1, AG1, AE5 AA5, AD4, AC2, Y5, AC3, AB2, W5, AB3	G3_TXD[7:0] G2_TXD[7:0] G1_TXD[7:0] G0_TXD[7:0]	Output	G[3:0]port – PMA Transmit Data Bit [7:0]
AB28, Y26, AB29, AB30, AA27, AC28, AC29, AA26 AE26, AF28, AG30, AG28, AG27, AH29, AH28, AJ30 AK24, AJ24, AG24, AF24, AH24, AF23, AK23, AJ23 AJ19, AH19, AJ18, AH18, AF20, AK17, AG19, AJ17	NC		
AG13, AH8, AK2, AD2	G[3:0]_TXD[8]	Output w/ pull up	G[3:0]port – PMA Transmit Data Bit [**8**]
Y27, AG29, AH25, AK19,	NC		
AJ14, AK9, AJ3, AB5	G[3:0]_TX_D[9]	Output w/ pull up	G[3:0]port – PMA Transmit Data Bit [9]
AB27, AF30, AF25, AH20,	NC		

Ball No(s)	Symbol	I/O	Description
AH11, AG5, AG2, AB4	G[3:0]_ TXCLK	Output	G[3:0]port – PMA Gigabit Transmit Clock
AD28, AH30, AK22, AH17,	NC		
Test Facility (3)			
U29	T_MODE0	I/O-TS with pull up	Test – Set upon Reset, and provides NAND Tree test output during test mode Use external Pull up for normal operation
U28	T_MODE1	I/O-TS with pull up	Test – Set upon Reset, and provides NAND Tree test output during test mode Use external Pull up for normal operation
A3	SCAN_EN	Input w/ pull down	Enable test mode For normal operation leave it open
LED Interface (serial and parallel)			
R28, T26, R27, T27, U27, T28, T29, T30	T_D[7:0]/ LE_PD[7:0]	Output	While resetting, T_D[7,0] are in input mode and are used as strapping pins. Internal pullup LE_PD - Parallel Led data [7:0]
P27, R26, R30, R29	T_D[11:8]/ LE_PT[3:0]	Output	While resetting, T_D[11:8] are in input mode and are used as strapping pins. Internal pullup LED_PR[3:0] – Parallel Led port sel [3:0]
P26, P30, P29, P28,	T_D[15:12]/ LE_PT[7:4]	Output	While resetting, T_D[15:12] are in input mode and are used as strapping pins. Internal pullup LED_PR[7:4] – Meanless
V29	LE_CLK0/ LE_PT[8]	Output	LE_CLK0 – LED Serial Interface Output Clock LE_PT[8] – Parallel Led port sel [8]
V30	LED_BLINK/ LE_DO/ LE_PT[9]	Output	While resetting, LED-BLINK is in input mode and is used as strapping pin. 1: No Blink, 0: Blink. Internal pullup. LE_DO - LED Serial Data Output Stream LE_PT[9] – Parallel Led port sel [9]

Ball No(s)	Symbol	I/O	Description
V28	LED_PM/ LE_SYNCO#	Output w/ pull up	While resetting, LED_PM is in input mode and is used as strapping pin. Internal pull up. 1: Enable parallel interface, 0: enable serial interface. LE_SYNCO# - LED Output Data Stream Envelop
System Clock, Power, and Ground Pins			
A16	S_CLK	Input	System Clock at 133 MHz
U26	S_RST#	Input – ST	Reset Input
U30	RESOUT#	Output	Reset PHY
B1	DEV_CFG[0]	Input w/ pull down	Not used
B28	DEV_CFG[1]	Input w/ pull down	Not used
AE7, AE9, F10, F21, F22, F9, G25, G6, J25, J6, K25, K6, AA25, AA6, AB25, AB6, AD25, AE10, AE21, AE22	VDD	Power core	+2.5 Volt DC Supply
V14, V15, V16, V17, V18, F16, F24, F25, F6, F7, N13, N14, N15, N16, N17, N18, P13, P14, P15, P16, P17, P18, R13, R14, R15, R16, R17, R18, R25, R6, T13, T14, T15, T16, T17, T18, T25, T6, U13, U14, U15, U16, U17, U18, V13, AD6, AE15, AE16, AE24, AE25, AE6, F15	VSS	Ground	Ground
A1, C28	AVDD	Power	Analog +2.5 Volt DC Supply
E5, E25	AVSS	Ground	Analog Ground
AE12, AE13, AE14, AE17, AE18, AE19, F12, F13, F14, F17, F18, F19, M25, M6, N25, N6, P25, P6, U25, U6, V25, V6, W25, W6	VDD33	Power I/O	+3.3 Volt DC Supply
Bootstrap Pins (Default= pull up, 1= pull up 0= pull down)			
AD2	G0_TX_EN	Default: PCS	Giga0 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS

Ball No(s)	Symbol	I/O	Description
AB5	G0_TX_ER	Default: PCS	
AK2	G1_TX_EN	Default: PCS	Giga1 Mode: G1_TXEN G1_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AJ3	G1_TXER	Default: PCS	
AH8	G2_TX_EN	Default: PCS	Giga2 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AK9	G2_TX_ER	Default: PCS	
AG13	G3_TX_EN	Default: PCS	Giga3 Mode: G0_TXEN G0_TXER 0 0 MII 0 1 Invalid 1 0 GMII 1 1 PCS
AJ14	G3_TX_ER	Default: PCS	
After reset T_d[15:0] are used by the LED interface			
T30	T_d[0]	1	Giga link active status 0 – active low 1 – active high
T29	T_d[1]	1	Power saving 0 – No power saving 1 – Power saving Stop MAC clock if no MAC activity.
T28	T_d[2]	Must be pulled-down	Reserved - Must be pulled-down
U27	T_d[3]	1	Hot plug port module detection enable 0 – module detection enable 1 – module detection disable
T27	T_d[4]	Must be pulled-down	Reserved - Must be pulled-down
R27	T_d[5]	1	SRAM memory size 0 – 512K SRAM 1 – 256K SRAM

Ball No(s)	Symbol	I/O	Description
T26	T_d[6]	1	CPU Port mode 0 – 8 bit cpu data bus 1 – 16 bit cpu data bus
R28	T_d[7]	1	FDB memory depth 1– one memory layer 0 – two memory layers
W4, E21	La_a[20], Lb_a[20]	11	FDB memory size 11 - 2M per bank = 4M total 10 - 1M per bank = 2M total 0x - 512K per bank = 1M total
R29	T_d[8]	1	EEPROM installed 0 – EEPROM is installed 1 – EEPROM is not installed
R30	T_d[9]	1	MCT Aging enable 0 – MCT aging disable 1 – MCT aging enable
R26	T_d[10]	1	FCB handle aging enable 0 – FCB handle aging disable 1 – FCB handle aging enable
P27	T_d[11]	1	Timeout reset enable 0 – timeout reset disable 1 – timeout reset enable Issue reset if any state machine did not go back to idle for 5sec.
P28,P29	T_d[13:12]		Reserved
P30	T_d[14]	1	CPU installed 0 – CPU installed. 1 – CPU is not installed.
P26	T_d[15]	1	External RAM test 0 – Perform the infinite loop of ZBT RAM BIST. Debug test only 1 – Regular operation.
N30, N29, N28	P_d[2:0]	111	ZBT RAM la_clk turning 3’b000 - control by reg. **LCLKCR**[2:0] 3’b001 - delay by method # 0 3’b010 - delay by method # 1 3’b011 - delay by method # 2 3’b100 - delay by method # 3 3’b101 - delay by method # 4 3’b110 - delay by method # 5 3’b111 - delay by method # 6 – USE THIS METHOD
M30, M29, M28	P_d[5:3]	111	No Use

