

**Features**

October 2003

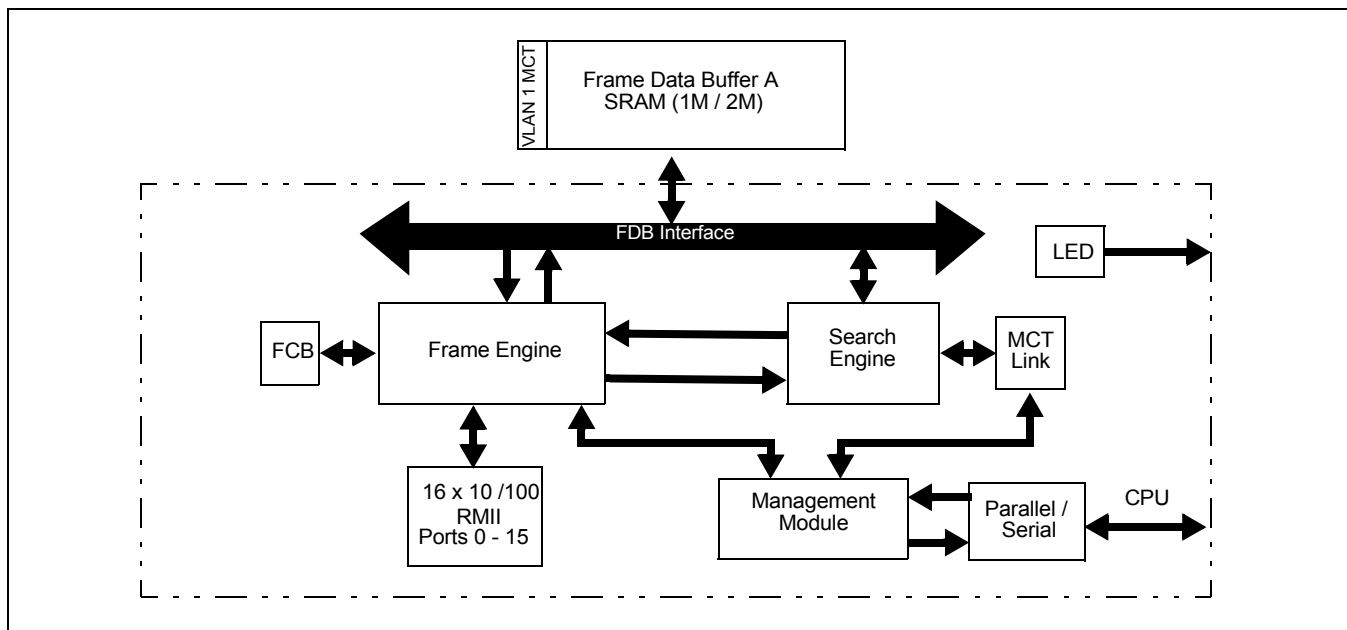
- Integrated Single-Chip 10/100 Mbps Ethernet Switch
- 16 10/100 Mbps Autosensing, Fast Ethernet Ports with RMII or Serial Interface (7WS). Each port can independently use one of the two interfaces.
- Supports 8/16-bit CPU interface in managed mode
- Serial interface in unmanaged mode
- Supports one Frame Buffer Memory domain with SRAM at 100 MHz
- Supports SRAM domain memory size 1 MB, or 2 MB
- Applies centralized shared memory architecture
- Up to 64K MAC addresses
- Maximum throughput is 1.6 Gbps non-blocking
- High performance packet forwarding (4.762M packets per second) at full wire speed
- Provides port based and ID tagged VLAN support (IEEE 802.1Q), up to 255 VLANs
- Supports IP Multicast with IGMP snooping
- Supports spanning tree with CPU, on per port or per VLAN basis
- Packet Filtering and Port Security
  - Static address filtering for source and/or destination MAC

**Ordering Information**

ZL50416/GKC 553 Pin HSBGA

**-40°C to +85°C**

- Static MAC address not subject to aging
- Secure mode freezes MAC address learning. Each port may independently use this mode.
- Full Duplex Ethernet IEEE 802.3x Flow Control
- Backpressure flow control for Half Duplex ports
- Supports Ethernet multicasting and broadcasting and flooding control
- Supports per-system option to enable flow control for best effort frames even on QoS-enabled ports
- Traffic Classification
  - 4 transmission priorities for Fast Ethernet ports with 2 dropping levels
  - Classification based on:
    - Port based priority
    - VLAN Priority field in VLAN tagged frame
    - DS/TOS field in IP packet
    - UDP/TCP logical ports: 8 hard-wired and 8 programmable ports, including one programmable range
- The precedence of the above classifications is programmable.


**Figure 1 - ZL50416 System Block Diagram**

- QoS Support
  - Supports IEEE 802.1p/Q Quality of Service with 4 transmission priority queues with delay bounded, strict priority, and WFQ service disciplines
  - Provides 2 levels of dropping precedence with WRED mechanism
  - User controls the WRED thresholds.
  - Buffer management: per class and per port buffer reservations
  - Port-based priority: VLAN priority in a tagged frame can be overwritten by the priority of Port VLAN ID.
- 2 port trunking groups with up to 4 10/100 ports per group
- Load sharing among trunked ports can be based on source MAC and/or destination MAC.
- Port Mirroring to any two ports of 0-15 in managed mode or to a dedicated mirroring port in unmanaged mode
- Full set of LED signals provided by a serial interface
- Built-in MIB statistics counters
- Recognizes Simple Bandwidth Management (SBM) and Resource Reservation Protocol (RSVP) packets and forwards to CPU
- Hardware auto-negotiation through serial management interface (MDIO) for Ethernet ports
- Built-in reset logic triggered by system malfunction
- Built-In Self Test for internal and external SRAM
- I<sup>2</sup>C EEPROM for configuration
- 553 BGA package

## Description

The ZL50416 is a high density, low cost, high performance, non-blocking Ethernet switch chip. A single chip provides 16 ports at 10/100 Mbps, and a CPU interface for managed and unmanaged switch applications.

The chip supports up to 64K MAC addresses and up to 255 port-based Virtual LANs (VLANs). The centralized shared memory architecture permits a very high performance packet forwarding rate at up to 3.571M packets per second at full wire speed. The chip is optimized to provide low-cost, high-performance workgroup switching.

The Frame Buffer Memory domains utilize cost-effective, high-performance synchronous SRAM with aggregate bandwidth of 6.4 Gbps to support full wire speed on all ports simultaneously.

With delay bounded, strict priority, and/or WFQ transmission scheduling, and WRED dropping schemes, the ZL50416 provides powerful QoS functions for various multimedia and mission-critical applications. The chip provides 4 transmission priorities and 2 levels of dropping precedence. Each packet is assigned a transmission priority and dropping precedence based on the VLAN priority field in a VLAN tagged frame, or the DS/TOS field, and UDP/TCP logical port fields in IP packets. The ZL50416 recognizes a total of 16 UDP/TCP logical ports, 8 hard-wired and 8 programmable (including one programmable range).

The ZL50416 supports 2 groups of port trunking/load sharing. Each 10/100 group can contain up to 4 ports. Port trunking/load sharing can be used to group ports between interlinked switches to increase the effective network bandwidth.

In half-duplex mode, all ports support backpressure flow control, to minimize the risk of losing data during long activity bursts. In full-duplex mode, IEEE 802.3x flow control is provided. The ZL50416 also supports a per-system option to enable flow control for best effort frames, even on QoS-enabled ports.

Statistical information for SNMP and the Remote Monitoring Management Information Base (RMON MIB) are collected independently for all ports. Access to these statistical counters/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 ZL50416 is fabricated using 0.25 micron technology. Inputs, however, are 3.3 V tolerant, and the outputs are capable of directly interfacing to LVTTTL levels. The ZL50416 is packaged in a 553-pin Ball Grid Array package.

## Table of Contents

<b>1.0 Block Functionality</b>	<b>12</b>
1.1 Frame Data Buffer (FDB) Interfaces	12
1.2 10/100 MAC Module (RMAC)	12
1.3 CPU Interface Module	12
1.4 Management Module	12
1.5 Frame Engine	12
1.6 Search Engine	12
1.7 LED Interface	12
1.8 Internal Memory	12
<b>2.0 System Configuration</b>	<b>13</b>
2.1 Management and Configuration	13
2.2 Managed Mode	13
2.3 Register Configuration, Frame Transmission, and Frame Reception	14
2.3.1 Register Configuration	14
2.3.2 Rx/Tx of Standard Ethernet Frames	14
2.3.3 Control Frames	14
2.4 Unmanaged Mode	15
2.5 I <sup>2</sup> C Interface	15
2.5.1 Start Condition	15
2.5.2 Address	15
2.5.3 Data Direction	16
2.5.4 Acknowledgment	16
2.5.5 Data	16
2.5.6 Stop Condition	16
2.6 Synchronous Serial Interface	16
2.6.1 Write Command	17
2.6.2 Read Command	17
<b>3.0 ZL50416 Data Forwarding Protocol</b>	<b>17</b>
3.1 Unicast Data Frame Forwarding	17
3.2 Multicast Data Frame Forwarding	18
3.3 Frame Forwarding To and From CPU	18
<b>4.0 Memory Interface</b>	<b>19</b>
4.1 Overview	19
4.2 Detailed Memory Information	19
4.3 Memory Requirements	19
<b>5.0 Search Engine</b>	<b>20</b>
5.1 Search Engine Overview	20
5.2 Basic Flow	20
5.3 Search, Learning, and Aging	21
5.3.1 MAC Search	21
5.3.2 Learning	21
5.3.3 Aging	22
5.3.4 VLAN Table	22
5.4 MAC Address Filtering	23
5.5 Quality of Service	23
5.6 Priority Classification Rule	24
5.7 Port and Tag Based VLAN	24
5.7.1 Port-Based VLAN	24
5.7.2 Tag-Based VLAN	25
5.8 Memory Configurations	25
<b>6.0 Frame Engine</b>	<b>28</b>
6.1 Data Forwarding Summary	28

## Table of Contents

6.2 Frame Engine Details	28
6.2.1 FCB Manager	28
6.2.2 Rx Interface	28
6.2.3 RxDMA	28
6.2.4 TxQ Manager	28
6.3 Port Control	29
6.4 TxDMA	29
<b>7.0 Quality of Service and Flow Control</b>	<b>29</b>
7.1 Model	29
7.2 Four QoS Configurations	30
7.3 Delay Bound	31
7.4 Strict Priority and Best Effort	31
7.5 Weighted Fair Queuing	31
7.6 Rate Control	32
7.7 WRED Drop Threshold Management Support	32
7.8 Buffer Management	33
7.8.1 Dropping When Buffers Are Scarce	34
7.9 ZL50416 Flow Control Basics	34
7.9.1 Unicast Flow Control	35
7.9.2 Multicast Flow Control	35
7.10 Mapping to IETF Diffserv Classes	35
<b>8.0 Port Trunking</b>	<b>36</b>
8.1 Features and Restrictions	36
8.2 Unicast Packet Forwarding	36
8.3 Multicast Packet Forwarding	37
8.4 Unmanaged Trunking	37
<b>9.0 Port Mirroring</b>	<b>38</b>
9.1 Port Mirroring Features	38
9.2 Setting Registers for Port Mirroring	38
<b>10.0 GPSI (7WS) Interface</b>	<b>39</b>
10.1 GPSI connection	39
10.2 SCAN LINK and SCAN COL interface	40
<b>11.0 LED Interface</b>	<b>40</b>
11.1 LED Interface Introduction	40
11.2 Port Status	40
11.3 LED Interface Timing Diagram	41
<b>12.0 Hardware Statistics Counter</b>	<b>42</b>
12.1 Hardware Statistics Counters List	42
12.2 IEEE 802.3 HUB Management (RFC 1516)	43
12.2.1 Event Counters	43
12.2.1.1 ReadableOctet	43
12.2.1.2 ReadableFrame	43
12.2.2 FCSErrors	44
12.2.2.1 AlignmentErrors	44
12.2.2.2 FrameTooLongs	44
12.2.2.3 ShortEvents	44
12.2.2.4 Runts	44
12.2.2.5 Collisions	45
12.2.2.6 LateEvents	45
12.2.2.7 VeryLongEvents	45
12.2.2.8 DataRateMisatches	45

12.2.2.9 AutoPartitions	45
12.2.2.10 TotalErrors	45
12.3 IEEE – 802.1 Bridge Management (RFC 1286)	45
12.3.1 Event Counters	45
12.3.1.1 InFrames	45
12.3.1.2 OutFrames	46
12.3.1.3 InDiscards	46
12.3.1.4 DelayExceededDiscards	46
12.3.1.5 MtuExceededDiscards	46
12.4 RMON – Ethernet Statistic Group (RFC 1757)	46
12.4.1 Event Counters	46
12.4.1.1 Drop Events	46
12.4.1.2 Octets	46
12.4.1.3 BroadcastPkts	46
12.4.1.4 MulticastPkts	46
12.4.1.5 CRCAlignErrors	46
12.4.1.6 UndersizePkts	47
12.4.1.7 OversizePkts	47
12.4.1.8 Fragments	47
12.4.1.9 Jabbers	47
12.4.1.10 Collisions	47
<b>13.0 Register Definition</b>	<b>48</b>
13.1 ZL50416 Register Description	48
13.2 Directly Accessed Registers	52
13.2.1 INDEX_REG0	52
13.2.2 INDEX_REG1 (only needed for 8-bit mode)	52
13.2.3 DATA_FRAME_REG	52
13.2.4 CONTROL_FRAME_REG	53
13.2.5 COMMAND&STATUS Register	53
13.2.6 Interrupt Register	54
13.2.7 Control Command Frame Buffer1 Access Register	54
13.2.8 Control Command Frame Buffer2 Access Register	54
13.3 (Group 0 Address) MAC Ports Group	54
13.3.1 ECR1Pn: Port N Control Register	54
13.3.2 ECR2Pn: Port N Control Register	55
13.4 (Group 1 Address) VLAN Group	57
13.4.1 AVTCL – VLAN Type Code Register Low	57
13.4.2 AVTCH – VLAN Type Code Register High	57
13.4.3 PVMAP00_0 – Port 00 Configuration Register 0	57
13.4.4 PVMAP00_1 – Port 00 Configuration Register 1	57
13.4.5 PVMAP00_3 – Port 00 Configuration Register 3	58
13.5 Port Configuration Registers	60
13.5.1 PVMODE	60
13.5.2 PVROUTE0	61
13.5.3 PVROUTE1	61
13.5.4 PVROUTE2	62
13.5.5 PVROUTE3	62
13.5.6 PVROUTE4	62
13.5.7 PVROUTE5	63
13.5.8 PVROUTE6	63
13.5.9 PVROUTE7	64
13.6 (Group 2 Address) Port Trunking Groups	64
13.6.1 TRUNK0_L – Trunk group 0 Low (Managed mode only)	64
13.6.2 TRUNK0_M – Trunk group 0 Medium (Managed mode only)	64

13.6.3 TRUNK0_MODE– Trunk group 0 mode. . . . .	65
13.6.4 TRUNK0_HASH0 – Trunk group 0 hash result 0 destination port number . . . . .	65
13.6.5 TRUNK0_HASH1 – Trunk group 0 hash result 1 destination port number . . . . .	65
13.6.6 TRUNK0_HASH2 – Trunk group 0 hash result 2 destination port number . . . . .	65
13.6.7 TRUNK0_HASH3 – Trunk group 0 hash result 3 destination port number . . . . .	65
13.6.8 TRUNK1_L – Trunk group 1 Low (Managed mode only). . . . .	66
13.6.9 TRUNK1_M – Trunk group 1 Medium (Managed mode only) . . . . .	66
13.6.10 TRUNK1_MODE – Trunk group 1 mode . . . . .	66
13.6.11 TRUNK1_HASH0 – Trunk group 1 hash result 0 destination port number . . . . .	66
13.6.12 TRUNK1_HASH1 – Trunk group 1 hash result 1 destination port number . . . . .	66
13.6.13 TRUNK1_HASH2 – Trunk group 1 hash result 2 destination port number . . . . .	66
13.6.14 TRUNK1_HASH3 – Trunk group 1 hash result 3 destination port number . . . . .	67
13.6.15 Multicast Hash Registers . . . . .	67
13.6.15.1 Multicast_HASH0-0 – Multicast hash result 0 mask byte 0 . . . . .	67
13.6.15.2 Multicast_HASH0-1 – Multicast hash result 0 mask byte 1 . . . . .	67
13.6.15.3 Multicast_HASH0-3 – Multicast hash result 0 mask byte 3 . . . . .	67
13.6.15.4 Multicast_HASH1-0 – Multicast hash result 1 mask byte 0 . . . . .	68
13.6.15.5 Multicast_HASH1-1 – Multicast hash result 1 mask byte 1 . . . . .	68
13.6.15.6 Multicast_HASH1-3 – Multicast hash result 1 mask byte 3 . . . . .	68
13.6.15.7 Multicast_HASH2-0 – Multicast hash result 2 mask byte 0 . . . . .	68
13.6.15.8 Multicast_HASH2-1 – Multicast hash result 2 mask byte 1 . . . . .	68
13.6.15.9 Multicast_HASH2-3 – Multicast hash result 2 mask byte 3 . . . . .	68
13.6.15.10 Multicast_HASH3-0 – Multicast hash result 3 mask byte 0 . . . . .	68
13.6.15.11 Multicast_HASH3-1 – Multicast hash result 3 mask byte 1 . . . . .	69
13.6.15.12 Multicast_HASH3-3 – Multicast hash result 3 mask byte 3 . . . . .	69
13.7 (Group 3 Address) CPU Port Configuration Group . . . . .	69
13.7.1 MAC0 – CPU Mac address byte 0 . . . . .	69
13.7.2 MAC1 – CPU Mac address byte 1 . . . . .	69
13.7.3 MAC2 – CPU Mac address byte 2 . . . . .	69
13.7.4 MAC3 – CPU Mac address byte 3 . . . . .	69
13.7.5 MAC4 – CPU Mac address byte 4 . . . . .	70
13.7.6 MAC5 – CPU Mac address byte 5 . . . . .	70
13.7.7 INT_MASK0 – Interrupt Mask 0 . . . . .	70
13.7.8 INTP_MASK0 – Interrupt Mask for MAC Port 0,1 . . . . .	70
13.7.9 INTP_MASK1 – Interrupt Mask for MAC Port 2,3 . . . . .	71
13.7.10 INTP_MASK2 – Interrupt Mask for MAC Port 4,5 . . . . .	71
13.7.11 INTP_MASK3 – Interrupt Mask for MAC Port 6,7 . . . . .	71
13.7.12 INTP_MASK4 – Interrupt Mask for MAC Port 8,9 . . . . .	71
13.7.13 INTP_MASK5 – Interrupt Mask for MAC Port 10,11 . . . . .	71
13.7.14 INTP_MASK6 – Interrupt Mask for MAC Port 12,13 . . . . .	71
13.7.15 INTP_MASK7 – Interrupt Mask for MAC Port 14,15 . . . . .	71
13.7.16 RQS – Receive Queue Select CPU Address:h323). . . . .	71
13.7.17 RQSS – Receive Queue Status . . . . .	72
13.7.18 TX_AGE – Tx Queue Aging timer . . . . .	72
13.8 (Group 4 Address) Search Engine Group . . . . .	72
13.8.1 AGETIME_LOW – MAC address aging time Low . . . . .	72
13.8.2 AGETIME_HIGH –MAC address aging time High . . . . .	73
13.8.3 V_AGETIME – VLAN to Port aging time . . . . .	73
13.8.4 SE_OPMODE – Search Engine Operation Mode . . . . .	73
13.8.5 SCAN – SCAN Control Register (default 00). . . . .	74
13.9 (Group 5 Address) Buffer Control/QOS Group . . . . .	75
13.9.1 FCBAT – FCB Aging Timer . . . . .	75
13.9.2 QOSC – QOS Control . . . . .	75
13.9.3 FCR – Flooding Control Register. . . . .	75