Ball No(s)	Symbol	I/O	Description
L29, L28, N26	P_d[8:6]	111	SBRAM b_clk turning 3'b000 - control by **BCLKCR**[2:0] 3'b001 - delay by method # 0 3'b010 - delay by method # 1 3'b011 - delay by method # 2 3'b100 - delay by method # 3 3'b101 - delay by method # 4 3'b110 - delay by method # 5 3'b111 - delay by method # 6– USE THIS METHOD

Note:

# =	Active low signal
Input =	Input signal
In-ST =	Input signal with Schmitt-Trigger
Output =	Output signal (Tri-State driver)
Out-OD=	Output signal with Open-Drain driver
I/O-TS =	Input & Output signal with Tri-State driver
I/O-OD =	Input & Output signal with Open-Drain driver

12.4 Ball Signal Name

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
A1	AVDD	M1	LA_D[34]	Y2	LA_A[13]
B1	DEV_CFG[0]	M2	LA_D[35]	V4	LA_A[14]
B2	LA_D[0]	M3	LA_D[36]	Y1	LA_A[15]
C2	LA_CLK	K4	LA_D[37]	AA3	LA_A[16]
C1	LA_D[1]	N1	LA_D[38]	AA2	LA_A[17]
D1	LA_D[2]	P5	LA_D[39]	V5	LA_A[18]
C3	LA_D[3]	N2	LA_D[40]	AA1	LA_A[19]
E4	LA_D[4]	L5	LA_D[41]	W4	LA_A[20]
D2	LA_D[5]	N3	LA_D[42]	Y4	G0_CRS/LINK
E3	LA_D[6]	P1	LA_D[43]	AA4	GREF_CLK[0]
E2	LA_D[7]	P2	LA_D[44]	AB4	G0_TXCLK
E1	LA_D[8]	P3	LA_D[45]	AB3	G0_TXD[0]
D3	LA_D[9]	L4	LA_D[46]	W5	G0_TXD[1]
F1	LA_D[10]	R5	LA_D[47]	AB2	G0_TXD[2]
F2	LA_D[11]	M5	LA_D[48]	AB1	MII_TX_CLK[0]
F3	LA_D[12]	R1	LA_D[49]	AC3	G0_TXD[3]
F4	LA_D[13]	R2	LA_D[50]	Y5	G0_TXD[4]
F5	LA_D[14]	R3	LA_D[51]	AC2	G0_TXD[5]
G1	LA_D[15]	R4	LA_D[52]	AC1	G0_RXCLK
G2	LA_D[16]	M4	LA_D[53]	AD3	G0_COL
G5	LA_D[17]	T4	LA_D[54]	AD4	G0_TXD[6]
G4	LA_D[18]	T3	LA_D[55]	AA5	G0_TXD[7]
G3	LA_D[19]	N5	LA_D[56]	AD2	G0_TX_EN
H1	LA_D[20]	T2	LA_D[57]	AB5	G0_TX_ER
H2	LA_D[21]	T1	LA_D[58]	AD1	G0_RXD[0]
H3	LA_D[22]	U4	LA_D[59]	AE4	G0_RXD[1]
J1	LA_D[23]	U3	LA_D[60]	AC4	G0_RXD[2]
H5	LA_D[24]	N4	LA_D[61]	AE3	G0_RXD[3]
J2	LA_D[25]	U2	LA_D[62]	AE2	G0_RXD[4]
J3	LA_D[26]	U1	LA_D[63]	AE1	G0_RXD[5]

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
J4	LA_D[27]	V3	LA_A[3]	AC5	G0_RXD[6]
K1	LA_D[28]	P4	LA_A[4]	AF1	G0_RXD[7]
H4	LA_D[29]	V2	LA_A[5]	AD5	G0_RX_DV
K2	LA_D[30]	V1	LA_A[6]	AF2	G0_RX_ER
J5	LA_D[31]	T5	LA_A[7]	AF3	GREF_CLK[1]
K3	LA_CS0#	W3	LA_A[8]	AG2	G1_TXCLK
L1	LA_CS1#	W2	LA_A[9]	AG3	G1_CRS/LINK
L2	LA_RW#	W1	LA_A[10]	AE5	G1_TXD[0]
L3	LA_D[32]	U5	LA_A[11]	AG1	G1_TXD[1]
K5	LA_D[33]	Y3	LA_A[12]	AH1	G1_TXD[2]
AH2	G1_TXD[3]	AG10	G2_RXCLK	AG19	NC
AJ2	G1_TXD[4]	AK10	G2_COL	AK17	NC
AJ1	G1_TXD[5]	AJ10	G2_RXD[6]	AF20	NC
AK1	G1_TXD[6]	AG11	G2_RXD[7]	AH18	NC
AG4	G1_TXD[7]	AH10	G2_RX_DV	AJ18	NC
AK2	G1_TX_EN	AG12	G2_RX_ER	AK18	NC
AH3	MII_TX_CLK[1]	AK11	G3_CRS/LINK	AH19	NC
AJ3	G1_TX_ER	AJ11	GREF_CLK[3]	AJ19	NC
AH4	G1_RXD[0]	AH11	G3_TXCLK	AK19	NC
AK3	G1_RXD[1]	AK12	G3_TXD[0]	AH20	NC
AF4	G1_RXD[2]	AF12	G3_TXD[1]	AJ20	NC
AK4	G1_RXD[3]	AJ12	G3_TXD[2]	AF21	NC
AH5	G1_RXCLK	AH12	G3_TXD[3]	AK20	NC
AJ4	G1_COL	AK13	G3_TXD[4]	AH21	NC
AG6	G1_RXD[4]	AJ13	MII_TX_CLK[3]	AJ21	NC
AF5	G1_RXD[5]	AH13	G3_TXD[5]	AK21	NC
AJ5	G1_RXD[6]	AF13	G3_TXD[6]	AF22	NC
AF6	G1_RXD[7]	AK14	G3_TXD[7]	AG20	NC
AK5	G1_RX_DV	AG13	G3_TX_EN	AG21	NC
AK6	G1_RX_ER	AJ14	G3_TX_ER	AG22	NC
AJ6	GREF_CLK[2]	AH14	G3_RXCLK	AH22	NC