13.9.4 AVPML – VLAN Tag Priority Map	76
13.9.5 AVPMM – VLAN Priority Map	76
13.9.6 AVPMH – VLAN Priority Map	77
13.9.7 TOSPML – TOS Priority Map	77
13.9.8 TOSPMM – TOS Priority Map	77
13.9.9 TOSPMH – TOS Priority Map	78
13.9.10 AVDM – VLAN Discard Map	78
13.9.11 TOSDML – TOS Discard Map	79
13.9.12 BMRC - Broadcast/Multicast Rate Control	79
13.9.13 UCC – Unicast Congestion Control	79
13.9.14 MCC – Multicast Congestion Control	80
13.9.15 PR100 – Port Reservation for 10/100 ports	80
13.9.16 SFCB – Share FCB Size	80
13.9.17 C2RS – Class 2 Reserve Size	81
13.9.18 C3RS – Class 3 Reserve Size	81
13.9.19 C4RS – Class 4 Reserve Size	81
13.9.20 C5RS – Class 5 Reserve Size	81
13.9.21 C6RS – Class 6 Reserve Size	82
13.9.22 C7RS – Class 7 Reserve Size	82
13.9.23 QOSCn - Classes Byte Limit Set 0	82
13.9.24 Classes Byte Limit Set 1	82
13.9.25 Classes Byte Limit Set 2	83
13.9.26 Classes Byte Limit Set 3	83
13.9.27 Classes WFQ Credit Set 0	83
13.9.28 Classes WFQ Credit Set 1	83
13.9.29 Classes WFQ Credit Set 2	84
13.9.30 Classes WFQ Credit Set 3	84
13.9.31 RDRC0 – WRED Rate Control 0	84
13.9.32 RDRC1 – WRED Rate Control 1	85
13.9.33 User Defined Logical Ports and Well Known Ports	85
13.9.33.1 USER_PORT0_(0~7) – User Define Logical Port (0~7)	86
13.9.33.2 USER_PORT_[1:0]_PRIORITY - User Define Logic Port 1 and 0 Priority	86
13.9.33.3 USER_PORT_[3:2]_PRIORITY - User Define Logic Port 3 and 2 Priority	86
13.9.33.4 USER_PORT_[5:4]_PRIORITY - User Define Logic Port 5 and 4 Priority	87
13.9.33.5 USER_PORT_[7:6]_PRIORITY - User Define Logic Port 7 and 6 Priority	87
13.9.33.6 USER_PORT_ENABLE[7:0] – User Define Logic 7 to 0 Port Enables	87
13.9.33.7 WELL_KNOWN_PORT[1:0] PRIORITY- Well Known Logic Port 1 and 0 Priority	87
13.9.33.8 WELL_KNOWN_PORT[3:2] PRIORITY- Well Known Logic Port 3 and 2 Priority	88
13.9.33.9 WELL_KNOWN_PORT [5:4] PRIORITY- Well Known Logic Port 5 and 4 Priority	88
13.9.33.10 WELL_KNOWN_PORT [7:6] PRIORITY- Well Known Logic Port 7 and 6 Priority	88
13.9.33.11 WELL_KNOWN_PORT_ENABLE [7:0] – Well Known Logic 7 to 0 Port Enables	88
13.9.33.12 RLOWL – User Define Range Low Bit 7:0	89
13.9.33.13 RLOWH – User Define Range Low Bit 15:8	89
13.9.33.14 RHIGHL – User Define Range High Bit 7:0	89
13.9.33.15 RHIGHH – User Define Range High Bit 15:8	89
13.9.33.16 RPRIORITY – User Define Range Priority	89
13.9.34 CPUQOSC123	90
13.10 (Group 6 Address) MISC Group	90
13.10.1 MII_OP0 – MII Register Option 0	90
13.10.2 MII_OP1 – MII Register Option 1	90
13.10.3 FEN – Feature Register	91
13.10.4 MIIC0 – MII Command Register 0	92
13.10.5 MIIC1 – MII Command Register 1	92
13.10.6 MIIC2 – MII Command Register 2	92



13.10.7 MIIC3 – MII Command Register 3	92
13.10.8 MIID0 – MII Data Register 0	93
13.10.9 MIID1 – MII Data Register 1	93
13.10.10 0LED Mode – LED Control	93
13.10.11 CHECKSUM - EEPROM Checksum	94
13.11 (Group 7 Address) Port Mirroring Group	94
13.11.1 MIRROR1_SRC – Port Mirror source port	94
13.11.2 MIRROR1_DEST – Port Mirror destination	94
13.11.3 MIRROR2_SRC – Port Mirror source port	95
13.11.4 MIRROR2_DEST – Port Mirror destination	95
13.12 (Group F Address) CPU Access Group	95
13.12.1 GCR-Global Control Register	95
13.12.2 DCR-Device Status and Signature Register	96
13.12.3 DCR1-Chip Status	96
13.12.4 DPST – Device Port Status Register	97
13.12.5 DTST – Data Read Back Register	97
13.12.6 PLLCR - PLL Control Register	97
13.12.7 LCLK - LA_CLK delay from internal OE_CLK	98
13.12.8 OECLK - Internal OE_CLK delay from SCLK	98
13.12.9 DA – DA Register	98
<b>14.0 BGA and Ball Signal Descriptions</b>	<b>99</b>
14.1 BGA Views (TOP-View)	99
14.1.1 Encapsulated View In Unmanaged Mode	99
14.1.2 Encapsulated View In Managed Mode	100
14.2 Ball – Signal Descriptions in Managed Mode	101
14.2.1 Ball Signal Descriptions in Managed Mode	101
14.2.2 Ball – Signal Descriptions in Unmanaged Mode	108
14.3 Ball – Signal Name in Unmanaged Mode	115
14.4 Ball – Signal Name in Managed Mode	121
14.5 AC/DC Timing	127
14.5.2 DC Electrical Characteristics	127
14.5.3 Recommended Operating Conditions	128
14.5.4 Typical Reset & Bootstrap Timing Diagram	129
14.5.5 Typical CPU Timing Diagram for a CPU Write Cycle	130
14.5.6 Typical CPU Timing Diagram for a CPU Read Cycle	131
14.6 Local Frame Buffer SDRAM Memory Interface	132
14.6.1 Local SDRAM Memory Interface:	132
14.7 AC Characteristics	133
14.7.1 Reduced Media Independent Interface	133
14.7.2 LED Interface	134
14.7.3 SCANLINK SCANCOL Output Delay Timing	135
14.7.4 MDIO Input Setup and Hold Timing	136
14.7.5 I <sup>2</sup> C Input Setup Timing	137
14.7.6 Serial Interface Setup Timing	138

## List of Tables

Table 1 - Memory Configuration . . . . .	19
Table 2 - VLAN Index Mapping Table . . . . .	22
Table 3 - VLAN Index Port Association Table . . . . .	22
Table 4 - Port-Based VLAN . . . . .	24
Table 5 - Supported Memory Configurations (Pipeline SDRAM Mode) . . . . .	25
Table 6 - Options for Memory Configuration . . . . .	26
Table 7 - Two-dimensional World Traffic . . . . .	29
Table 8 - Four QoS Configurations for a 10/100 Mbps Port . . . . .	30
Table 9 - Weighted Random Early Detection (WRED) logic . . . . .	32
Table 10 - Mapping between ZL50416 and IETF Diffserv Classes for 10/100 Ports . . . . .	35
Table 11 - ZL50416 Features Enabling IETF Diffserv Standards . . . . .	36
Table 12 - Select via trunk0_mode register . . . . .	37
Table 13 - Select via trunk1_mode register . . . . .	37
Table 14 - Register Description . . . . .	48
Table 15 - Reset & Bootstrap Timing . . . . .	129
Table 16 - Write Cycle . . . . .	130
Table 17 - Read Cycle . . . . .	131
Table 18 - AC Characteristics – Local frame buffer SDRAM Memory Interface . . . . .	133
Table 19 - AC Characteristics – Reduced Media Independent Interface . . . . .	134
Table 20 - AC Characteristics – LED Interface . . . . .	134
Table 21 - TSCANLINK, SCANCOL Timing . . . . .	135
Table 22 - MDIO Timing . . . . .	136
Table 23 - I <sup>2</sup> C Timing . . . . .	137
Table 24 - Serial Interface Timing . . . . .	138

## List of Figures

Figure 1 - ZL50416 System Block Diagram . . . . .	1
Figure 2 - Overview of the ZL50416 CPU Interface . . . . .	13
Figure 3 - Data Transfer Format for I <sup>2</sup> C Interface . . . . .	15
Figure 4 - Write Command . . . . .	17
Figure 5 - Read Command . . . . .	17
Figure 6 - ZL50416 SRAM Interface Block Diagram (DMAs for 10/100 Ports Only) . . . . .	19
Figure 7 - Memory Map . . . . .	20
Figure 8 - Priority Classification Rule . . . . .	24
Figure 9 - Memory Configuration For: 1 bank, 1 Layer, 1MB total . . . . .	26
Figure 10 - Memory Configuration For: 1 Bank, 2 Layers, 2MB total . . . . .	27
Figure 11 - Memory Configuration For: 1 bank, 1 Layer, 2MB . . . . .	27
Figure 12 - Buffer Partition Scheme Used to Implement Buffer Management . . . . .	34
Figure 13 - GPSI (7WS) mode connection diagram . . . . .	39
Figure 14 - SCAN LINK and SCAN COLLISION status diagram . . . . .	40
Figure 15 - Timing Diagram of LED Interface . . . . .	41
Figure 16 - Typical Reset & Bootstrap Timing Diagram . . . . .	129
Figure 17 - Typical CPU Timing Diagram for a CPU Write Cycle . . . . .	130
Figure 18 - Typical CPU Timing Diagram for a CPU Read Cycle . . . . .	131
Figure 19 - Local Memory Interface – Input Setup and Hold Timing . . . . .	132
Figure 20 - Local Memory Interface - Output Valid Delay Timing . . . . .	132
Figure 21 - AC Characteristics – Reduced media independent Interface . . . . .	133
Figure 22 - AC Characteristics – Reduced Media Independent Interface . . . . .	133
Figure 23 - AC Characteristics – LED Interface . . . . .	134
Figure 24 - SCANLINK SCANCOL Output Delay Timing . . . . .	135
Figure 25 - SCANLINK, SCANCOL Setup Timing . . . . .	135
Figure 26 - MDIO Input Setup and Hold Timing . . . . .	136
Figure 27 - MDIO Output Delay Timing . . . . .	136
Figure 28 - I <sup>2</sup> C Input Setup Timing . . . . .	137
Figure 29 - I <sup>2</sup> C Output Delay Timing . . . . .	137
Figure 30 - Serial Interface Setup Timing . . . . .	138
Figure 31 - Serial Interface Output Delay Timing . . . . .	138

## 1.0 Block Functionality

### 1.1 Frame Data Buffer (FDB) Interfaces

The FDB interface supports SBRAM memory at 100 MHz. To ensure a non-blocking switch, one memory domain with a 64 bit wide memory bus is required. At 100 MHz, the aggregate memory bandwidth is 6.4 Gbps, which is enough to support 16 10/100 Mbps.

The Switching Database is also located in the external SRAM; it is used for storing MAC addresses and their physical port number.

### 1.2 10/100 MAC Module (RMAC)

The 10/100 Media Access Control module provides the necessary buffers and control interface between the Frame Engine (FE) and the external physical device (PHY). The ZL50416 has two interfaces, RMI or Serial (only for 10M). The 10/100 MAC of the ZL50416 device meets the IEEE 802.3 specification. It is able to operate in either Half or Full Duplex mode with a back pressure/flow control mechanism. In addition, it will automatically retransmit upon collision for up to 16 total transmissions. The PHY addresses for 16 10/100 MAC are from 08h to 17h.

### 1.3 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 (Bootstrap TSTOUT6 makes the selection). It also supports a serial and an I<sup>2</sup>C interface, which provides an easy way to configure the system if unmanaged.

### 1.4 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.

### 1.5 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, 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.

### 1.6 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.

### 1.7 LED Interface

The LED interface provides a serial interface for carrying 16 port status signals.

### 1.8 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.
- MAC address Control Table (MCT) Link Table - The MCT Link Table stores the linked list of MCT entries that have collisions in the external MAC Table.

**Note** that the external MAC table is located in the external SSRAM Memory.

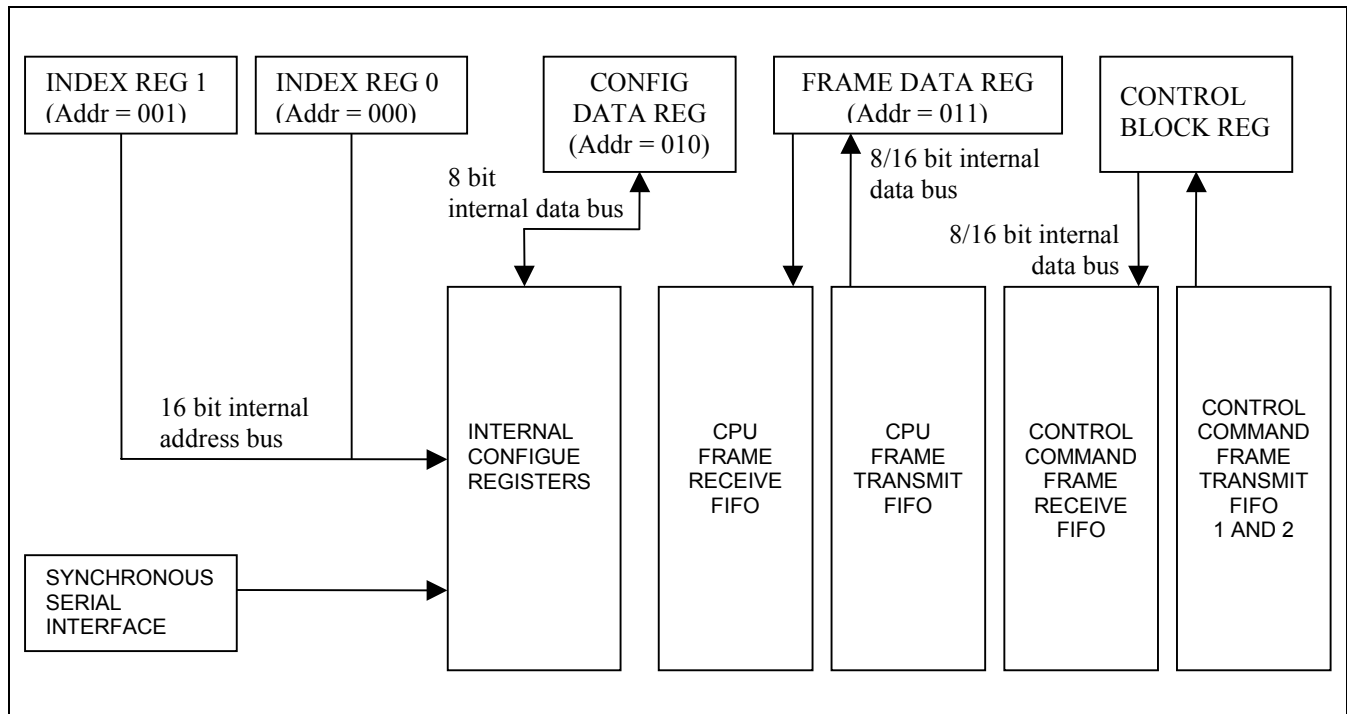
## 2.0 System Configuration

### 2.1 Management and Configuration

Two modes are supported in the ZL50416: managed and unmanaged. In managed mode, the ZL50416 uses an 8 or 16 bit CPU interface very similar to the Industry Standard Architecture (ISA) specification. In unmanaged mode, the ZL50416 has no CPU but can be configured by EEPROM using an I<sup>2</sup>C interface at bootup, or via a synchronous serial interface otherwise.

### 2.2 Managed Mode

In managed mode, the ZL50416 uses an 8 or 16 bit CPU interface very similar to the ISA bus. The ZL50416 CPU interface provides for easy and effective management of the switching system. Figure 2 provides an overview of the CPU interface.



**Figure 2 - Overview of the ZL50416 CPU Interface**

## 2.3 Register Configuration, Frame Transmission, and Frame Reception

### 2.3.1 Register Configuration

The ZL50416 has many programmable parameters, covering such functions as QoS weights, VLAN control, and port mirroring setup. In managed mode, the CPU interface provides an easy way of configuring these parameters. The parameters are contained in 8-bit configuration registers. The ZL50416 allows indirect access to these registers, as follows:

- If operating in 8 bits-interface mode, two “index” registers (addresses 000 and 001) need to be written, to indicate the desired 8-bit register address. In 16-bit mode, only one register (address 000) needs to be written for 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.

### 2.3.2 Rx/Tx of Standard 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 (minimum 64 bytes). After writing all the data, it then writes the frame size, destination port number, and frame status.
- The ZL50416 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.

### 2.3.3 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 ZL50416 and sent to the CPU. These proprietary frames are related to such tasks as statistics collection, MAC address learning, and aging, etc... All Control frames are up to 40 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 111).

Specifically, there are eight types of control frames generated by the CPU and sent to the ZL50416:

- 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 ZL50416 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

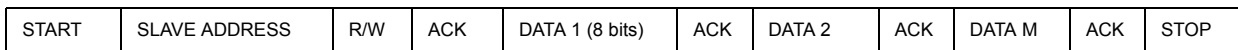
The format of the Control Frame is described in the processor interface application note.

## 2.4 Unmanaged Mode

In unmanaged mode, the ZL50416 can be configured by EEPROM (24C02 or compatible) via an I<sup>2</sup>C interface at boot time, or via a synchronous serial interface during operation.

## 2.5 I<sup>2</sup>C Interface

The I<sup>2</sup>C 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 bidirectional, 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. Figure 3 depicts the data transfer format.



**Figure 3 - Data Transfer Format for I<sup>2</sup>C Interface**

### 2.5.1 Start Condition

Generated by the master (in our case, the ZL50416). 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<sup>2</sup>C 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 acknowledgment-related clock pulse is generated by the master. However, the transmitter releases the SDA line (High) during the acknowledgment clock pulse. Furthermore, the receiver must pull down the SDA line during the 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<sup>2</sup>C interface serves the function of configuring the ZL50416 at boot time. The master is the ZL50416, and the slave is the EEPROM memory.

## 2.6 Synchronous Serial Interface

The synchronous serial interface serves the function of configuring the ZL50416 not at boot time but via a PC. The PC serves as master and the ZL50416 serves as slave. The protocol for the synchronous serial interface is nearly identical to the I<sup>2</sup>C protocol. The main difference is that there is no acknowledgment bit after each byte of data transferred.

The unmanaged ZL50416 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 D0 pin. STROBE- pin is used as the shift clock. AUTOFD- 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 a ABORT pulse to the ZL50416.

A START command is detected when D0 is sampled high when STROBE- rise and D0 is sampled low when STROBE- fall.

An ABORT command is detected when D0 is sampled low when STROBE- rise and D0 is sampled high when STROBE- fall.



### 2.6.1 Write Command

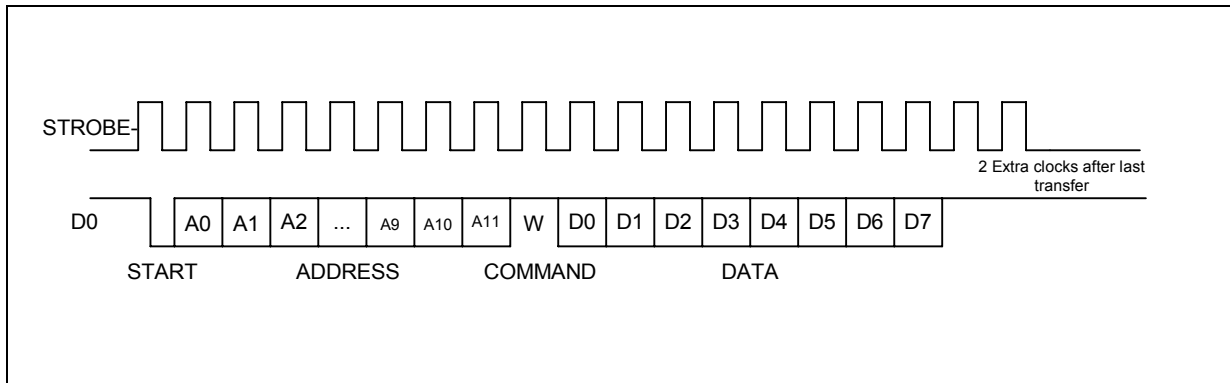


Figure 4 - Write Command

### 2.6.2 Read Command

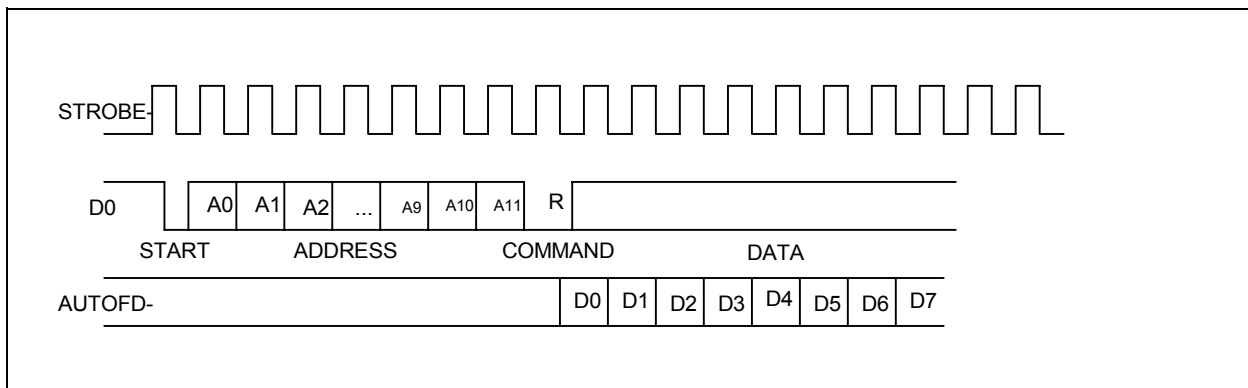


Figure 5 - Read Command

All registers in ZL50416 can be modified through this synchronous serial interface.

## 3.0 ZL50416 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 (SRAM) interface is a 64-bit bus connected to SRAM bank. 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 Rx FIFO 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 4 transmission classes for each of the 16 10/ 100 ports

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.

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 2 multicast queues for each of the 16 10/100 ports. The queue with higher priority has room for 32 entries and the queue with lower priority has room for 64 entries. There is one multicast queue for every two priority classes. For the 10/100 ports to map the 8 transmit priorities into 2 multicast queues, the 2 LSB are discarded.

During scheduling, the TxQ manager treats the unicast queue and the multicast queue of the same class as one logical queue. The older head of line of the two queues is forwarded first.

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.

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 Zarlink Semiconductor 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 receive queue.

## 4.0 Memory Interface

### 4.1 Overview

The ZL50416 provides a 64-bit-wide SRAM bank with a 64-bit. Each DMA can read and write from the SRAM bank. The following figure provides an overview of the ZL50416 SRAM bank.

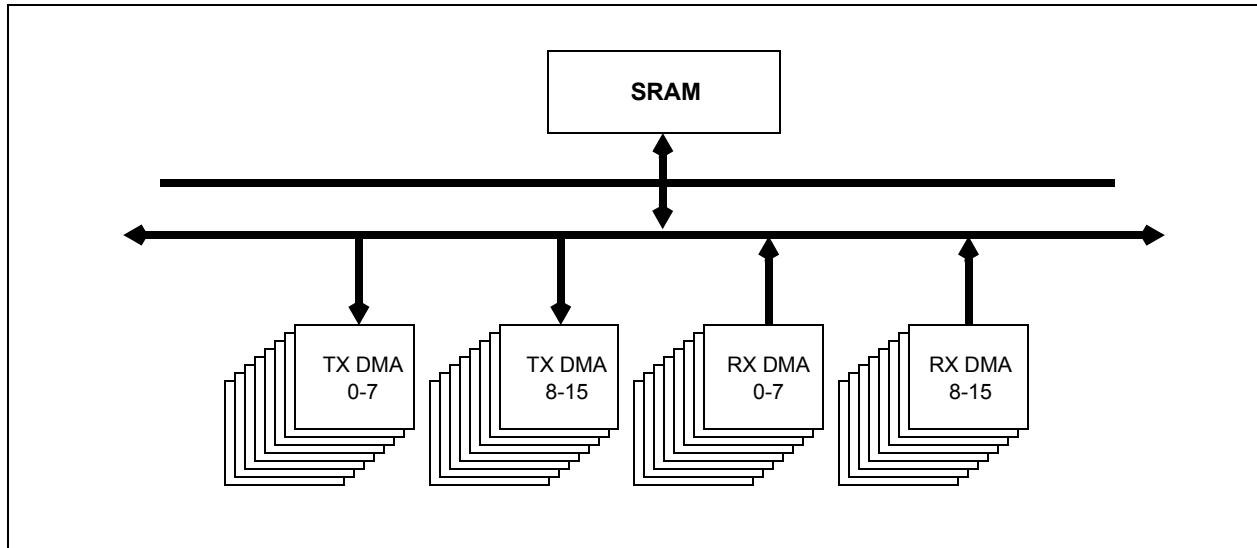


Figure 6 - ZL50416 SRAM Interface Block Diagram (DMAs for 10/100 Ports Only)

### 4.2 Detailed Memory Information

Because the bus for each bank is 64 bits wide, frames are broken into 8-byte granules, written to and read from memory.

### 4.3 Memory Requirements

To support 64K MAC address, 2 MB memory is required. When VLAN support is enabled, 512 entries of the MAC address table are used for storing the VLAN ID at VLAN Index Mapping Table.

Up to 1K Ethernet frame buffers are supported and they will use 1.5 MB of memory. Each frame uses 1536 bytes. The maximum system memory requirement is 2 MB. If less memory is desired, the configuration can scale down.

Memory Bank	Tag based VLAN	Frame Buffer	Max MAC Address
1M	Disable	1K	32K
1M	Enable	1K	31.5K
2M	Disable	2K	64K
2M	Enable	2K	63.5K

Table 1 - Memory Configuration

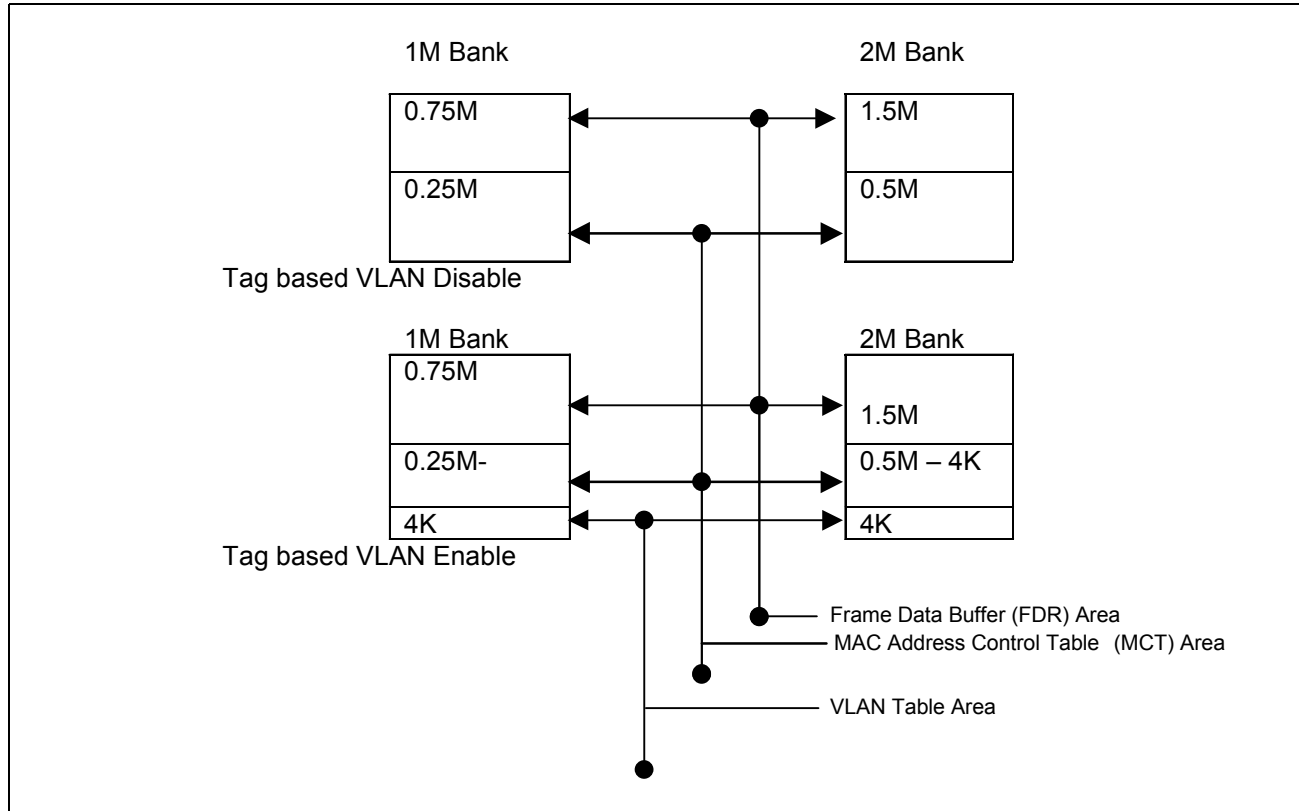


Figure 7 - Memory Map

## 5.0 Search Engine

### 5.1 Search Engine Overview

The ZL50416 search engine is optimized for high throughput searching, with enhanced features to support:

- Up to 64K MAC addresses
- Up to 255 VLAN and IP Multicast groups
- 2 groups of port trunking
- Traffic classification into 4 transmission priorities, and 2 drop precedence levels
- Packet filtering
- Security
- IP Multicast
- Flooding, Broadcast, Multicast Storm Control
- MAC address learning and aging

### 5.2 Basic Flow

Shortly after a frame enters the ZL50416 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 to locate the associated entries in the external hash 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 fast learning mode is enabled, learning and port change will be performed when memory slot is available, and a latter "Learn MAC Address" message is sent to the CPU when CPU queue has room.

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 MAC 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; all others are static. The MAC entry type is stored in the “status” field of the MCT data structure.

**5.3.4 VLAN Table**

The table below provides a mapping from VLAN ID to VLAN index. It is maintained by system software and is checked by the hardware search engine for every incoming frame. This table has 4K entries and is stored in external SRAM. It is organized as 512 × 8 entries (total of 4K VLAN indexes) as shown. Each VLAN index is 8 bits.

VIX7	VIX6	VIX5	VIX4	VIX3	VIX2	VIX1	VIX0
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
VIX4095	VIX4094	VIX4093	VIX4092	VIX4091	VIX4090	VIX4089	VIX4088

**Table 2 - VLAN Index Mapping Table**

Each VIX represents the mapping result from the associated VLAN ID (VLANID = 0x004 is mapped to VIX4). Unused VLAN ID’s have their corresponding VIX programmed to hexadecimal 00. Used VLAN ID’s have their corresponding VIX programmed to hexadecimal 01 through FF. In other words, 255 VLAN’s are supported. The VIX value is a pointer to the entries in the VLAN Index port association table (internal memory).

The VLAN Index port association table is used by both software and hardware. It contains 256 entries. Each entry has 17 fields, such that each field represents the port status of that particular VLAN.

	Port	Not Used	CPU	Not Use	P15	.....	P3	P2	P1	P0
	Bit	63 to 50	49 48	47 32	31 30		7 6	5 4	3 2	1 0
ENTRIES	0									
	1									
	:									
	:									
	255									

**Table 3 - VLAN Index Port Association Table**

Each entry has 64 bits. Each port has a VLAN status field with the following two bits values:

- 00: Port not a member of VLAN
- 01: Port is a member of VLAN, and is subject to aging (Do not use. Used by the aging module)
- 10: Port is a member of VLAN, and is subject to aging
- 11: Portmember of VLAN, and is not subject to aging

**Note:** The VLAN aging time is controlled by register 402h.

## 5.4 MAC Address Filtering

The ZL50416's implementation of intelligent traffic switching provides filters for source and destination MAC addresses. This feature filters unnecessary traffic, thereby providing intelligent control over traffic flows and broadcast traffic.

MAC address filtering allows the ZL50416 to block an incoming packet to an interface when it sees a specified MAC address in either the source address or destination address of the incoming packet. For example, if your network is congested because of high utilization from a MAC address, you can filter all traffic transmitted from that address and restore network flow, while you troubleshoot the problem.

## 5.5 Quality of Service

Quality of Service (QoS) refers to the ability of a network to provide better service to selected network traffic over various technologies. Primary goals of QoS include dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved loss characteristics.