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AG5	G2_TXCLK	AF14	G3_COL	AJ22	NC
AH6	G2_CRS/LINK	AG14	G3_RXD[0]	AK22	NC
AF7	G2_TXD[0]	AK15	CM_CLK	AH23	NC
AK7	G2_TXD[1]	AF17	IND_CM	AG23	NC
AJ7	G2_TXD[2]	AJ15	G3_RXD[1]	AJ23	NC
AG8	G2_TXD[3]	AH15	G3_RXD[2]	AK23	NC
AG7	G2_TXD[4]	AF15	G3_RXD[3]	AF23	NC
AH7	MII_TX_CLK[2]	AF18	G3_RXD[4]	AH24	NC
AK8	G2_TXD[5]	AG15	G3_RXD[5]	AF24	NC
AJ8	G2_TXD[6]	AF16	G3_RXD[6]	AG24	NC
AF8	G2_TXD[7]	AG16	G3_RXD[7]	AJ24	NC
AH8	G2_TX_EN	AH16	G3_RX_DV	AK24	NC
AK9	G2_TX_ER	AF19	G3_RX_ER	AG25	NC
AJ9	G2_RXD[0]	AJ16	M_MDC	AH25	NC
AH9	G2_RXD[1]	AG18	M_MDIO	AF25	NC
AF9	G2_RXD[2]	AK16	NC	AJ25	NC
AG9	G2_RXD[3]	AG17	NC	AG26	NC
AF10	G2_RXD[4]	AH17	NC	AK25	NC
AF11	G2_RXD[5]	AJ17	NC	AK26	NC
AJ26	NC	AA27	NC	P29	T_D[13]
AH26	NC	AB30	NC	P30	T_D[14]
AJ27	NC	AB29	NC	P26	T_D[15]
AF26	NC	Y26	NC	N28	P_D[0]
AH27	NC	AB28	NC	N29	P_D[1]
AK27	NC	Y27	NC	N30	P_D[2]
AK28	NC	AB27	NC	M28	P_D[3]
AJ28	NC	AA30	NC	M29	P_D[4]
AJ29	NC	AA29	NC	M30	P_D[5]
AK29	NC	AA28	NC	N26	P_D[6]
AK30	NC	Y30	NC	L28	P_D[7]
AJ30	NC	W27	NC	L29	P_D[8]

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AH28	NC	Y29	NC	N27	P_D[9]
AH29	NC	W26	NC	L30	P_D[10]
AG27	NC	Y28	NC	K28	P_D[11]
AG28	NC	W30	NC	K29	P_D[12]
AH30	NC	W29	NC	K30	P_D[13]
AG30	NC	V26	NC	L27	P_D[14]
AF28	NC	W28	NC	K27	P_D[15]
AE26	NC	V27	NC	M26	P_A[0]
AG29	NC	V30	LE_DO	J27	P_A[1]
AF27	NC	V29	LE_CLK0	J28	P_A[2]
AF29	NC	V28	LE_SYNCO#	J29	P_WE#
AF30	NC	U26	S_RST#	J30	P_RD#
AD26	NC	U30	RESOUT#	L26	P_CS#
AE30	NC	U29	T_MODE[0]	H28	P_INT#
AC26	NC	U28	T_MODE[1]	M27	B_D[0]
AE29	NC	T30	T_D[0]	H29	B_D[1]
AC27	NC	T29	T_D[1]	H30	B_D[2]
AE28	NC	T28	T_D[2]	G28	B_D[3]
AE27	NC	U27	T_D[3]	G27	B_D[4]
AB26	NC	T27	T_D[4]	K26	B_D[5]
AD30	NC	R27	T_D[5]	G29	B_D[6]
AD29	NC	T26	T_D[6]	G30	B_D[7]
AD27	NC	R28	T_D[7]	H27	B_D[8]
AD28	NC	R29	T_D[8]	F27	B_D[9]
AC30	NC	R30	T_D[9]	F28	B_D[10]
AA26	NC	R26	T_D[10]	F29	B_D[11]
AC29	NC	P27	T_D[11]	F30	B_D[12]
AC28	NC	P28	T_D[12]	J26	B_D[13]
E30	B_D[14]	A23	B_A[12]	E14	NC
H26	B_D[15]	B23	B_A[13]	C15	NC
E29	B_D[16]	C23	B_A[14]	B15	NC

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
E26	B_D[17]	E22	B_A[15]	E13	NC
D29	B_D[18]	A22	B_A[16]	A15	NC
E28	B_D[19]	B22	B_A[17]	D14	NC
G26	B_D[20]	C22	B_A[18]	C14	NC
D30	B_D[21]	E21	LB_A[20]	D13	NC
C30	B_D[22]	D22	NC	B14	NC
E27	B_D[23]	D20	NC	A14	NC
C29	B_CLK	E20	NC	C13	NC
D28	B_D[24]	D21	NC	E12	NC
B30	B_D[25]	A21	NC	B13	NC
F26	NC1	D19	NC	A13	NC
D26	NC2	B21	NC	D12	NC
A30	NC3	C21	NC	C12	NC
A29	NC4	A20	NC	B12	NC
B29	NC5	B20	NC	A12	NC
E25	AGND	E19	NC	C11	NC
B28	DEV_CFG[1]	C20	NC	E11	NC
C28	AVDD	A19	NC	B11	NC
A28	B_D[26]	B19	NC	A11	NC
A27	B_D[27]	E18	NC	E10	NC
C27	B_D[28]	C19	NC	C10	NC
D27	B_D[29]	A18	NC	B10	NC
B27	B_D[30]	D18	NC	E9	NC
E24	B_D[31]	B18	NC	A10	NC
D25	B_ADSC#	C18	NC	D11	NC
B26	B_WE#	A17	NC	D10	NC
A26	B_OE#	E17	NC	D8	NC
A25	B_A[2]	B17	NC	D9	NC
B25	B_A[3]	C17	NC	C9	NC
C26	B_A[4]	E16	NC	B9	NC
C25	B_A[5]	D17	NC	A9	NC

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
E23	B_A[6]	A16	S_CLK	C8	NC
A24	B_A[7]	B16	NC	B8	NC
B24	B_A[8]	E15	NC	A8	NC
D23	B_A[9]	C16	NC	C7	NC
D24	B_A[10]	D16	NC	E7	NC
C24	B_A[11]	D15	NC	D7	NC
B7	NC	P15	VSS	AE7	VDD
E8	NC	P16	VSS	AE9	VDD
A7	NC	P17	VSS	F10	VDD
D6	NC	P18	VSS	F21	VDD
C6	NC	R13	VSS	F22	VDD
E6	NC	R14	VSS	F9	VDD
B6	NC	R15	VSS	G25	VDD
A6	NC	R16	VSS	G6	VDD
A5	NC	R17	VSS	J25	VDD
B5	NC	R18	VSS	J6	VDD
C5	NC	R25	VSS	K25	VDD
B4	NC	R6	VSS	K6	VDD
D5	NC	T13	VSS	AE12	VD33
A4	NC	T14	VSS	AE13	VD33
A3	SCAN_EN	T15	VSS	AE14	VD33
E5	AGND	T16	VSS	AE17	VD33
C4	NC6	T17	VSS	AE18	VD33
B3	NC7	T18	VSS	AE19	VD33
D4	NC8	T25	VSS	F12	VD33
A2	NC9	T6	VSS	F13	VD33
AD6	VSS	U13	VSS	F14	VD33
AE15	VSS	U14	VSS	F17	VD33
AE16	VSS	U15	VSS	F18	VD33
AE24	VSS	U16	VSS	F19	VD33
AE25	VSS	U17	VSS	M25	VD33