Traditional Ethernet networks have had no prioritization of traffic. Without a protocol to prioritize or differentiate traffic, a service level known as "best effort" attempts to get all the packets to their intended destinations with minimum delay; however, there are no guarantees. In a congested network or when a low-performance switch/router is overloaded, "best effort" becomes unsuitable for delay-sensitive traffic and mission-critical data transmission.

The advent of QoS for packet-based systems accommodates the integration of delay-sensitive video and multimedia traffic onto any existing Ethernet network. It also alleviates the congestion issues that have previously plagued such "best effort" networking systems. QoS provides Ethernet networks with the breakthrough technology to prioritize traffic and ensure that a certain transmission will have a guaranteed minimum amount of bandwidth.

Extensive core QoS mechanisms are built into the ZL50416 architecture to ensure policy enforcement and buffering of the ingress port, as well as weighted fair-queue(WFQ) scheduling at the egress port.

In the ZL50416, QoS-based policies sort traffic into a small number of classes and mark the packets accordingly. The QoS identifier provides specific treatment to traffic in different classes, so that different quality of service is provided to each class. Frame and packet scheduling and discarding policies are determined by the class to which the frames and packets belong. For example, the overall service given to frames and packets in the premium class will be better than that given to the standard class; the premium class is expected to experience lower loss rate or delay.

The ZL50416 supports the following QoS techniques:

- In a port-based setup, any station connected to the same physical port of the switch will have the same transmit priority.
- In a tag-based setup, a 3-bit field in the VLAN tag provides the priority of the packet. This priority can be mapped to different queues in the switch to provide QoS.
- In a TOS/DS-based set up, TOS stands for "Type of Service" that may include "minimize delay," "maximize throughput," or "maximize reliability." Network nodes may select routing paths or forwarding behaviors that are suitably engineered to satisfy the service request.

- In a logical port-based set up, a logical port provides the application information of the packet. Certain applications are more sensitive to delays than others; using logical ports to classify packets can help speed up delay sensitive applications, such as VoIP.

### 5.6 Priority Classification Rule

Figure 8 shows the ZL50416 priority classification rule.

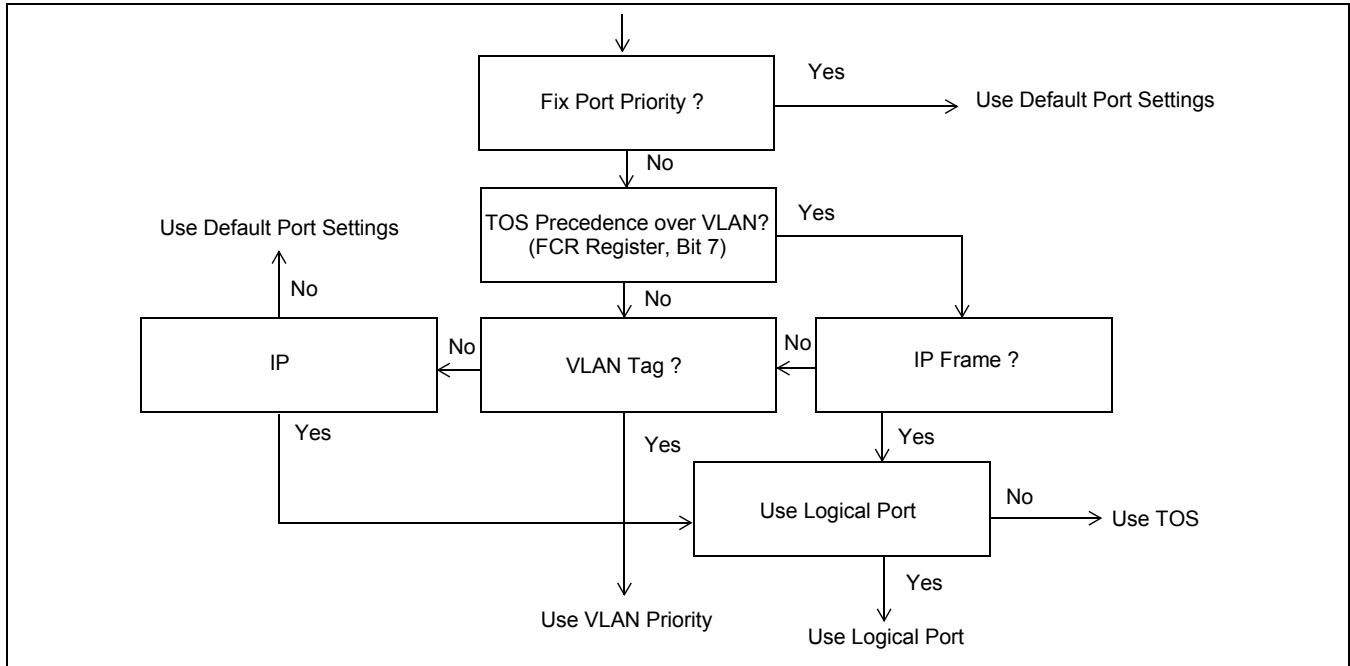


Figure 8 - Priority Classification Rule

### 5.7 Port and Tag Based VLAN

The ZL50416 supports two models for determining and controlling how a packet gets assigned to a VLAN: port priority and tag -based VLAN.

#### 5.7.1 Port-Based VLAN

An administrator can use the PVMAP Registers to configure the ZL50416 for port-based VLAN (see “Registration Definition” on page 42). For example, ports 1-3 might be assigned to the Marketing VLAN, ports 4-6 to the Engineering VLAN, and ports 7-9 to the Administrative VLAN. The ZL50416 determines the VLAN membership of each packet by noting the port on which it arrives. From there, the ZL50416 determines which outgoing port(s) is/are eligible to transmit each packet, or whether the packet should be discarded.

Port Registers	Destination Port Numbers Bit Map				
	15	...	2	1	0
Register for Port #0 PVMAP00_0[7:0] to PVMAP00_1[7:0]	0		1	1	0
Register for Port #1 PVMAP01_0[7:0] to PVMAP01_1[7:0]	0		1	1	1

Table 4 - Port-Based VLAN



	Destination Port Numbers Bit Map				
Register for Port #2 PVMAP02_0[7:0] to PVMAP02_1[7:0]	0		0	0	0
...					
Register for Port #15 PVMAP15_0[7:0] to PVMAP15_1[7:0]	0		0	0	0

**Table 4 - Port-Based VLAN**

For example, in the above table a 1 denotes that an outgoing port is eligible to receive a packet from an incoming port. A 0 (zero) denotes that an outgoing port is not eligible to receive a packet from an incoming port.

In this example:

Data packets received at port #0 are eligible to be sent to outgoing ports 1 and 2.

Data packets received at port #1 are eligible to be sent to outgoing ports 0 and 2.

Data packets received at port #2 are **NOT** eligible to be sent to ports 0 and 1.

### 5.7.2 Tag-Based VLAN

The ZL50416 supports the IEEE 802.1q specification for “tagging” frames. The specification defines a way to coordinate VLANs across multiple switches. In the specification, an additional 4-octet header (or “tag”) is inserted in a frame after the source MAC address and before the frame type. 12 bits of the tag are used to define the VLAN ID. Packets are then switched through the network with each ZL50416 simply swapping the incoming tag for an appropriate forwarding tag rather than processing each packet's contents to determine the path. This approach minimizes the processing needed once the packet enters the tag-switched network. In addition, coordinating VLAN IDs across multiple switches enables VLANs to extend to multiple switches.

Up to 255 VLANs are supported in the ZL50416. The 4 K VLANs specified in the IEEE 802.1q are mapped to 255 VLAN indexes. The mapping is made by the VLAN index mapping table. Based on the VLAN index (VIXn), the source and destination port membership is checked against the content in the VLAN Index Port association table. If the destination port is a member of the VLAN, the packet is forwarded; otherwise it is discarded. If the source port is not a member, a “New VLAN Port” message is sent to the CPU. A filter can be applied to discard the packet if the source port is not a member of the VLAN.

### 5.8 Memory Configurations

The ZL50416 supports the following memory configurations. It supports 1 M and 2M per bank configurations.

Configuration	1 M (Bootstrap pin TSTOUT7 = open)	2 M (Bootstrap pin TSTOUT7 = pull down)	Connections
Single Layer (Bootstrap pin TSTOUT13 = open)	Two 128 K x 32 SRAM/bank or One 128 K x 64 SRAM/bank	Two 256K x 32 SRAM/bank	Connect 0E# and WE#
Double Layer (Bootstrap pin TSTOUT13 = pull down)	NA	Four 128 K x 32 SRAM/bank or Two 128 K x 64 SRAM/bank	Connect 0E0# and WE0# Connect 0E1# and WE1#

**Table 5 - Supported Memory Configurations (Pipeline SBRAM Mode)**

	Frame Data Buffer			
	Only Bank A		Bank A and Bank B	
	1M (SRAM)	2M (SRAM)	1M/bank (SRAM)	2M/bank (SRAM)
ZL50415	X	X		
ZL50416	X	X		
ZL50417			X	X
ZL50418			X	X

Table 6 - Options for Memory Configuration

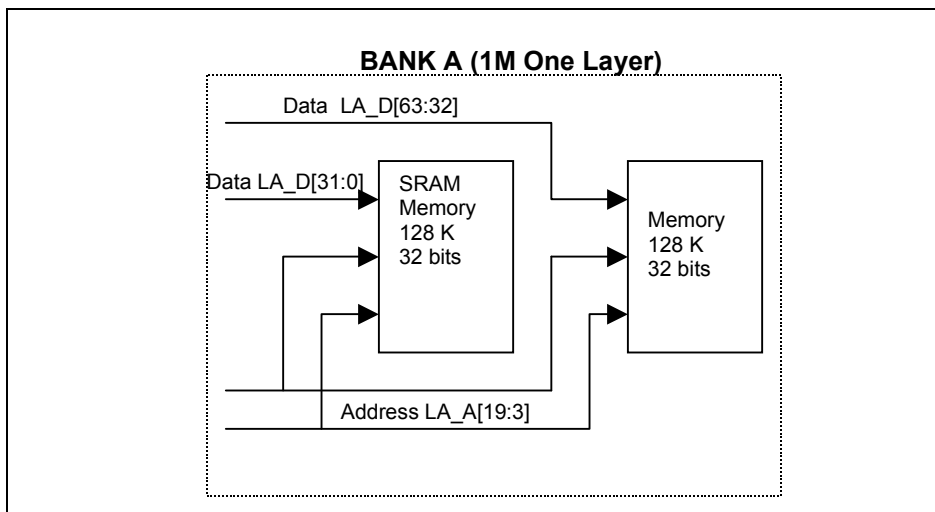
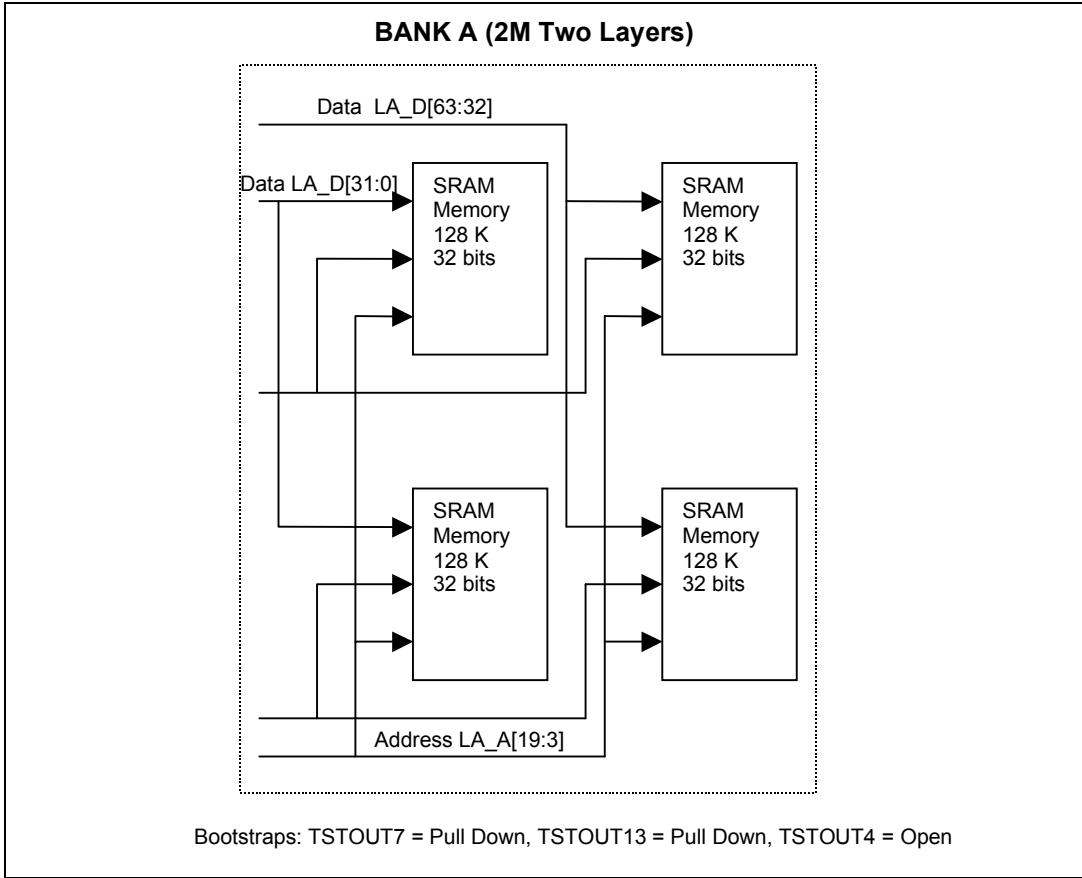
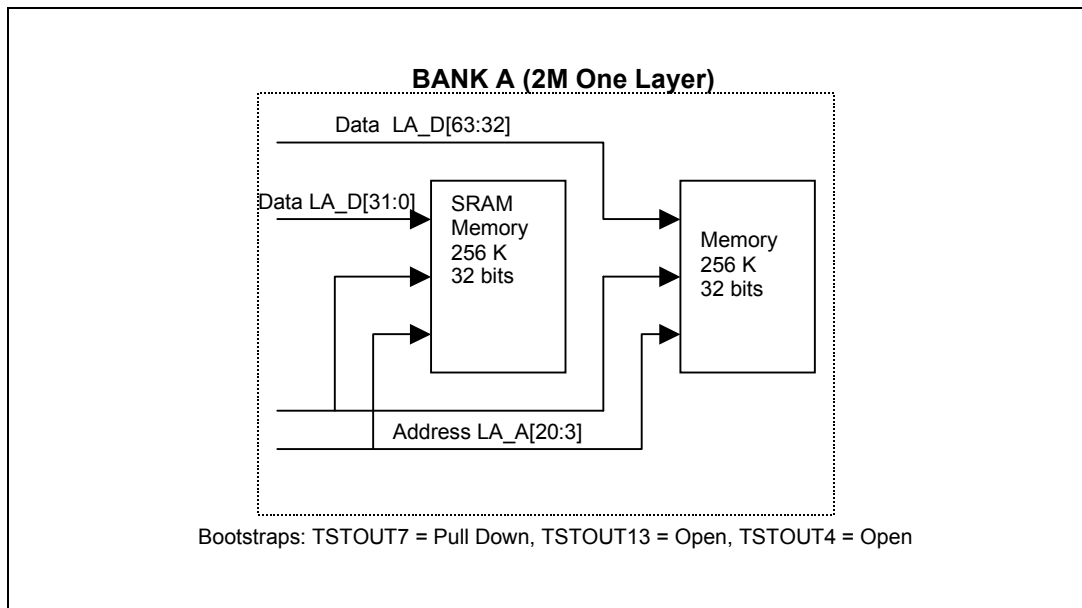


Figure 9 - Memory Configuration For: 1 bank, 1 Layer, 1MB total



**Figure 10 - Memory Configuration For: 1 Bank, 2 Layers, 2MB total**



**Figure 11 - Memory Configuration For: 1 bank, 1 Layer, 2MB**

## 6.0 Frame Engine

### 6.1 Data Forwarding Summary

When a frame 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 are 4 TxSch Q for each 10/100, 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 older HOL between the two queues goes first. For 10/100 ports multicast queue 0 is associated with unicast queue 0 and multicast queue 1 is associated with unicast queue 2.

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 ZL50416 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 7. 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 is an all-encompassing term for which different people have different interpretations. In general, the approach to quality of service described here assumes that we do not know the offered traffic pattern. We also assume that the incoming traffic is not policed or shaped. Furthermore, 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. As an added bonus, although we do not assume anything about the arrival pattern, if the incoming traffic is policed or shaped, we may be able to provide additional assurances about our switch's performance.

Table 7 shows examples of QoS applications with three transmission priorities, but best effort (P0) traffic may form a fourth class with no bandwidth or latency assurances.

Goals	Total Assured Bandwidth (user defined)	Low Drop Probability (low-drop)	High Drop Probability (high-drop)
Highest transmission priority, P3	50 Mbps	Apps: phone calls, circuit emulation. Latency: < 1 ms. Drop: No drop if P3 not oversubscribed.	Apps: training video. Latency: < 1 ms. Drop: No drop if P3 not oversubscribed; first P3 to drop otherwise.
Middle transmission priority, P2	37.5 Mbps	Apps: interactive apps, Web business. Latency: < 4-5 ms. Drop: No drop if P2 not oversubscribed.	Apps: non-critical interactive apps. Latency: < 4-5 ms. Drop: No drop if P2 not oversubscribed; first P2 to drop otherwise.
Low transmission priority, P1	12.5 Mbps	Apps: emails, file backups. Latency: < 16 ms desired, but not critical. Drop: No drop if P1 not oversubscribed.	Apps: casual web browsing. Latency: < 16 ms desired, but not critical. Drop: No drop if P1 not oversubscribed; first to drop otherwise.
Total	100 Mbps		

**Table 7 - Two-dimensional World Traffic**

A class is capable of offering traffic that exceeds the contracted bandwidth. 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, such leniency must not degrade the quality of service (QoS) received by well-behaved classes.

As Table 7 illustrates, the six traffic types may each have their own distinct properties and applications. As shown, classes may receive bandwidth assurances or latency bounds. In the table, P3, the highest transmission class, requires that all frames be transmitted within 1 ms, and receives 50% of the 100 Mbps of bandwidth at that port.

Best-effort (P0) traffic forms a fourth class that only receives bandwidth when none of the other classes have any traffic to offer. It is also possible to add a fourth class that has strict priority over the other three; if this class has even one frame to transmit, then it goes first. In the ZL50416, each 10/100 Mbps port will support four total classes, and each 1000 Mbps port will support eight classes. We will discuss the various modes of scheduling these classes in the next section.

In addition, each transmission class has two subclasses, high-drop and low-drop. Well-behaved users should rarely lose packets. But poorly behaved users – users who send frames 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, and then all frames in the worst case.

Table 7 shows that different types of applications may be placed in different boxes in the traffic table. For example, casual web browsing fits 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 ZL50416: 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 the tables below. For 10/100 Mbps ports, the following registers select these modes:

QOSC24 [7:6] CREDIT\_C00  
 QOSC28 [7:6] CREDIT\_C10  
 QOSC32 [7:6] CREDIT\_C20  
 QOSC36 [7:6] CREDIT\_C30

	P3	P2	P1	P0
Op1 (default)	Delay Bound			BE
Op2	SP	Delay Bound		BE
Op3	SP	WFQ		
Op4	WFQ			

**Table 8 - Four QoS Configurations for a 10/100 Mbps Port**

The default configuration for a 10/100 Mbps port is three delay-bounded queues and one best-effort queue. The delay bounds per class are 0,8 ms for P3, 3.2 ms for P2, and 12.8 ms for P1. Best effort traffic is only served when there is no delay-bounded traffic to be served.

We have a second configuration for a 10/100 Mbps port in which there is one strict priority queue, two delay bounded queues, and one best effort queue. The delay bounds per class are 3.2 ms for P2 and 12.8 ms for P1. If the user is to choose this configuration, it is important that P3 (SP) traffic be either policed or implicitly bounded

(e.g. if the incoming P3 traffic is very light and predictably patterned). Strict priority traffic, if not admission-controlled at a prior stage to the ZL50416, can have a deleterious effect on all other classes' performance.

The third configuration for a 10/100 Mbps port contains one strict priority queue and three 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 ZL50416 may not know 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 ZL50416, we do not enforce a fair bandwidth partition by dropping strict priority traffic. To summarize, dropping to enforce 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 ZL50416 provides the user with a WFQ option with the understanding that delay assurances can not be provided if the incoming traffic pattern is uncontrolled. The user sets four 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 ZL50416 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, queue P0 for a 10/100 port 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 Rate Control

The ZL50416 provides a rate control function on its 10/100 ports. This rate control function applies to the outgoing traffic aggregate on each 10/100 port. It provides a way of reducing the outgoing average rate below full wire speed. Note that the rate control function does not shape or manipulate any particular traffic class. Furthermore, though the average rate of the port can be controlled with this function, the peak rate will still be full line rate.

Two principal parameters are used to control the average rate for a 10/100 port. A port's rate is controlled by allowing, on average, M bytes to be transmitted every N microseconds. Both of these values are programmable. The user can program the number of bytes in 8-byte increments, and the time may be set in units of 10 ms.

The value of M/N will, of course, equal the average data rate of the outgoing traffic aggregate on the given 10/100 port. Although there are many (M,N) pairs that will provide the same average data rate performance, the smaller the time interval N, the "smoother" the output pattern will appear.

In addition to controlling the average data rate on a 10/100 port, the rate control function also manages the maximum burst size at wire speed. The maximum burst size can be considered the memory of the rate control mechanism; if the line has been idle for a long time, to what extent can the port "make up for lost time" by transmitting a large burst? This value is also programmable, measured in 8-byte increments.

Example: Suppose that the user wants to restrict Fast Ethernet port P's average departure rate to 32 Mbps – 32% of line rate – when the average is taken over a period of 10 ms. In an interval of 10 ms, exactly 40000 bytes can be transmitted at an average rate of 32 Mbps.

So how do we set the parameters? The rate control parameters are contained in an internal RAM block accessible through the CPU port (See Programming QoS Registers application note and Processor interface application note). The data format is shown below.

63:40	39:32	31:16	15:0
0	Time interval	Maximum burst size	Number of bytes

As we indicated earlier, the number of bytes is measured in 8-byte increments, so the 16-bit field "Number of bytes" should be set to 40000/8, or 500. In addition, the time interval has to be indicated in units of 10 ms. Though we want the average data rate on port P to be 32 Mbps when measured over an interval of 10 ms, we can also adjust the maximum number of bytes that can be transmitted at full line rate in any single burst. Suppose we wish this limit to be 12 kilobytes. The number of bytes is measured in 8-byte increments, so the 16-bit field "Maximum burst size" is set to 12000/8, or 1500.

## 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 behavior of the WRED logic.

In KB (kilobytes)	P3	P2	P1	High Drop	Low Drop
Level 1 N ≥ 120	P3 ≥ AKB	P2 ≥ BKB	P1 ≥ CKB	X%	0%
Level 2 N ≥ 140				Y%	Z%
Level 3 N ≥ 160				100%	100%

**Table 9 - Weighted Random Early Detection (WRED) logic**



$P_x$  is the total byte count, in the priority queue  $x$ . 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  $P_3*16+P_2*4+P_1$ . If using WFQ scheduling,  $N$  equals  $P_3+P_2+P_1$ . Each drop level from one to three has defined high-drop and low-drop percentages, which indicate the minimum and maximum percentages of the data that can be discarded. The  $X, Y, Z$  percent can be programmed by the register  $RDRC_0, RDRC_1$ . In Level 3, all packets are dropped if the bytes in each priority queue exceed the threshold. Parameters  $A, B, C$  are the byte count thresholds for each priority queue. They can be programmed by the QoS control register (refer to the register group 5). See Programming QoS Registers application note for more information.

## 7.8 Buffer Management

Because the number of 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 ZL50416. Our buffer management scheme is designed to divide the total buffer space into numerous reserved regions and one shared pool, as shown in Figure 12.

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 ZL50416, 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 them to the frame drop discipline after classifying.

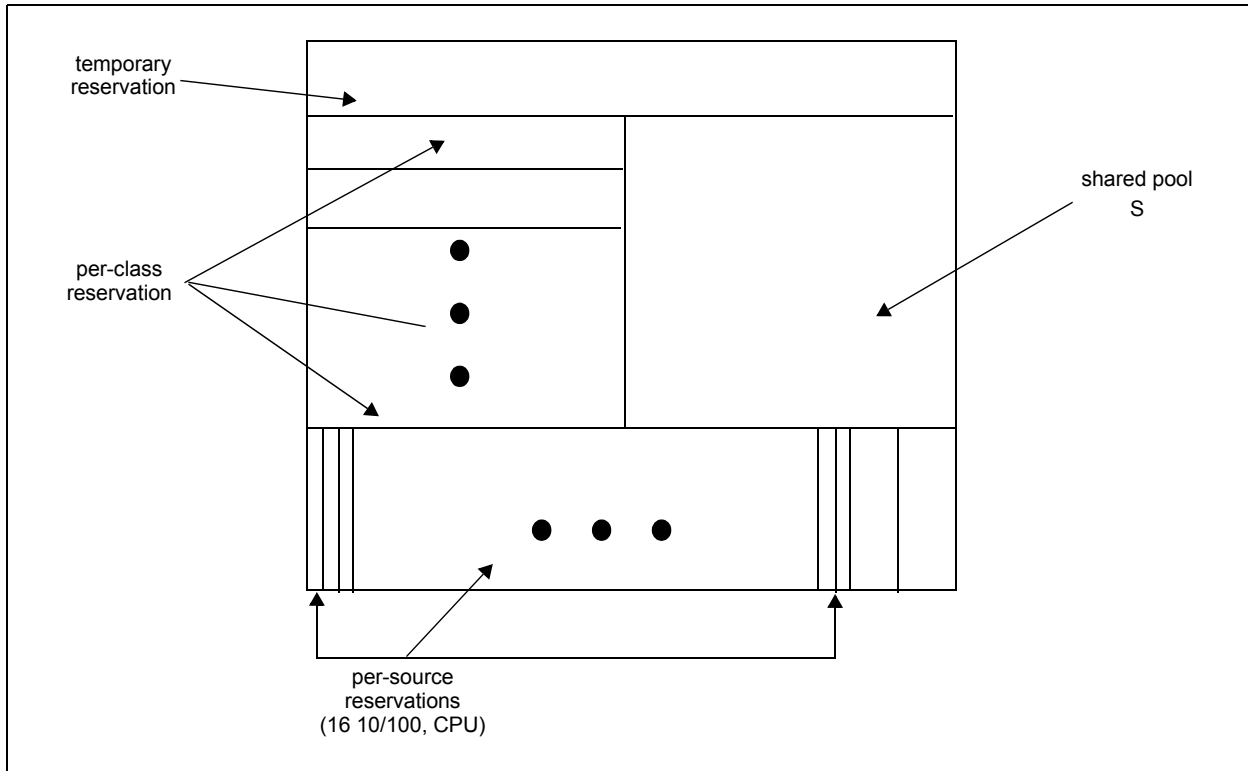
Six reserved sections, one for each of the first six priority classes, ensure a programmable number of FDB slots per class. The lowest two classes do not receive any buffer reservation. Furthermore, even for 10/100 Mbps ports, a frame is stored in the region of the FDB corresponding to its class. As we have indicated, the eight classes use only four transmission scheduling queues for 10/100 Mbps ports, but as far as buffer usage is concerned, there are still eight distinguishable classes.

Another segment of the FDB reserves space for each of the 17 regions — 16 ports for Ethernet and one CPU port (port number 24). One parameter can be set, one for the source port reservation for 10/100 Mbps ports and CPU port. These 17 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 frame engine allocates the frames first in the six priority sections. When the priority section is full or the packet has priority 1 or 0, the frame is allocated in the shared pool. Once the shared pool is full the frames are allocated in the section reserved for the source port.

The following registers define the size of each section of the Frame data Buffer:

- PR100- Port Reservation for 10/100 Ports
- SFCB- Share FCB Size
- C2RS- Class 2 Reserve Size
- C3RS- Class 3 Reserve Size
- C4RS- Class 4 Reserve Size
- C5RS- Class 5 Reserve Size
- C6RS- Class 6 Reserve Size
- C7RS- Class 7 Reserve Size



**Figure 12 - Buffer Partition Scheme Used to Implement Buffer Management**

### 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, we also drop frames when global buffer space becomes scarce. The function of buffer management is to make sure that such dropping causes as little blocking as possible.

## 7.9 ZL50416 Flow Control Basics

Because frame loss is unacceptable for some applications, the ZL50416 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 that is 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 uncongested 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 ZL50416, 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 ZL50416 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 ZL50416’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 ZL50416’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 ZL50416’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, flow control for multicast frames is triggered by a global buffer counter. When the system exceeds a programmable threshold of multicast packets, Xoff is triggered. Xon is triggered when the system returns below this 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. How so? The ZL50416 checks each destination to which a multicast packet is headed. For each destination port, the occupancy of the lowest-priority transmission multicast 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 a 26-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 26-bit vector is reset to zero.

The ZL50416 also provides the option of disabling VLAN multicast flow control.

**Note:** If per-Port flow control is on, QoS performance will be affected.

## 7.10 Mapping to IETF Diffserv Classes

For 10/100 Mbps ports, the classes of Table 10 are merged in pairs—one class corresponding to NM+EF, two AF classes, and a single BE class.

ZL	P3	P2	P1	P0
IETF	NM+EF	AF0	AF1	BE0

**Table 10 - Mapping between ZL50416 and IETF Diffserv Classes for 10/100 Ports**

Features of the ZL50416 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 Option of strict priority scheduling No dropping if admission controlled
Assured forwarding (AF)	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)	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 11 - ZL50416 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 when using stack all of the ports in a group must be in the same ZL50416.

There are two trunk groups.

Load distribution among the ports in a trunk for unicast is performed using hashing based on source MAC address and destination MAC address. Three other options include source MAC address only, destination MAC address only, and source port (in bidirectional ring mode 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 ZL50416 also provides a safe fail-over mode for port trunking automatically. If one of the ports in the trunking group goes down, the ZL50416 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 port 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.

A hash key, based on some combination of the source and destination MAC addresses for the current packet, selects the appropriate forwarding port, as specified in the Trunk\_Hash registers.

### 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.

Preventing the multicast packet 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.

### 8.4 Unmanaged Trunking

In unmanaged mode, 2 trunk groups are supported. Groups 0 and 1 can trunk up to 4 10/100 ports. The supported combinations are shown in the following table.

Group 0	Port 0	Port 1	Port 2	Port 3
	✓	✓		
	✓	✓	✓	
	✓	✓	✓	✓

**Table 12 - Select via trunk0\_mode register**

Group 1	Port 4	Port 5	Port 6	Port 7
	✓	✓		
	✓	✓	✓	✓

**Table 13 - Select via trunk1\_mode register**

In unmanaged mode, the trunks are individually enabled/disabled by controlling pin trunk0,1.

## 9.0 Port Mirroring

### 9.1 Port Mirroring Features

The received or transmitted data of any 10/100 port in the ZL50416 chip can be “mirrored” to any other port. We support two such mirrored source-destination pairs. A mirror port can not also serve as a data port.

### 9.2 Setting Registers for Port Mirroring

**MIRROR1\_SRC:** Sets the source port for the first port mirroring pair. Bits [4:0] select the source port to be mirrored. An illegal port number is used to disable mirroring (which is the default setting). Bit [5] is used to select between ingress (Rx) or egress (Tx) data.

**MIRROR1\_DEST:** Sets the destination port for the first port mirroring pair. Bits [4:0] select the destination port to be mirrored.

**MIRROR2\_SRC:** Sets the source port for the second port mirroring pair. Bits [4:0] select the source port to be mirrored. An illegal port number is used to disable mirroring (which is the default setting). Bit [5] is used to select between ingress (Rx) or egress (Tx) data.

**MIRROR2\_DEST:** Sets the destination port for the second port mirroring pair. Bits [4:0] select the destination port to be mirrored. The default is port 0.

Refer to Port Mirroring Application Notes for further information.

## 10.0 GPSI (7WS) Interface

### 10.1 GPSI connection

The 10/100 RMII ethernet port can function in GPSI (7WS) mode when the corresponding TXEN pin is strapped low with a 1K pull down resistor. In this mode, the TXD[0], TXD[1], RXD[0] and RXD[1] serve as TX data, TX clock, RX data and RX clock respectively. The link status and collision from the PHY are multiplexed and shifted into the switch device through external glue logic. The duplex of the port can be controlled by programming the ECR register. The GPSI interface can be operated in port based VLAN mode only.

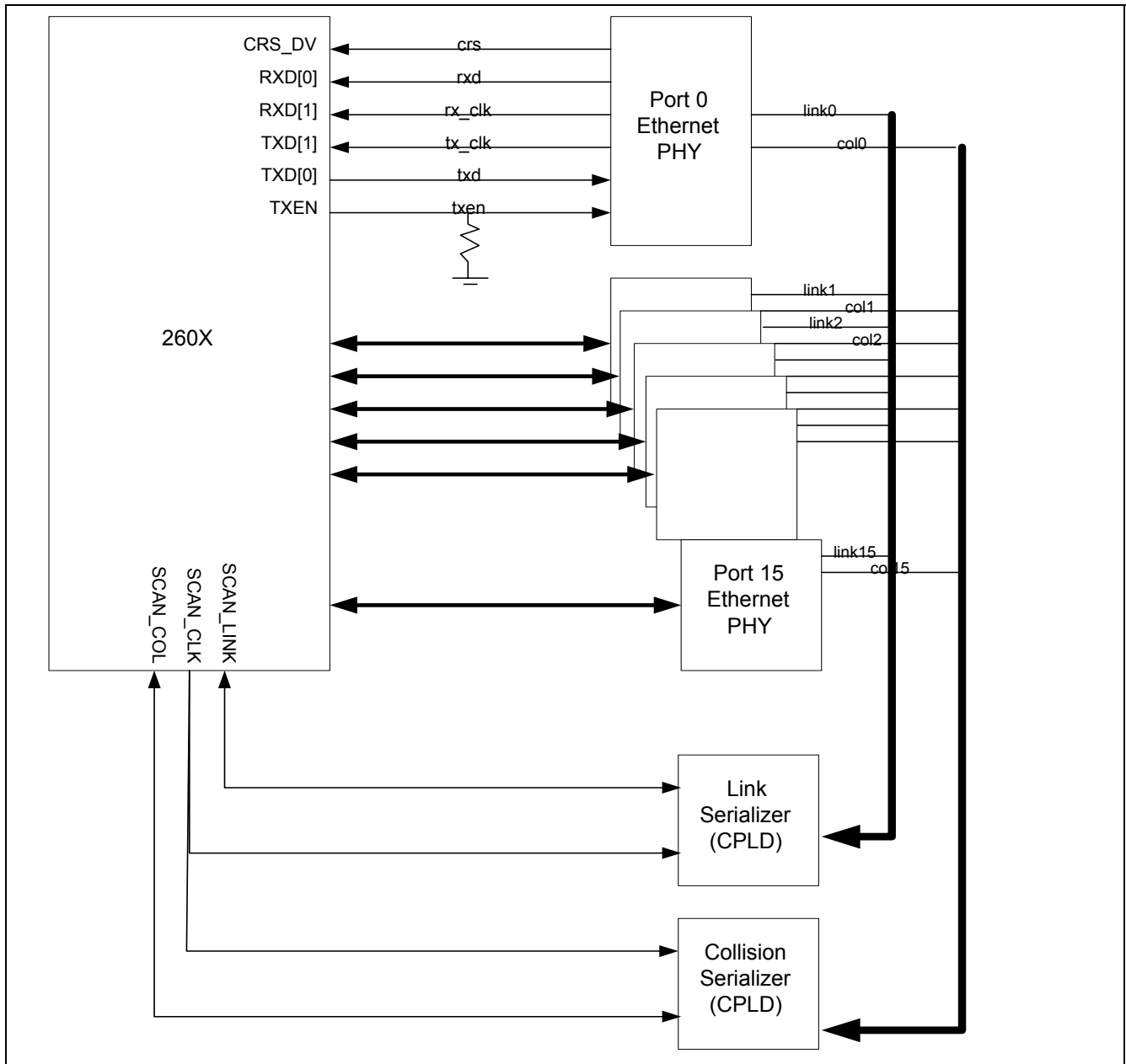


Figure 13 - GPSI (7WS) mode connection diagram

## 10.2 SCAN LINK and SCAN COL interface

An external CPLD logic is required to take the link signals and collision signals from the GPSI PHYS and shift them into the switch device. The switch device will drive out a signature to indicate the start of the sequence. After that, the CPLD should shift in the link and collision status of the PHYS as shown in the figure. The extra link status indicates the polarity of the link signal. One indicates the polarity of the link signal is active high.