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AE6	VSS	U18	VSS	M6	VD33
F15	VSS	V13	VSS	N25	VD33
F16	VSS	V14	VSS	N6	VD33
F24	VSS	V15	VSS	P25	VD33
F25	VSS	V16	VSS	P6	VD33
F6	VSS	V17	VSS	U25	VD33
F7	VSS	V18	VSS	U6	VD33
N13	VSS	AA25	VDD	V25	VD33
N14	VSS	AA6	VDD	V6	VD33
N15	VSS	AB25	VDD	W25	VD33
N16	VSS	AB6	VDD	W6	VD33
N17	VSS	AD25	VDD		
N18	VSS	AE10	VDD		
P13	VSS	AE21	VDD		
P14	VSS	AE22	VDD		

12.5 AC/DC Timing

12.5.1 Absolute Maximum Ratings

Storage Temperature -65C to +150C

Operating Temperature -40°C to +85°C

Supply Voltage VDD33 with Respect to VSS +3.0 V to +3.6 V

Supply Voltage VDD with Respect to VSS +2.38 V to +2.75 V

Voltage on Input Pins -0.5 V to (VDD33 + 3.3 V)

Caution: Stress above those listed may damage the device. Exposure to the Absolute Maximum Ratings for extended periods may affect device reliability. Functionality at or above these limits is not implied.

12.5.2 DC Electrical Characteristics

VDD33 = 3.0 V to 3.6 V (3.3v +/- 10%) $T_{AMBIENT}$ = -40°C to +85°C

VDD = 2.5V +10% - 5%

12.5.3 Recommended Operation Conditions

Symbol	Parameter Description	Preliminary			Unit
		Min	Type	Max	
f_{osc}	Frequency of Operation		133		MHz
I_{DD1}	Supply Current – @ 133 MHz (VDD33 = 3.3V)	680		850	mA
I_{DD2}	Supply Current – @ 133 MHz (VDD = 2.5V)	1300		1500	mA
V_{OH}	Output High Voltage (CMOS)	VDD33 - 0.5			V
V_{OL}	Output Low Voltage (CMOS)			0.5	V
V_{IH-TTL}	Input High Voltage (TTL 5V tolerant)	VDD33 x 70%		VDD33 + 2.0	V
V_{IL-TTL}	Input Low Voltage (TTL 5V tolerant)			VDD33 x 30%	V
I_{IH-5VT}	Input Leakage Current (0.1 V < V_{IN} < VDD33)			10	µA
C_{IN}	Input Capacitance			5	pF
C_{OUT}	Output Capacitance			5	pF
$C_{I/O}$	I/O Capacitance			7	pF

12.5.4 Typical CPU Timing Diagram for a CPU Write Cycle

Description		(SCLK=133Mhz)		
Write Cycle	Symbol	Min (ns)	Max (ns)	
Write Set up Time	T_{WS}	10		
Write Active Time	T_{WA}	15		At least 2 SCLK
Write Hold Time	T_{WH}	2		
Write Recovery time	T_{WR}	22.5		At least 3 SCLK
Data Set Up time	T_{DS}	10		
Data Hold time	T_{DH}	2		