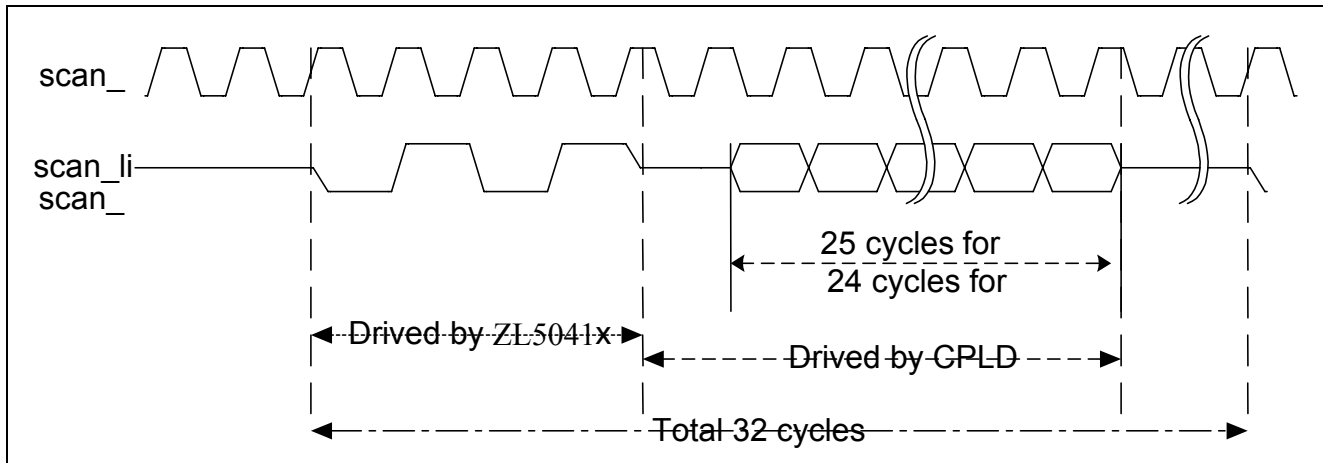


Figure 14 - SCAN LINK and SCAN COLLISION status diagram

## 11.0 LED Interface

### 11.1 LED Interface Introduction

A serial output channel provides port status information from the ZL50416 chips. It requires three additional pins.

LED\_CLK at 12.5 MHz

LED\_SYN a sync pulse that defines the boundary between status frames

LED\_DATA a continuous serial stream of data for all status LEDs that repeats once every frame time.

A low cost external device (44 pin PAL) is used to decode the serial data and to drive an LED array for display. This device can be customized for different needs.

### 11.2 Port Status

In the ZL50416, each port has 8 status indicators, each represented by a single bit. The 8 LED status indicators are:

- Bit 0: Flow control
- Bit 1: Transmit data
- Bit 2: Receive data
- Bit 3: Activity (where activity includes either transmission or reception of data)
- Bit 4: Link up
- Bit 5: Speed (1= 100 Mb/s; 0= 10 Mb/s)
- Bit 6: Full-duplex
- Bit 7: Collision

Eight clocks are required to cycle through the eight status bits for each port.



When the LED\_SYN pulse is asserted, the LED interface will present 256 LED clock cycles with the clock cycles providing information for the following ports.

- Port 0 (10/100): cycles #0 to cycle #7
- Port 1 (10/100): cycles#8 to cycle #15
- Port 2 (10/100): cycle #16 to cycle #23
- ...
- Port 15 (10/100): cycle #120 to cycle #127
- Reserved: cycle #128 to cycle #207
- Byte 26 (additional status): cycle #208 to cycle #215
- Byte 27 (additional status): cycle #216 to cycle #223

Cycles #224 to 256 present data with a value of zero.

Byte 26 and byte 27 provides bist status

- 26[1:0]: Reserved
- 26[2]: initialization done
- 26[3]: initialization start
- 26[4]: checksum ok
- 26[5]: link\_init\_complete
- 26[6]: bist\_fail
- 26[7]: ram\_error
- 27[0]: bist\_in\_process
- 27[1]: bist\_done

### 11.3 LED Interface Timing Diagram

The signal from the ZL50416 to the LED decoder is shown in Figure .

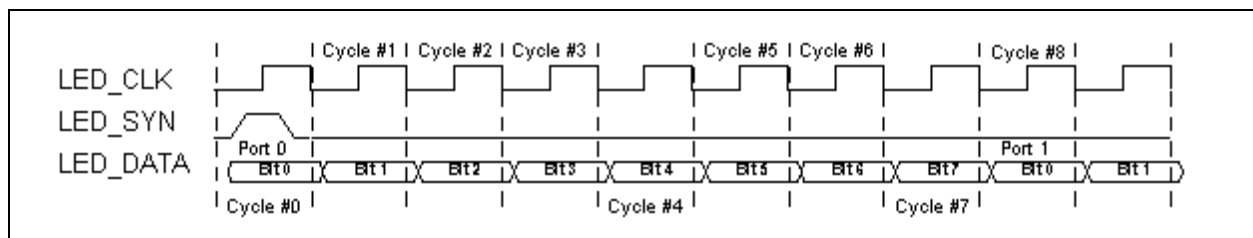


Figure 15 - Timing Diagram of LED Interface

## 12.0 Hardware Statistics Counter

### 12.1 Hardware Statistics Counters List

ZL50416 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.

31	30	26	25	0
Status Wrapped Signal				
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-I	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-I	Frames with Length Between 128-255 Bytes		
B[17]	A-u	Frames with Length Between 256-511 Bytes		
B[18]	B-I	Frames with Length Between 512-1023 Bytes		
B[19]	B-u	Frames with Length Between 1024-1528 Bytes		
B[20]	C-I	Fragments		
B[21]	C-U1	Alignment Error		
B[22]	C-U	Undersize Frames		
B[23]	D-I	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]

## 12.2 IEEE 802.3 HUB Management (RFC 1516)

### 12.2.1 Event Counters

#### 12.2.1.1 ReadableOctet

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

Frame size:  $\geq 64$  bytes,  $\leq 1522$  bytes if VLAN Tagged;  
1518 bytes if not VLAN Tagged

No FCS (i.e. checksum) error

No collisions

#### 12.2.1.2 ReadableFrame

Counts number of good valid frames received.

Frame size:  $\geq 64$  bytes,  $\leq 1522$  bytes if VLAN Tagged;  
1518 bytes if not VLAN Tagged

No FCS error

No collisions

**12.2.2 FCSErrors**

Counts number of valid frames received with bad FCS.

Frame size:  $\geq 64$  bytes,  $\leq 1522$  bytes if VLAN Tagged;  
1518 bytes if not VLAN Tagged

No framing error

No collisions

**12.2.2.1 AlignmentErrors**

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

Frame size:  $\geq 64$  bytes,  $\leq 1522$  bytes if VLAN Tagged;  
1518 bytes if not VLAN Tagged

No framing error

No collisions

**12.2.2.2 FrameTooLongs**

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

Frame size:  $\geq 64$  bytes,  $> 1522$  bytes if VLAN Tagged;  
1518 bytes if not VLAN Tagged

FCS error: don't care

Framing error: don't care

No collisions

**12.2.2.3 ShortEvents**

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

Frame size:  $\geq 64$  bytes,  $< 10$  bytes

FCS error: don't care

Framing error: don't care

No collisions

**12.2.2.4 Runts**

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

Frame size:  $\geq 10$  bytes,  $< 64$  bytes

FCS error: don't care

Framing error: don't care

No collisions

### 12.2.2.5 Collisions

Counts number of collision events.

Frame size: any size

### 12.2.2.6 LateEvents

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

Frame size: any size

Events are also counted by collision counter

### 12.2.2.7 VeryLongEvents

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

Frame size: > Jabber

### 12.2.2.8 DataRateMisatches

For repeaters or HUB application only.

### 12.2.2.9 AutoPartitions

For repeaters or HUB application only.

### 12.2.2.10 TotalErrors

Sum of the following errors:

FCS errors

Alignment errors

Frame too long

Short events

Late events

Very long events

## 12.3 IEEE – 802.1 Bridge Management (RFC 1286)

### 12.3.1 Event Counters

#### 12.3.1.1 InFrames

Counts number of frames received by this port or segment.

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

### 12.3.1.2 OutFrames

Counts number of frames transmitted by this port.

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

### 12.3.1.3 InDiscards

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

### 12.3.1.4 DelayExceededDiscards

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

### 12.3.1.5 MtuExceededDiscards

Counts number of frames discarded due to excessive size.

## 12.4 RMON – Ethernet Statistic Group (RFC 1757)

### 12.4.1 Event Counters

#### 12.4.1.1 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.

#### 12.4.1.2 Octets

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

#### 12.4.1.3 BroadcastPkts

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

Does not include non-broadcast multicast frames.

#### 12.4.1.4 MulticastPkts

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

Does not include broadcast frames.

#### 12.4.1.5 CRCAAlignErrors

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

#### 12.4.1.6 UndersizePkts

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

Frame size: < 64 bytes,  
No FCS error  
No framing error  
No collisions

#### 12.4.1.7 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

#### 12.4.1.8 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

#### 12.4.1.9 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

#### 12.4.1.10 Collisions

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

## 13.0 Register Definition

### 13.1 ZL50416 Register Description

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
<b>0. ETHERNET Port Control Registers Substitute [N] with Port number (0..F,18)</b>						
ECR1P"N"	Port Control Register 1 for Port N	000 + 2 x N	R/W	000-018	020	
ECR2P"N"	Port Control Register 2 for Port N	001 + 2 x N	R/W	01B-033	000	
<b>1. VLAN Control Registers Substitute [N] with Port number (0..F,18)</b>						
AVTCL	VLAN Type Code Register Low	100	R/W	036	000	
AVTCH	VLAN Type Code Register High	101	R/W	037	081	
PVMAP"N"_0	Port "N" Configuration Register 0	102 + 4N	R/W	038-050	0FF	
PVMAP"N"_1	Port "N" Configuration Register 1	103 + 4N	R/W	053-06B	0FF	
PVMAP"N"_3	Port "N" Configuration Register 3	105 + 4N	R/W	089-0A1	007	
PVMODE	VLAN Operating Mode	170	R/W	0A4	000	
PVROUTE7-0	VLAN Router Group Enable	171-178	R/W	NA	000	
<b>2. TRUNK Control Registers</b>						
TRUNK0_L	Trunk Group 0 Low	200	R/W	NA	000	
TRUNK0_M	Trunk Group 0 Medium	201	R/W	NA	000	
TRUNK0_MODE	Trunk Group 0 Mode	203	R/W	0A5	003	
TRUNK0_HASH0	Trunk Group 0 Hash 0 Destination Port	204	R/W	NA	000	
TRUNK0_HASH1	Trunk Group 0 Hash 1 Destination Port	205	R/W	NA	001	
TRUNK0_HASH2	Trunk Group 0 Hash 2 Destination Port	206	R/W	NA	002	
TRUNK0_HASH3	Trunk Group 0 Hash 3 Destination Port	207	R/W	NA	003	
TRUNK1_L	Trunk Group 1 Low	208	R/W	NA	000	
TRUNK1_M	Trunk Group 1 Medium	209	R/W	NA	000	
TRUNK1_MODE	Trunk Group 1 Mode	20B	R/W	0A6	003	
TRUNK1_HASH0	Trunk Group 1 Hash 0 Destination Port	20C	R/W	NA	004	
TRUNK1_HASH1	Trunk Group 1 Hash 1 Destination Port	20D	R/W	NA	005	
TRUNK1_HASH2	Trunk Group 1 Hash 2 Destination Port	20E	R/W	NA	006	
TRUNK1_HASH3	Trunk Group 1 Hash 3 Destination Port	20F	R/W	NA	007	
Multicast_HASH0-0	Multicast hash result 0 mask byte 0	220	R/W	NA	0FF	
Multicast_HASH0-1	Multicast hash result 0 mask byte 1	221	R/W	NA	0FF	

**Table 14 - Register Description**



Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
Multicast_HASH0-2	Multicast hash result 0 mask byte 2	222	R/W	NA	0FF	
Multicast_HASH0-3	Multicast hash result 0 mask byte 3	223	R/W	NA	0FF	
Multicast_HASH1-0	Multicast hash result 1 mask byte 0	224	R/W	NA	0FF	
Multicast_HASH1-1	Multicast hash result 1 mask byte 1	225	R/W	NA	0FF	
Multicast_HASH1-2	Multicast hash result 1 mask byte 2	226	R/W	NA	0FF	
Multicast_HASH1-3	Multicast hash result 1 mask byte 3	227	R/W	NA	0FF	
Multicast_HASH2-0	Multicast hash result 2 mask byte 0	228	R/W	NA	0FF	
Multicast_HASH2-1	Multicast hash result 2 mask byte 1	229	R/W	NA	0FF	
Multicast_HASH2-2	Multicast hash result 2 mask byte 2	22A	R/W	NA	0FF	
Multicast_HASH2-3	Multicast hash result 2 mask byte 3	22B	R/W	NA	0FF	
Multicast_HASH3-0	Multicast hash result 3 mask byte 0	22C	R/W	NA	0FF	
Multicast_HASH3-1	Multicast hash result 3 mask byte 1	22D	R/W	NA	0FF	
Multicast_HASH3-2	Multicast hash result 3 mask byte 2	22E	R/W	NA	0FF	
Multicast_HASH3-3	Multicast hash result 3 mask byte 3	22F	R/W	NA	0FF	
<b>3. CPU Port Configuration</b>						
MAC0	CPU MAC Address byte 0	300	R/W	NA	000	
MAC1	CPU MAC Address byte 1	301	R/W	NA	000	
MAC2	CPU MAC Address byte 2	302	R/W	NA	000	
MAC3	CPU MAC Address byte 3	303	R/W	NA	000	
MAC4	CPU MAC Address byte 4	304	R/W	NA	000	
MAC5	CPU MAC Address byte 5	305	R/W	NA	000	
INT_MASK0	Interrupt Mask 0	306	R/W	NA	000	
INTP_MASK"N"	Interrupt Mask for MAC Port 2N, 2N+1	310+N (310-313)	R/W	NA	000	
RQS	Receive Queue Select	323	R/W	NA	000	
RQSS	Receive Queue Status	324	RO	NA	N/A	
TX_AGE	Transmission Queue Aging Time	325	R/W	0A7	008	
<b>4. Search Engine Configurations</b>						
AGETIME_LOW	MAC Address Aging Time Low	400	R/W	0A8	1M:05C/ 2M:02E	
AGETIME_HIGH	MAC Address Aging Time High	401	R/W	0A9	000	
V_AGETIME	VLAN to Port Aging Time	402	R/W	NA	0FF	

Table 14 - Register Description (continued)

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
SE_OPMODE	Search Engine Operating Mode	403	R/W	NA	000	
SCAN	Scan control register	404	R/W	NA	000	
<b>5. Buffer Control and QOS Control</b>						
FCBAT	FCB Aging Timer	500	R/W	0AA	0FF	
QOSC	QOS Control	501	R/W	0AB	000	
FCR	Flooding Control Register	502	R/W	0AC	008	
AVPML	VLAN Priority Map Low	503	R/W	0AD	000	
AVPMM	VLAN Priority Map Middle	504	R/W	0AE	000	
AVPMH	VLAN Priority Map High	505	R/W	0AF	000	
TOSPML	TOS Priority Map Low	506	R/W	0B0	000	
TOSPMM	TOS Priority Map Middle	507	R/W	0B1	000	
TOSPMH	TOS Priority Map High	508	R/W	0B2	000	
AVDM	VLAN Discard Map	509	R/W	0B3	000	
TOSDML	TOS Discard Map	50A	R/W	0B4	000	
BMRC	Broadcast/Multicast Rate Control	50B	R/W	0B5	000	
UCC	Unicast Congestion Control	50C	R/W	0B6	1M:008/ 2M:010	
MCC	Multicast Congestion Control	50D	R/W	0B7	050	
PR100	Port Reservation for 10/100 Ports	50E	R/W	0B8	1M:035/ 2M:058	
SFCB	Share FCB Size	510	R/W	0BA	1M:046/ 2M:0E6	
C2RS	Class 2 Reserve Size	511	R/W	0BB	000	
C3RS	Class 3 Reserve Size	512	R/W	0BC	000	
C4RS	Class 4 Reserve Size	513	R/W	0BD	000	
C5RS	Class 5 Reserve Size	514	R/W	0BE	000	
C6RS	Class 6 Reserve Size	515	R/W	0BF	000	
C7RS	Class 7 Reserve Size	516	R/W	0C0	000	
QOSC"N"	QOS Control (N=0 - 5)	517- 51C	R/W	0C1-0C6	000	
	QOS Control (N=6 - 11)	51D- 522	R/W	NA	000	
	QOS Control (N=12 - 23)	523- 52E	R/W	0C7-0D2	000	
	QOS Control (N=24 - 59)	52F- 552	R/W	NA	000	

Table 14 - Register Description (continued)

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
RDR0	WRED Drop Rate Control 0	553	R/W	0FB	08F	
RDR1	WRED Drop Rate Control 1	554	R/W	0FC	088	
USER_PORT"N"_LOW	User Define Logical Port "N" Low (N=0-7)	580 + 2N	R/W	0D6-0DD	000	
USER_PORT"N"_HIGH	User Define Logical Port "N" High	581 + 2N	R/W	0DE-0E5	000	
USER_PORT1:0_PRIORITY	User Define Logic Port 1 and 0 Priority	590	R/W	0E6	000	
USER_PORT3:2_PRIORITY	User Define Logic Port 3 and 2 Priority	591	R/W	0E7	000	
USER_PORT5:4_PRIORITY	User Define Logic Port 5 and 4 Priority	592	R/W	0E8	000	
USER_PORT7:6_PRIORITY	User Define Logic Port 7 and 6 Priority	593	R/W	0E9	000	
USER_PORT_ENABLE	User Define Logic Port Enable	594	R/W	0EA	000	
WLPP10	Well known Logic Port Priority for 1 and 0	595	R/W	0EB	000	
WLPP32	Well known Logic Port Priority for 3 and 2	596	R/W	0EC	000	
WLPP54	Well known Logic Port Priority for 5 and 4	597	R/W	0ED	000	
WLPP76	Well-known Logic Port Priority for 7 & 6	598	R/W	0EE	000	
WLPE	Well known Logic Port Enable	599	R/W	0EF	000	
RLOWL	User Define Range Low Bit7:0	59A	R/W	0F4	000	
RLOWH	User Define Range Low Bit 15:8	59B	R/W	0F5	000	
RHIGHL	User Define Range High Bit 7:0	59C	R/W	0D3	000	
RHIGHH	User Define Range High Bit 15:8	59D	R/W	0D4	000	
RPRIORITY	User Define Range Priority	59E	R/W	0D5	000	
CPUQOSC1~3	Byte limit for TxQ on CPU port	5A0-5A2	R/W	NA	000	
<b>6. MISC Configuration Registers</b>						
MII_OP0	MII Register Option 0	600	R/W	0F0	000	
MII_OP1	MII Register Option 1	601	R/W	0F1	000	
FEN	Feature Registers	602	R/W	0F2	010	
MIIC0	MII Command Register 0	603	R/W	N/A	000	

Table 14 - Register Description (continued)

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
MIIC1	MII Command Register 1	604	R/W	N/A	000	
MIIC2	MII Command Register 2	605	R/W	N/A	000	
MIIC3	MII Command Register 3	606	R/W	N/A	000	
MIID0	MII Data Register 0	607	RO	N/A	N/A	
MIID1	MII Data Register 1	608	RO	N/A	N/A	
LED	LED Control Register	609	R/W	0F3	000	
SUM	EEPROM Checksum Register	60B	R/W	0FF	000	
<b>7. Port Mirroring Controls</b>						
MIRROR1_SRC	Port Mirror 1 Source Port	700	R/W	N/A	07F	
MIRROR1_DEST	Port Mirror 1 Destination Port	701	R/W	N/A	017	
MIRROR2_SRC	Port Mirror 2 Source Port	702	R/W	N/A	0FF	
MIRROR2_DEST	Port Mirror 2 Destination Port	703	R/W	N/A	000	
<b>F. Device Configuration Register</b>						
GCR	Global Control Register	F00	R/W	N/A	000	
DCR	Device Status and Signature Register	F01	RO	N/A	N/A	
DCR1	Chip status	F02	RO	N/A	N/A	
DPST	Device Port Status Register	F03	R/W	N/A	000	
DTST	Data read back register	F04	RO	N/A	N/A	
DA	DA Register	FFF	RO	N/A	DA	

Table 14 - Register Description (continued)

## 13.2 Directly Accessed Registers

### 13.2.1 INDEX\_REG0

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

### 13.2.2 INDEX\_REG1 (only needed for 8-bit mode)

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

### 13.2.3 DATA\_FRAME\_REG

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

### 13.2.4 CONTROL\_FRAME\_REG

- CPU transmit/receive switch frames. (8/16 bits)
- Address = 3 (read/write)
- Format:
  - 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

### 13.2.5 COMMAND&STATUS Register

- CPU interface commands (write) and status
- Address = 4 (read/write)
- When the CPU **writes** to this register
  - Bit [0]: • Set Control Frame Receive buffer ready, after CPU writes a complete frame into the buffer. This bit is self-cleared.
  - Bit [1]: • Set Control Frame Transmit buffer1 ready, after CPU reads out a complete frame from the buffer. This bit is self-cleared.
  - Bit [2]: • Set Control Frame Transmit buffer2 ready, after CPU reads out a complete frame from the buffer. This bit is self-cleared.
  - Bit [3]: • Set this bit to indicate CPU received a whole frame (transmit FIFO frame receive done), and flushed the rest of frame fragment, If occurs. This bit will be self-cleared.
  - Bit [4]: • Set this bit to indicate that the following Write to the Receive FIFO is the last one (EOF). This bit will be self-cleared.
  - Bit [5]: • Set this bit to re-start the data that is sent from the CPU to Receive FIFO (re-align). This feature can be used for software debug. For normal operation must be '0'.
  - Bit [6]: • Do not use. Must be '0'
  - Bit [7]: • Reserved

When the CPU **reads** this register:

- Bit [0]: • Control Frame receive buffer ready, CPU can write a new frame
  - 1 – CPU can write a new control command 1
  - 0 – CPU has to wait until this bit is 1 to write a new control command 1
- Bit [1]: • Control Frame transmit buffer1 ready for CPU to read
  - 1 – CPU can read a new control command 1
  - 0 – CPU has to wait until this bit is 1 to read a new control command
- Bit [2]: • Control Frame transmit buffer2 ready for CPU to read
  - 1 – CPU can read a new control command 1
  - 0 – CPU has to wait until this bit is 1 to read a new control command
- Bit [3]: • Transmit FIFO has data for CPU to read (TXFIFO\_RDY)

- Bit [4]: • Receive FIFO has space for incoming CPU frame (RXFIFO\_SPOK)
- Bit [5]: • Transmit FIFO End Of Frame (TXFIFO\_EOF)
- Bit [6]: • Reserve
- Bit [7]: • Reserve

### 13.2.6 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]: • Reserved
- Bit [7:4]: • Reserved

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

### 13.2.7 Control Command 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

### 13.2.8 Control Command Frame Buffer2 Access Register

- Address = 7 (read only)
- When CPU reads this register, data is read from the Control Command Frame Transmit Buffer1

## Indirectly Accessed Registers

### 13.3 (Group 0 Address) MAC Ports Group

#### 13.3.1 ECR1Pn: Port N Control Register

I<sup>2</sup>C Address 000 - 018; CPU Address:0000+2xN (N = port number)

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

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

- Bit [0]      - 1 - Flow Control Off  
               - 0 - Flow Control On
- When Flow Control On:
  - In half duplex mode, the MAC transmitter applies back pressure for flow control.
  - 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 is received.
  - When Flow Control off:
  - In half duplex mode, the MAC Transmitter does not assert flow control by sending flow control frames or jamming collision.
  - 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 Received counter is not incremented.
- Bit [1]      - 1 - Half Duplex - Only in 10/100 mode  
               - 0 - Full Duplex
- Bit [2]      - 1 - 10Mbps  
               - 0 - 100Mbps
- Bit [4:3]    - 00 – Automatic Enable Auto Neg. - This enables hardware state machine for auto-negotiation.  
               - 01 - Limited Disable auto Neg. This disables hardware for speed auto-negotiation. Hardware Poll MII for link status.  
               - 10 - Link Down. Force link down (disable the port).  
               - 11 - Link Up. The configuration in ECR1[2:0] is used for (speed/half duplex/full duplex/flow control) setup.
- Bit [5]      • Asymmetric Flow Control Enable.  
               - 0 – Disable asymmetric flow control  
               - 01 – 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 processes flow control frames.
- Bit [7:6]    SS - Spanning tree state (802.1D spanning tree protocol) **Default is 11.**  
               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.

### 13.3.2 ECR2Pn: Port N Control Register

I<sup>2</sup>C Address: 01B-033; CPU Address: 0001+2xN (N = port number)

Accessed by CPU and serial interface (R/W)

7	6	5	4	3	2	1	0
Security En	QoS Sel	Reserve	DisL	Ftf	Futf		

- Bit[0]:
- Filter untagged frame (**Default 0**)
    - 0: Disable
    - 1: 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: All tagged frames from this port are discarded or follow security option when security is enable
- Bit[2]:
- Learning Disable (**Default 0**)
    - 1 Learning is disabled on this port
    - 0 Learning is enabled on this port
- Bit[3]:
- Must be '1'
- Bit [5:4:]
- QOS mode selection (**Default 00**)
  - Determines which of the 4 sets of QoS settings is used for 10/100 ports.
  - Note that there are 4 sets of per-queue byte thresholds, and 4 sets of WFQ ratios programmed. These bits select among the 4 choices for each 10/100 port. Refer to QOS Application Note.
    - 00: select class byte limit set 0 and classes WFQ credit set 0
    - 01: select class byte limit set 1 and classes WFQ credit set 1
    - 10: select class byte limit set 2 and classes WFQ credit set 2
    - 11: select class byte limit set 3 and classes WFQ credit set 3
- Bit[7:6]
- Security Enable (**Default 00**). The ZL50416 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). ZL50416 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



## 13.4 (Group 1 Address) VLAN Group

### 13.4.1 AVTCL – VLAN Type Code Register Low

I<sup>2</sup>C Address 036; CPU Address:h100

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

Bit[7:0]: VLANType\_LOW: Lower 8 bits of the VLAN type code **(Default 00)**

### 13.4.2 AVTCH – VLAN Type Code Register High

I<sup>2</sup>C Address 037; CPU Address:h101

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

Bit[7:0]: VLANType\_HIGH: Upper 8 bits of the VLAN type code **(Default is 81)**

### 13.4.3 PVMAP00\_0 – Port 00 Configuration Register 0

I<sup>2</sup>C Address 038, CPU Address:h102

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

In Port Based VLAN Mode

Bit[7:0]: VLAN Mask for ports 7 to 0 **(Default FF)**

This register indicates the legal egress ports. A “1” on bit 7 means that the packet can be sent to port 7. A “0” on bit 7 means that any packet destined to port 7 will be discarded. This register works with registers 1 and 3 to form a 17 bit mask to all egress ports.

In Tag based VLAN Mode

Bit[7:0]: PVID [7:0] **(Default is FF)**

This is the default VLAN tag. It works with configuration register PVMAP00\_1 [7:5] [3:0] to form a default VLAN tag. If the received packet is untagged, then the packet is classified with the default VLAN tag. If the received packet has a VLAN ID of 0, then PVID is used to replace the packet's VLAN ID.

### 13.4.4 PVMAP00\_1 – Port 00 Configuration Register 1

I<sup>2</sup>C Address h53, CPU Address:h103

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

In Port based VLAN Mode

Bit[7:0]: VLAN Mask for ports 15 to 8 **(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 1**)

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**)

### 13.4.5 PVMAP00\_3 – Port 00 Configuration Register 3

I<sup>2</sup>C Address h89, CPU Address:h105)

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

## In Port Based VLAN Mode

7	6	5	3	2	0
FP en	Drop	Default tx priority	VLAN Mask		

Bit [0]: VLAN Mask for Port 24 (CPU port) (**Default 1**).

Bit [2:1]: Reserved (**Default 3**).

Bit [5:3]: Default 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]: Default Discard priority. Used when Bit[7]=1 (**Default 0**)

- 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 Discard Priority are based on VLAN Tag, TOS or Logical Port.
- 1 Transmit Priority and Discard Priority are based on values programmed in bit [6:3]

---

### In Tag-based VLAN Mode

- Bit [0]:
- Not used
- 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 Ingress Filter. Packets with VLAN not belonging to source port are filtered. Asymmetric VLAN.
- Bit [2]: **Force untag out (VLAN tagging is based on 802.1q rule) (Default 1).**
- 0 Disable (Default)
  - 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]: **Default 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]: **Default Discard priority Used when Bit[7]=1 (Default 0)**
- 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 Discard Priority are based on VLAN Tag, TOS or Logical Port.
  - 1 Transmit Priority and Discard Priority are based on values programmed in bit [6:3]

### 13.5 Port Configuration Registers

- PVMAP01\_0,1,3** I<sup>2</sup>C Address h39,54,8A; CPU Address:h106,107,109
- PVMAP02\_0,1,3** I<sup>2</sup>C Address h3A,55,8B; CPU Address:h10A, 10B, 10D
- PVMAP03\_0,1,3** I<sup>2</sup>C Address h3B,56,8C; CPU Address:h10E, 10F, 111
- PVMAP04\_0,1,3** I<sup>2</sup>C Address h3C,57,8D; CPU Address:h112, 113, 115
- PVMAP05\_0,1,3** I<sup>2</sup>C Address h3D,58,8E; CPU Address:h116, 117, 119
- PVMAP06\_0,1,3** I<sup>2</sup>C Address h3E,59,8F; CPU Address:h11A, 11B, 11D
- PVMAP07\_0,1,3** I<sup>2</sup>C Address h3F,5A,90; CPU Address:h11E, 11F, 121
- PVMAP08\_0,1,3** I<sup>2</sup>C Address h40,5B,91; CPU Address:h122, 123, 125
- PVMAP09\_0,1,3** I<sup>2</sup>C Address h41,5C,92; CPU Address:h126, 127, 129
- PVMAP10\_0,1,3** I<sup>2</sup>C Address h42,5D,93; CPU Address:h12A, 12B, 12D
- PVMAP11\_0,1,3** I<sup>2</sup>C Address h43,5E,94; CPU Address:h12E, 12F, 131
- PVMAP12\_0,1,3** I<sup>2</sup>C Address h44,5F,95; CPU Address:h132, 133, 135
- PVMAP13\_0,1,3** I<sup>2</sup>C Address h45,60,96; CPU Address:h136, 137, 139
- PVMAP14\_0,1,3** I<sup>2</sup>C Address h46,61,97; CPU Address:h13A, 13B, 13D
- PVMAP15\_0,1,3** I<sup>2</sup>C Address h47,62,98; CPU Address:h13E, 13F, 141
- PVMAP24\_0,1,3** I<sup>2</sup>C Address h50,6B,A1; CPU Address:h162, 163, 165 (CPU port)

#### 13.5.1 PVMODE

I<sup>2</sup>C Address: h0A4, CPU Address:h170

Accessed by CPU, serial interface (R/W)

7	6	5	4	3	2	1	0
MAC05	MNA	STP	SM0		DF	SL	Vmod

- Bit [0]:
- VLAN Mode (Default = 0)
    - 1 Tag based VLAN Mode
    - 0 Port based VLAN Mode
- Bit [1]:
- Slow learning (Default = 0)
- Same function as SE\_OP MODE bit 7. Either bit can enable the function; both need to be turned off to disable the feature.
- Bit [2]:
- Disable dropping of frames with destination MAC addresses 0180C2000001 to 0180C200000F (Default = 0)
    - 0: Drop all frames in this range
    - 1: Disable dropping of frames in this range
- Bit [3]:
- Reserved
- Bit [4]:
- Support MAC address 0 (Default = 0)
    - 0: MAC address 0 is not learned.
    - 1: MAC address 0 is learned.

- Bit [5]:
  - Disable IEEE multicast control frame (0180C2000000 to 0180C20000FF) to CPU in managed mode (Default = 0)
    - 0: Packet is forwarded to CPU
    - 1: Packet is forwarded as multicast
- Bit [6]:
  - Multiple MAC addresses (Default = 0)
    - 0: Single MAC address is assigned to CPU. Registers MAC0 to MAC5 are used to program the CPU MAC address.
    - 1: One block of 32 MAC addresses are assigned to CPU. The block is defined in an increase way from the MAC address programmed in registers MAC0 to MAC5.
- Bit [7]:
  - Disable registers MAC 5 – 0 (CPU MAC address) in comparison with Ethernet frame destination MAC address. When disable, unicast frames are not forward to CPU. (Default = 0)
    - 1: Disable
    - 0: Enable

### 13.5.2 PVROUTE 0

Registers PVROUTE0 to PVROUTE7 allows the VLAN Index to be assigned an address of a router group. This feature is useful during IP Multicast mode when data is being sent to the VLAN group and no member of the group registers. By assigning a router group, the VLAN group always has a default address to handle the multicast traffic.