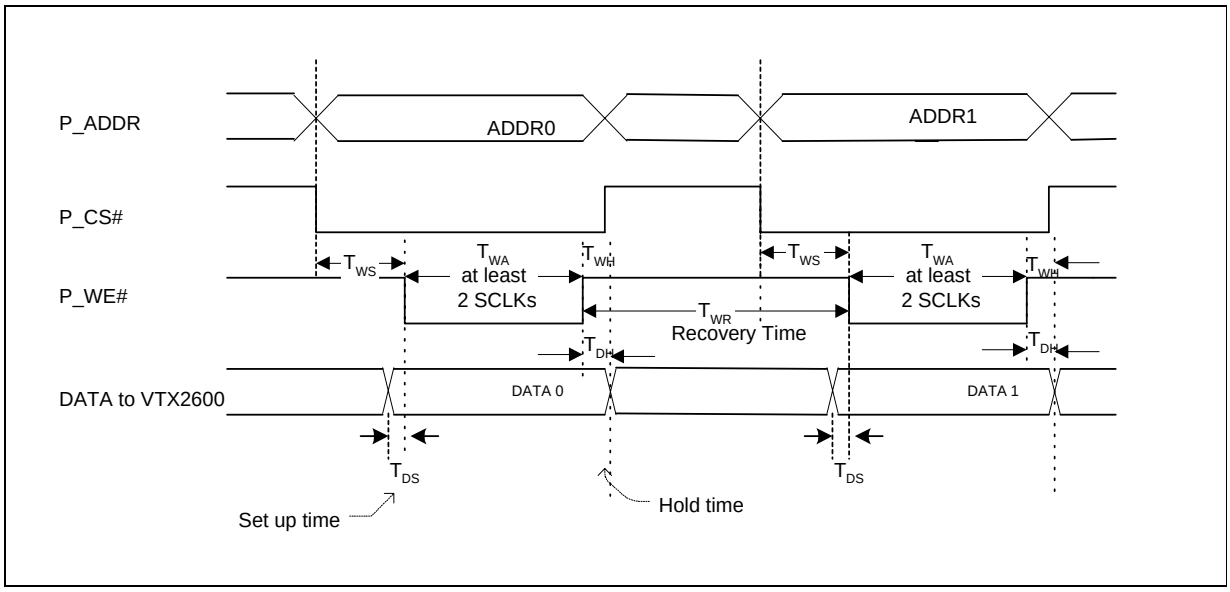


Figure 7 - Typical CPU Timing Diagram for a CPU Write Cycle

12.5.5 Typical CPU Timing Diagram for a CPU Read Cycle

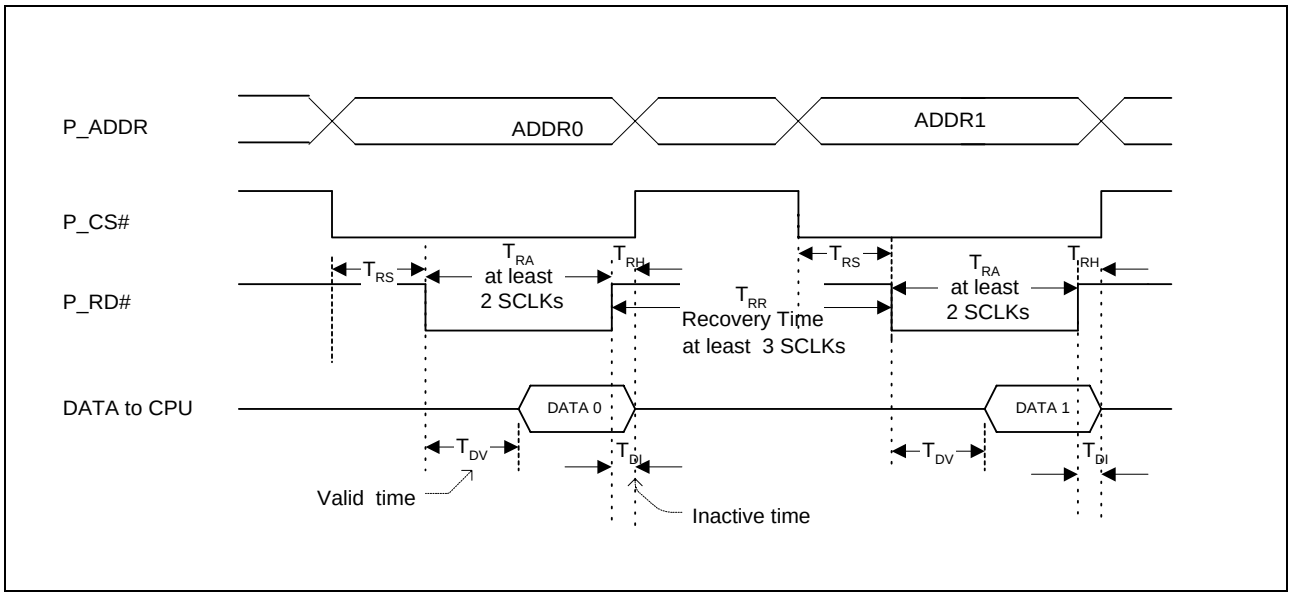


Figure 8 - Typical CPU Timing Diagram for a CPU Read Cycle

Description		(SCLK=133Mhz)		
Read Cycle	**Symbol**	**Min (ns)**	**Max (ns)**	
Read Set up Time	T_{RS}	10		
Read Active Time	T_{RA}	15		At least 2 SCLK
Read Hold Time	T_{RH}	2		
Read Recovery time	T_{RR}	22.5		At least 3 SCLK
Data Valid time	T_{DS}		10	
Data Inactive time	T_{DI}		2	

12.6 Local Frame Buffer ZBT SRAM Memory Interface

12.6.1 Local ZBT SRAM Memory Interface A

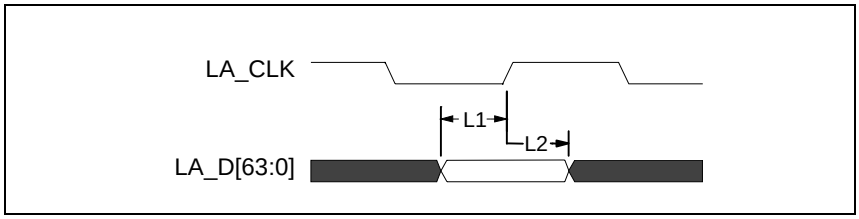


Figure 9 - Local Memory Interface – Input setup and hold timing

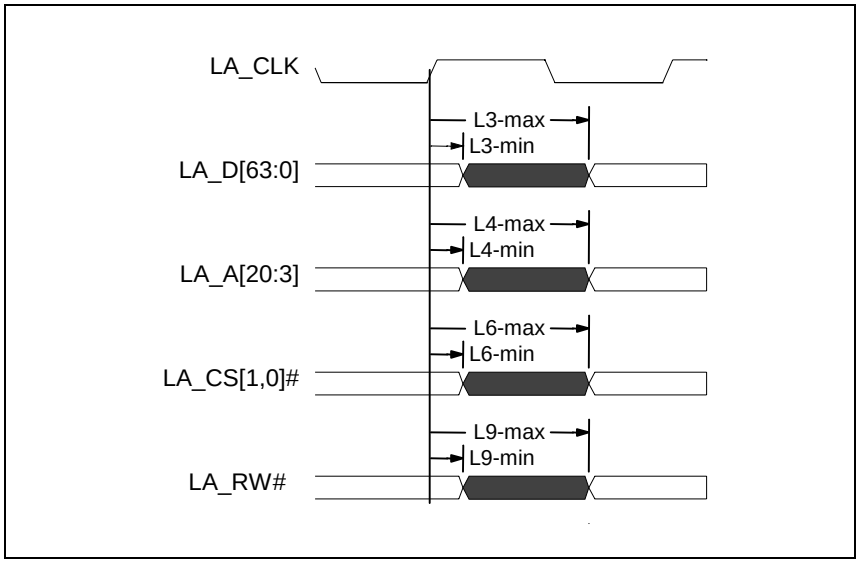


Figure 10 - Local Memory Interface - Output valid delay timing

Symbol	Parameter	(SCLK= 133MHz)		Note:
		Min (ns)	Max (ns)	
L1	LA_D[63:0] input set-up time	2.5		
L2	LA_D[63:0] input hold time	1		
L3	LA_D[63:0] output valid delay	3	5	C_L = 25pf
L4	LA_A[20:3] output valid delay	3	5	C_L = 30pf
L6	LA_CS[1:0]# output valid delay	3	5	C_L = 30pf
L9	LA_WE# output valid delay	3	5	C_L = 25pf

Table 6- AC Characteristics – Local frame buffer ZBT-SRAM Memory Interface A

12.7 Local Switch Database SBRAM Memory Interface

12.7.1 Local SBRAM Memory Interface

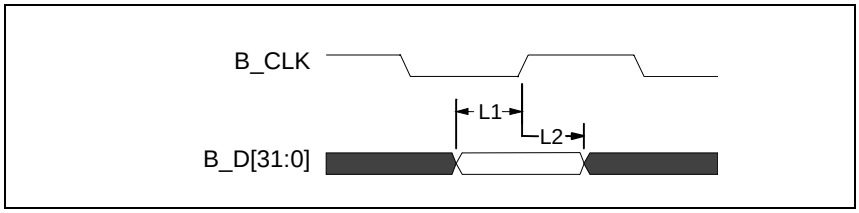


Figure 11 - Local Memory Interface – Input setup and hold timing

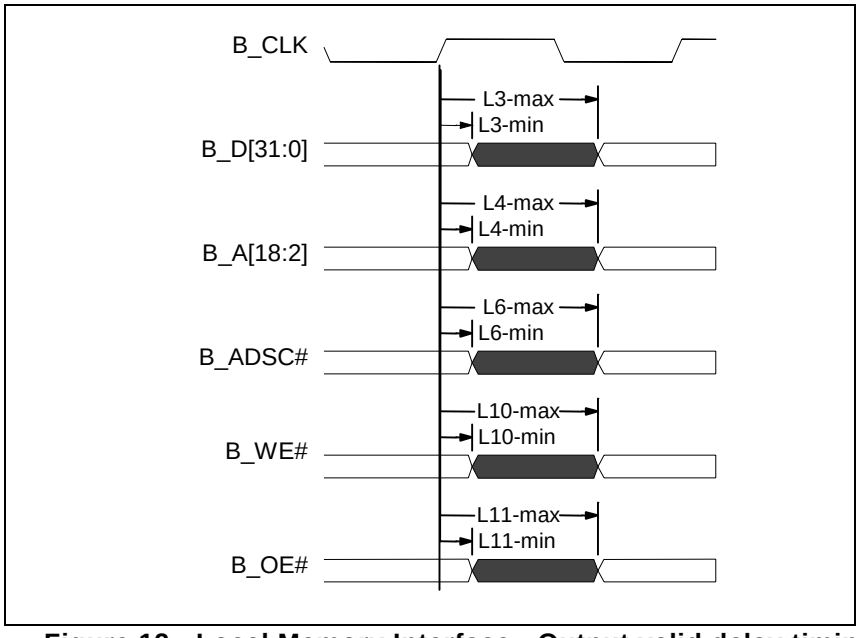


Figure 12 - Local Memory Interface - Output valid delay timing

Symbol	Parameter	(SCLK= 133MHz)		Note:
		Min (ns)	Max (ns)	
L1	B_D[31:0] input set-up time	2.5		
L2	B_D[31:0] input hold time	1		
L3	B_D[31:0] output valid delay	3	5	C_L = 25pf
L4	B_A[18:2] output valid delay	3	5	C_L = 30pf
L6	B_ADSC# output valid delay	3	5	C_L = 30pf
L10	B_WE# output valid delay	3	5	C_L = 25pf
L11	B_OE# output valid delay	3	4	C_L = 25pf

Table 7- AC Characteristics – Local Switch Database SBRAM Memory Interface

12.8 AC Characteristics

12.8.1 Media Independent Interface

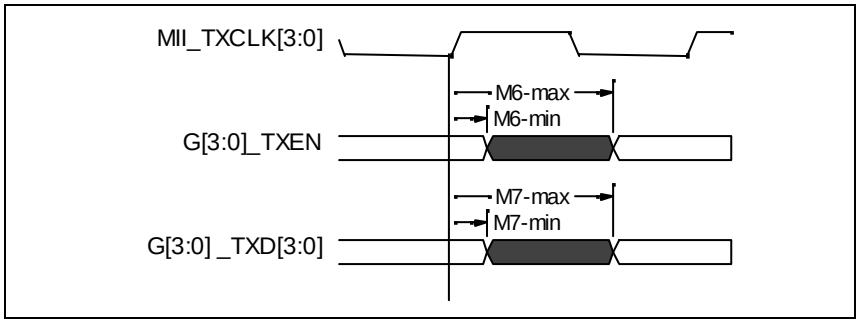


Figure 13 - AC Characteristics – Media Independent Interface

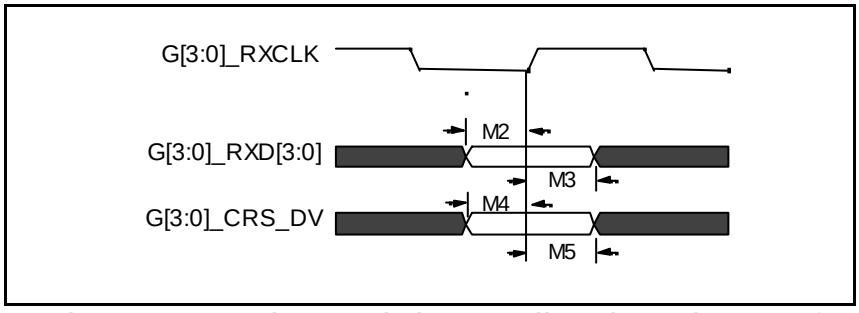


Figure 14 - AC Characteristics – Media Independent Interface

Symbol	Parameter	(MII_TXCLK & G_RXCLK = 25MHz)		Note:
		Min (ns)	Max (ns)	
M2	G[3:0]_RXD[3:0] Input Setup Time	4		
M3	G[3:0]_RXD[3:0] Input Hold Time	1		
M4	G[3:0]_CRS_DV Input Setup Time	4		
M5	G[3:0]_CRS_DV Input Hold Time	1		
M6	G[3:0]_TXEN Output Delay Time	3	11	C_L = 20 pF
M7	G[3:0]_TXD[3:0] Output Delay Time	3	11	C_L = 20 pF

Table 8- AC Characteristics – Media Independent Interface

12.8.2 Gigabit Media Independent Interface

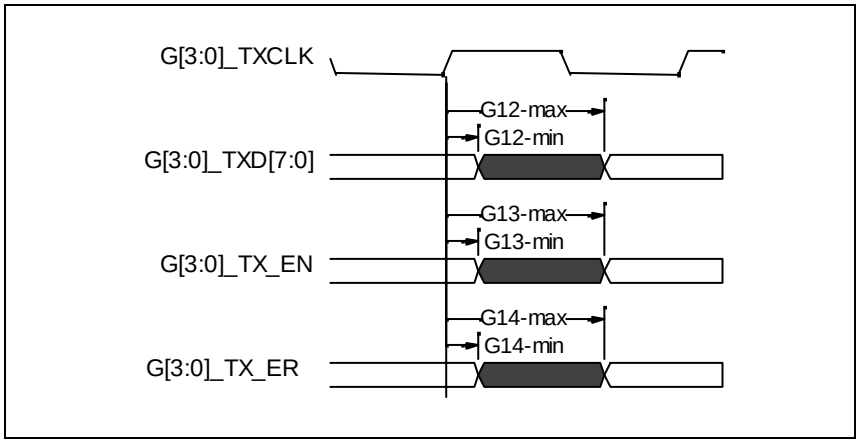


Figure 15 - AC Characteristics- GMII

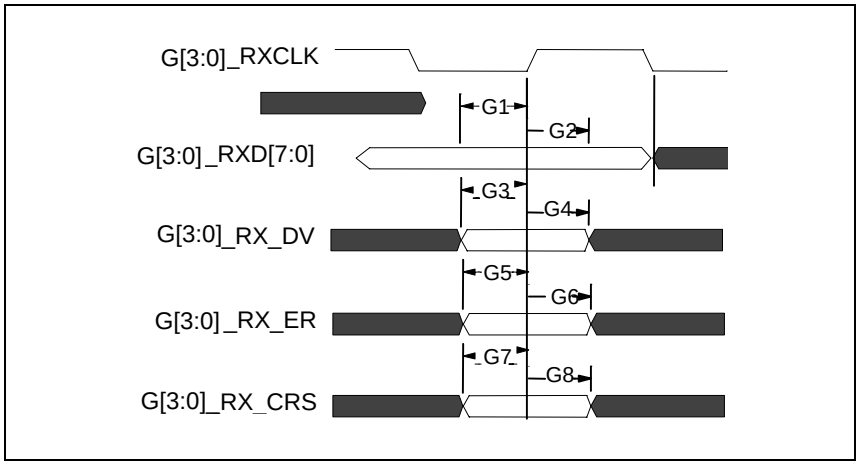


Figure 16 - AC Characteristics – Gigabit Media Independent Interface

Symbol	Parameter	(G_RCLK & G_REFCLK = 125MHz)		Note:
		Min (ns)	Max (ns)	
G1	**G[3:0]**_RXD[7:0] Input Setup Times	2		
G2	**G[3:0]**_RXD[7:0] Input Hold Times	1		
G3	**G[3:0]**_RX_DV Input Setup Times	2		
G4	**G[3:0]**_RX_DV Input Hold Times	1		
G5	**G[3:0]**_RX_ER Input Setup Times	2		
G6	**G[3:0]**_RX_ER Input Hold Times	1		
G7	**G[3:0]**_CRS Input Setup Times	2		
G8	**G[3:0]**_CRS Input Hold Times	1		
G12	**G[3:0]**_TXD[7:0] Output Delay Times	1	5	C_L = 20pf
G13	**G[3:0]**_TX_EN Output Delay Times	1	5	C_L = 20pf
G14	**G[3:0]**_TX_ER Output Delay Times	1	5	C_L = 20pf

Table 9- AC Characteristics – Gigabit Media Independent Interface

12.8.3 PCS Interface

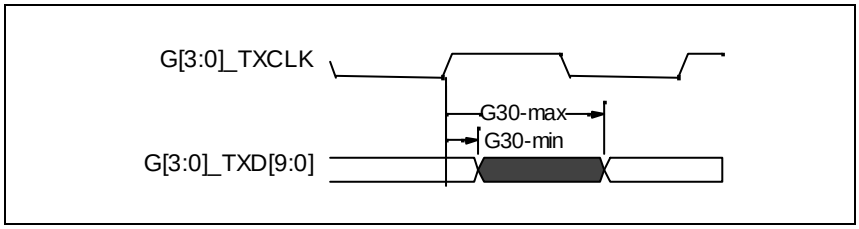


Figure 17 - AC Characteristics – PCS Interface

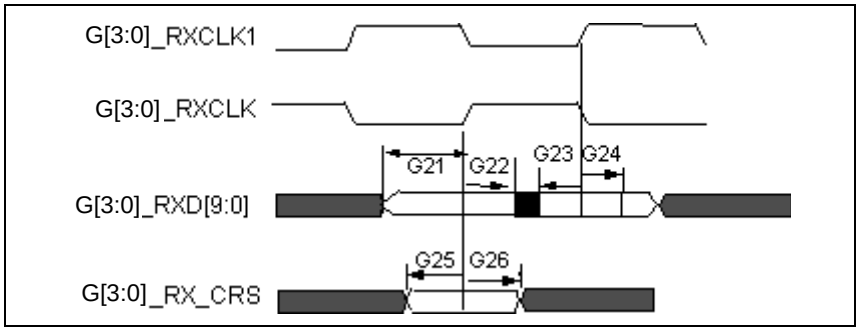


Figure 18 - AC Characteristics – PCS Interface

Symbol	Parameter	(G_RCLK & G_REFCLK = 125MHz)		Note:
		Min (ns)	Max (ns)	
G21	**G[3:0]**_RXD[9:0] Input Setup Times ref to G_RXCLK	2		
G22	**G[3:0]**_RXD[9:0] Input Hold Times ref to G_RXCLK	1		
G23	**G[3:0]**_RXD[9:0] Input Setup Times ref to G_RXCLK1	2		
G24	**G[3:0]**_RXD[9:0] Input Hold Times ref to G_RXCLK1	1		
G25	**G[3:0]**_CRS Input Setup Times	2		
G26	**G[3:0]**_CRS Input Hold Times	1		
G30	**G[3:0]**_TXD[9:0] Output Delay Times	1	5	C_L = 20pf

Table 10- AC Characteristics – PCS Interface

12.8.4 LED Interface

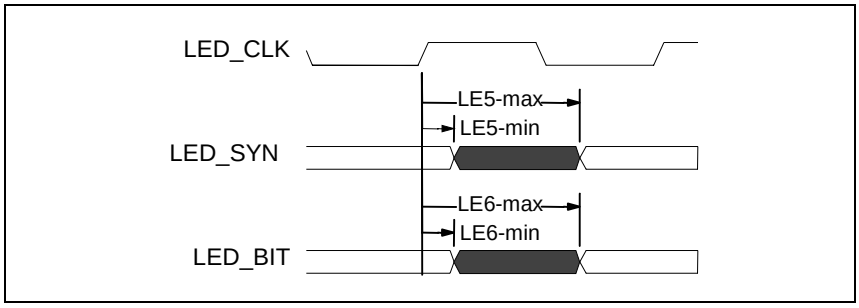


Figure 19 - AC Characteristics – LED Interface

Symbol	Parameter	Variable FREQ.		Note:
		Min (ns)	Max (ns)	
LE5	LED_SYN Output Valid Delay	1	7	C_L = 30pf
LE6	LED_BIT Output Valid Delay	1	7	C_L = 30pf

Table 11- AC Characteristics – LED Interface

12.8.5 MDIO Input Setup and Hold Timing

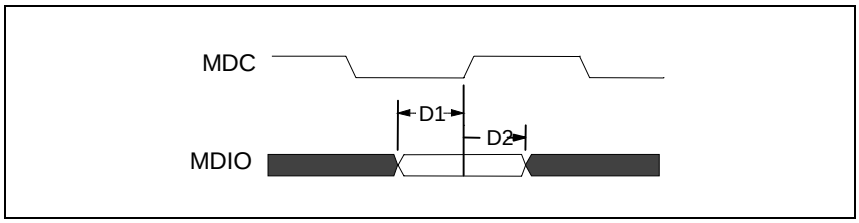


Figure 20 - MDIO Input Setup and Hold Timing

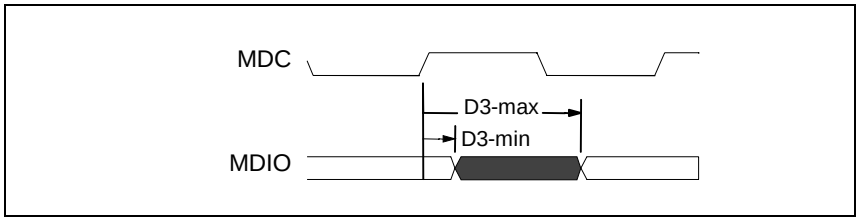


Figure 21 - MDIO Output Delay Timing

Symbol	Parameter	1MHz Min (ns)	1MHz Max (ns)	Note:
D1	MDIO input setup time	10		
D2	MDIO input hold time	2		
D3	MDIO output delay time	1	20	C_L = 50pf

Table 12- MDIO Timing

12.8.6 I^2C Input Setup Timing

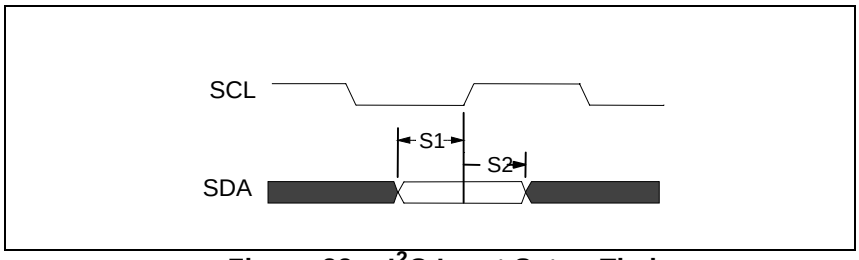


Figure 22 - I^2C Input Setup Timing

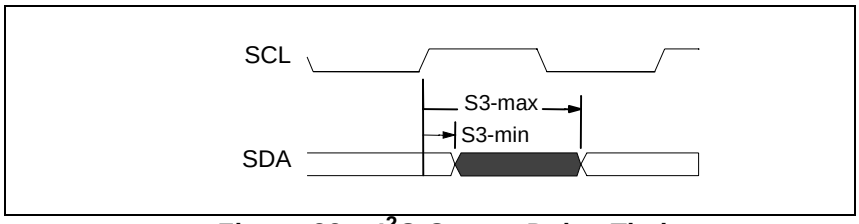


Figure 23 - I^2C Output Delay Timing

Symbol	Parameter	500KHz Min (ns)	Max (ns)	Note:
S1	SDA input setup time	20		
S2	SDA input hold time	1		
S3*	SDA output delay time	1	20	C_L = 30pf
* Open Drain Output. Low to High transistor is controlled by external pullup resistor.				

Table 13- I^2C Timing

12.8.7 Serial Interface Setup Timing

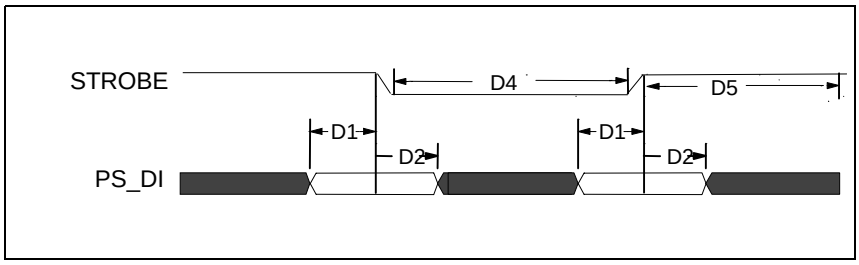


Figure 24 - Serial Interface Setup Timing

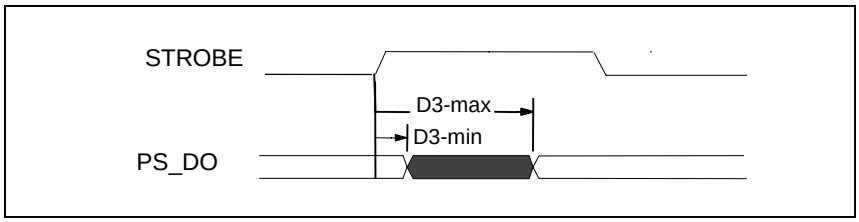
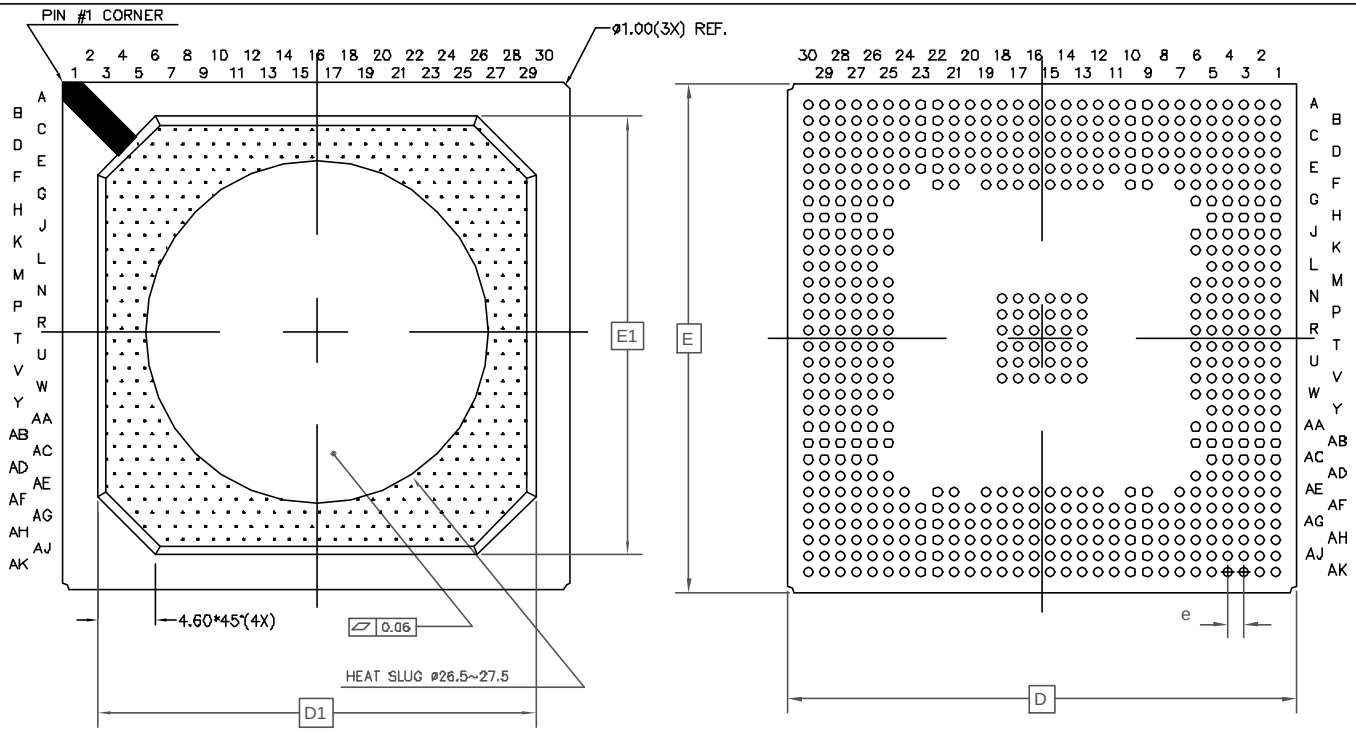
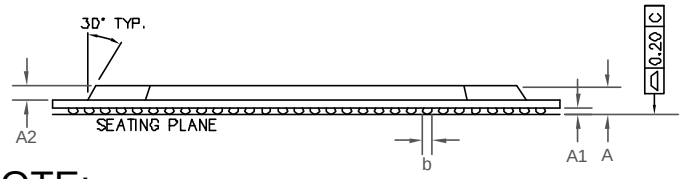


Figure 25 - Serial Interface Output Delay Timing

Symbol	Parameter	(SCLK =133 MHz) Min (ns)	Max (ns)	Note:
D1	PS_DI setup time	20		
D2	PS_DI hold time	10		
D3	PS_DO output delay time	1	50	C_L = 100pf
D4	Strobe low time	5µs		
D5	Strobe high time	5µs		

Table 14- Serial Interface Timing





DIMENSION	MIN	MAX
A	2.20	2.46
A1	0.50	0.70
A2	1.17 REF	
D	39.80	40.20
D1	34.50 REF	
E	39.80	40.20
E1	34.50 REF	
b	0.60	0.90
e	1.27	
N	596	
Conforms to JEDEC MS - 034		

NOTE:

1. CONTROLLING DIMENSIONS ARE IN MM
2. DIMENSION "b" IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER
3. SEATING PLANE IS DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
4. N IS THE NUMBER OF SOLDER BALLS
5. NOT TO SCALE.
6. SUBSTRATE THICKNESS IS 0.56 MM

ISSUE	1			
ACN	213931			
DATE	20Jan03			
APPRD.				

ZARLINK SEMICONDUCTOR

Previous package codes: BH / G

Package Code GK

Package Outline for 596 Ball HSBGA (40x40x2.33mm)

GPD00817

For more information about all Zarlink products visit our Web Site at

www.zarlink.com