CPU Address:h171

Accessed by CPU, serial interface (R/W)

- Bit [0]:
  - VLAN Index 8'hC0 has router group and the router group is VLAN Index 8'h40
- Bit [1]:
  - VLAN Index 8'hC1 has router group and the router group is VLAN Index 8'h41
- Bit [2]:
  - VLAN Index 8'hC2 has router group and the router group is VLAN Index 8'h42
- Bit [3]:
  - VLAN Index 8'hC3 has router group and the router group is VLAN Index 8'h43
- Bit [4]:
  - VLAN Index 8'hC4 has router group and the router group is VLAN Index 8'h44
- Bit [5]:
  - VLAN Index 8'hC5 has router group and the router group is VLAN Index 8'h45
- Bit [6]:
  - VLAN Index 8'hC6 has router group and the router group is VLAN Index 8'h46
- Bit [7]:
  - VLAN Index 8'hC7 has router group and the router group is VLAN Index 8'h47

### 13.5.3 PVROUTE1

CPU Address:h172

Accessed by CPU, serial interface (R/W)

- Bit [0]:
  - VLAN Index 8'hC8 has router group and the router group is VLAN Index 8'h48
- Bit [1]:
  - VLAN Index 8'hC9 has router group and the router group is VLAN Index 8'h48
- Bit [2]:
  - VLAN Index 8'hCA has router group and the router group is VLAN Index 8'h4A
- Bit [3]:
  - VLAN Index 8'hCB has router group and the router group is VLAN Index 8'h4B

- Bit [4]: • VLAN Index 8'hCC has router group and the router group is VLAN Index 8'h4C
- Bit [5]: • VLAN Index 8'hCD has router group and the router group is VLAN Index 8'h4D
- Bit [6]: • VLAN Index 8'hCE has router group and the router group is VLAN Index 8'h4E
- Bit [7]: • VLAN Index 8'hCF has router group and the router group is VLAN Index 8'h4F

#### 13.5.4 PVROUTE2

CPU Address:h173

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hD0 has router group and the router group is VLAN Index 8'h50
- Bit [1]: • VLAN Index 8'hD1 has router group and the router group is VLAN Index 8'h51
- Bit [2]: • VLAN Index 8'hD2 has router group and the router group is VLAN Index 8'h52
- Bit [3]: • VLAN Index 8'hD3 has router group and the router group is VLAN Index 8'h53
- Bit [4]: • VLAN Index 8'hD4 has router group and the router group is VLAN Index 8'h54
- Bit [5]: • VLAN Index 8'hD5 has router group and the router group is VLAN Index 8'h55
- Bit [6]: • VLAN Index 8'hD6 has router group and the router group is VLAN Index 8'h56
- Bit [7]: • VLAN Index 8'hD7 has router group and the router group is VLAN Index 8'h57

#### 13.5.5 PVROUTE3

CPU Address:h174

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hD8 has router group and the router group is VLAN Index 8'h58
- Bit [1]: • VLAN Index 8'hD9 has router group and the router group is VLAN Index 8'h59
- Bit [2]: • VLAN Index 8'hDA has router group and the router group is VLAN Index 8'h5A
- Bit [3]: • VLAN Index 8'hDB has router group and the router group is VLAN Index 8'h5B
- Bit [4]: • VLAN Index 8'hDC has router group and the router group is VLAN Index 8'h5C
- Bit [5]: • VLAN Index 8'hDD has router group and the router group is VLAN Index 8'h5D
- Bit [6]: • VLAN Index 8'hDE has router group and the router group is VLAN Index 8'h5E
- Bit [7]: • VLAN Index 8'hDF has router group and the router group is VLAN Index 8'h5F

#### 13.5.6 PVROUTE4

CPU Address:h175

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hE0 has router group and the router group is VLAN Index 8'h60

- Bit [1]: • VLAN Index 8'hE1 has router group and the router group is VLAN Index 8'h61
- Bit [2]: • VLAN Index 8'hE2 has router group and the router group is VLAN Index 8'h62
- Bit [3]: • VLAN Index 8'hE3 has router group and the router group is VLAN Index 8'h63
- Bit [4]: • VLAN Index 8'hE4 has router group and the router group is VLAN Index 8'h64
- Bit [5]: • VLAN Index 8'hE5 has router group and the router group is VLAN Index 8'h65
- Bit [6]: • VLAN Index 8'hE6 has router group and the router group is VLAN Index 8'h66
- Bit [7]: • VLAN Index 8'hE7 has router group and the router group is VLAN Index 8'h67

### 13.5.7 PVROUTE5

CPU Address:h176

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hE8 has router group and the router group is VLAN Index 8'h68
- Bit [1]: • VLAN Index 8'hE9 has router group and the router group is VLAN Index 8'h69
- Bit [2]: • VLAN Index 8'hEA has router group and the router group is VLAN Index 8'h6A
- Bit [3]: • VLAN Index 8'hEB has router group and the router group is VLAN Index 8'h6B
- Bit [4]: • VLAN Index 8'hEC has router group and the router group is VLAN Index 8'h6C
- Bit [5]: • VLAN Index 8'hED has router group and the router group is VLAN Index 8'h6D
- Bit [6]: • VLAN Index 8'hEE has router group and the router group is VLAN Index 8'h6E
- Bit [7]: • VLAN Index 8'hEF has router group and the router group is VLAN Index 8'h6F

### 13.5.8 PVROUTE6

CPU Address:h177

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hF0 has router group and the router group is VLAN Index 8'h70
- Bit [1]: • VLAN Index 8'hF1 has router group and the router group is VLAN Index 8'h71
- Bit [2]: • VLAN Index 8'hF2 has router group and the router group is VLAN Index 8'h72
- Bit [3]: • VLAN Index 8'hF3 has router group and the router group is VLAN Index 8'h73
- Bit [4]: • VLAN Index 8'hF4 has router group and the router group is VLAN Index 8'h74
- Bit [5]: • VLAN Index 8'hF5 has router group and the router group is VLAN Index 8'h75
- Bit [6]: • VLAN Index 8'hF6 has router group and the router group is VLAN Index 8'h76
- Bit [7]: • VLAN Index 8'hF7 has router group and the router group is VLAN Index 8'h77

**13.5.9 PVROUTE7**

CPU Address:h178

Accessed by CPU, serial interface (R/W)

- Bit [0]: • VLAN Index 8'hF8 has router group and the router group is VLAN Index 8'h78
- Bit [1]: • VLAN Index 8'hF9 has router group and the router group is VLAN Index 8'h79
- Bit [2]: • VLAN Index 8'hFA has router group and the router group is VLAN Index 8'h7A
- Bit [3]: • VLAN Index 8'hFB has router group and the router group is VLAN Index 8'h7B
- Bit [4]: • VLAN Index 8'hFC has router group and the router group is VLAN Index 8'h7C
- Bit [5]: • VLAN Index 8'hFD has router group and the router group is VLAN Index 8'h7D
- Bit [6]: • VLAN Index 8'hFE has router group and the router group is VLAN Index 8'h7E
- Bit [7]: • VLAN Index 8'hFF has router group and the router group is VLAN Index 8'h7F

**13.6 (Group 2 Address) Port Trunking Groups**

Trunk Group 0 - Up to four 10/100 ports can be selected for trunk group 0.

**13.6.1 TRUNK0\_L – Trunk group 0 Low (Managed mode only)**

CPU Address:h200

Accessed by CPU, serial interface (R/W)

Bit [7:0] Port7-0 bit map of trunk 0. (Default 00)

**13.6.2 TRUNK0\_M – Trunk group 0 Medium (Managed mode only)**

CPU Address:h201

Accessed by CPU, serial interface (R/W)

Bit [7:0] Port15-8 bit map of trunk 0. (Default 00)

TRUNK0\_M, and TRUNK0\_L provide a trunk map for trunk0. If ports 0 and 2 are to be trunked together, bit 0 and bit 2 of TRUNK0\_L are set to 1. All others are clear at “0” to indicate that they are not part of trunk 0. Up to 4 ports can be selected for trunk group 0.

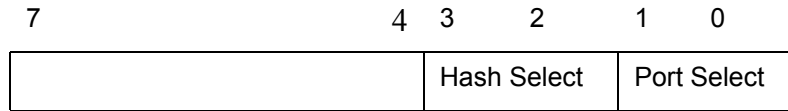
B i t 7	B B i i t t 0 7	B i t 0
TRUNK0_M	TRUNK0_L	
P o r t 15	P P o o r r t t 8 7	P o r t 0



### 13.6.3 TRUNK0\_MODE– Trunk group 0 mode

I<sup>2</sup>C Address h0A5; CPU Address:203

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



- Bit [1:0]:
- Port selection in unmanaged mode. Input pin TRUNK0 enable/disable trunk group 0 in unmanaged mode.
    - 00 Reserved
    - 01 Port 0 and 1 are used for trunk0
    - 10 Port 0,1 and 2 are used for trunk0
    - 11 Port 0,1,2 and 3 are used for trunk0
- Bit [3:2]
- Hash Select. The Hash selected is valid for Trunk 0, 1 and 2. (Default 00)
    - 00 Use Source and Destination Mac Address for hashing
    - 01 Use Source Mac Address for hashing
    - 10 Use Destination Mac Address for hashing
    - 11 Use source destination MAC address and ingress physical port number for hashing

### 13.6.4 TRUNK0\_HASH0 – Trunk group 0 hash result 0 destination port number

CPU Address:h204

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 0 destination port number (Default 00)

### 13.6.5 TRUNK0\_HASH1 – Trunk group 0 hash result 1 destination port number

CPU Address:h205

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 1 destination port number (Default 01)

### 13.6.6 TRUNK0\_HASH2 – Trunk group 0 hash result 2 destination port number

CPU Address:h206

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 2 destination port number (Default 02)

### 13.6.7 TRUNK0\_HASH3 – Trunk group 0 hash result 3 destination port number

CPU Address:h207

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 3 destination port number (Default 03)

**Trunk Group 1** - Up to four 10/100 ports can be selected for trunk group 1.

**13.6.8 TRUNK1\_L – Trunk group 1 Low (Managed mode only)**

Port selection for trunk group 1.

CPU Address:h208

Accessed by CPU, serial interface (R/W)

Bit [7:0] Port7-0 bit map of trunk 1. (Default 00)

**13.6.9 TRUNK1\_M – Trunk group 1 Medium (Managed mode only)**

CPU Address:h209

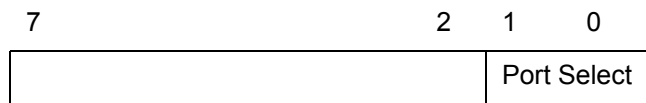
Accessed by CPU, serial interface (R/W)

Bit [7:0] Port15-8 bit map of trunk 1. (Default 00)

**13.6.10 TRUNK1\_MODE – Trunk group 1 mode**

I<sup>2</sup>C Address h0A6; CPU Address:20B

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



- Bit [1:0]:
- Port selection in unmanaged mode. Input pin TRUNK1 enable/disable trunk group 1 in unmanaged mode.
    - 00 Reserved
    - 01 Port 4 and 5 are used for trunk1
    - 10 Reserved
    - 11 Port 4,5,6 and 7 are used for trunk1

**13.6.11 TRUNK1\_HASH0 – Trunk group 1 hash result 0 destination port number**

CPU Address:h20C

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 0 destination port number (Default 04)

**13.6.12 TRUNK1\_HASH1 – Trunk group 1 hash result 1 destination port number**

CPU Address:h20D

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 1 destination port number (Default 05)

**13.6.13 TRUNK1\_HASH2 – Trunk group 1 hash result 2 destination port number**

CPU Address:h20E

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 1 destination port number (Default 06)

**13.6.14 TRUNK1\_HASH3 – Trunk group 1 hash result 3 destination port number**

CPU Address:h20F

Accessed by CPU, serial interface (R/W)

Bit [4:0] Hash result 1 destination port number (Default 07)

**13.6.15 Multicast Hash Registers**

Multicast Hash registers are used to distribute multicast traffic. 16 registers are used to form a 4-entry array; each entry has 27 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.

Hash Value =0	HASH0_3	HASH0_1	HASH0_0
Hash Value =1	HASH1_3	HASH1_1	HASH1_0
Hash Value =2	HASH2_3	HASH2_1	HASH2_0
Hash Value =3	HASH3_3	HASH3_1	HASH3_0
	P	P	P
	o	o	o
	r	r	r
	t	t	t
	24	1	8
	C	5	7
	P		0
	U		

**13.6.15.1 Multicast\_HASH0-0 – Multicast hash result 0 mask byte 0**

CPU Address:h220

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.2 Multicast\_HASH0-1 – Multicast hash result 0 mask byte 1**

CPU Address:h221

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.3 Multicast\_HASH0-3 – Multicast hash result 0 mask byte 3**

CPU Address:h223

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.4 Multicast\_HASH1-0 – Multicast hash result 1 mask byte 0**

CPU Address:h224

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.5 Multicast\_HASH1-1 – Multicast hash result 1 mask byte 1**

CPU Address:h225

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.6 Multicast\_HASH1-3 – Multicast hash result 1 mask byte 3**

CPU Address:h227

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.7 Multicast\_HASH2-0 – Multicast hash result 2 mask byte 0**

CPU Address:h228

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.8 Multicast\_HASH2-1 – Multicast hash result 2 mask byte 1**

CPU Address:h229

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.9 Multicast\_HASH2-3 – Multicast hash result 2 mask byte 3**

CPU Address:h22B

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.10 Multicast\_HASH3-0 – Multicast hash result 3 mask byte 0**

CPU Address:h22C

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.6.15.11 Multicast\_HASH3-1 – Multicast hash result 3 mask byte 1**

CPU Address:h22D

Accessed by CPU, serial interface (R/W)

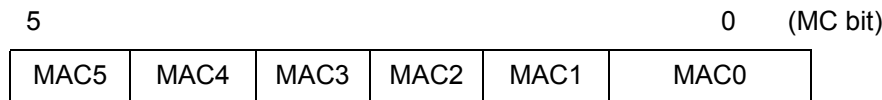
Bit [7:0] (Default FF)

**13.6.15.12 Multicast\_HASH3-3 – Multicast hash result 3 mask byte 3**

CPU Address:h22F

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**13.7 (Group 3 Address) CPU Port Configuration Group**

MAC5 to MAC0 registers form the CPU MAC address. When a packet with destination MAC address match MAC [5:0], the packet is forwarded to the CPU.

**13.7.1 MAC0 – CPU Mac address byte 0**

CPU Address:h300

Accessed by CPU

Bit [7:0] Byte 0 of the CPU MAC address. (Default 00)

**13.7.2 MAC1 – CPU Mac address byte 1**

CPU Address:h301

Accessed by CPU

Bit [7:0] Byte 1 of the CPU MAC address. (Default 00)

**13.7.3 MAC2 – CPU Mac address byte 2**

CPU Address:h302

Accessed by CPU

Bit [7:0] Byte 2 of the CPU MAC address. (Default 00)

**13.7.4 MAC3 – CPU Mac address byte 3**

CPU Address:h303

Accessed by CPU

Bit [7:0] Byte 3 of the CPU MAC address. (Default 00)

**13.7.5 MAC4 – CPU Mac address byte 4**

CPU Address:h304

Accessed by CPU

Bit [7:0] Byte 4 of the CPU MAC address. (Default 00)

**13.7.6 MAC5 – CPU Mac address byte 5**

CPU Address:h305

Accessed by CPU

Bit [7:0] Byte 5 of the CPU MAC address. (Default 00).

**13.7.7 INT\_MASK0 – Interrupt Mask 0**

CPU Address:h306

Accessed by CPU, serial interface (R/W)

The CPU can dynamically mask the interrupt when it is busy and doesn't want to be interrupted. **(Default 0xFF)**

Bit [7:0] MASK

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

Bit [0]: • CPU frame interrupt. CPU frame buffer has data for CPU to read

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

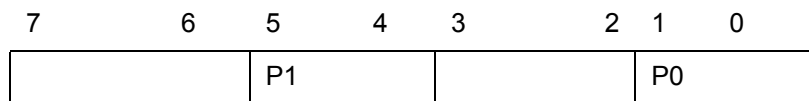
Bit [2]: • Control Command 2 interrupt. Control command Frame buffer2 has data for CPU to read

Bit [7:3]: • Reserved

**13.7.8 INTP\_MASK0 – Interrupt Mask for MAC Port 0,1**

CPU Address:h310

Accessed by CPU, serial interface (R/W)

The CPU can dynamically mask the interrupt when it is busy and doesn't want to be interrupted **(Default 0xFF)**

- 1: Mask the interrupt
- 0: Unmask the interrupt

Bit [0]: Port 0 statistic counter wrap around interrupt mask. An Interrupt is generated when a statistic counter wraps around. Refer to hardware statistic counter for interrupt sources.

Bit [1]: Port 0 link change mask

Bit [4]: Port 1 statistic counter wrap around interrupt mask. Refer to hardware statistic counter for interrupt sources.

Bit [5]: Port 1 link change mask

**13.7.9 INTP\_MASK1 – Interrupt Mask for MAC Port 2,3**

CPU Address:h311

Accessed by CPU, serial interface (R/W)

**13.7.10 INTP\_MASK2 – Interrupt Mask for MAC Port 4,5**

CPU Address:h312

Accessed by CPU, serial interface (R/W)

**13.7.11 INTP\_MASK3 – Interrupt Mask for MAC Port 6,7**

CPU Address:h313

Accessed by CPU, serial interface (R/W)

**13.7.12 INTP\_MASK4 – Interrupt Mask for MAC Port 8,9**

CPU Address:h314

Accessed by CPU, serial interface (R/W)

**13.7.13 INTP\_MASK5 – Interrupt Mask for MAC Port 10,11**

CPU Address:h315

Accessed by CPU, serial interface (R/W)

**13.7.14 INTP\_MASK6 – Interrupt Mask for MAC Port 12,13**

CPU Address:h316

Accessed by CPU, serial interface (R/W)

**13.7.15 INTP\_MASK7 – Interrupt Mask for MAC Port 14,15**

CPU Address:h317

Accessed by CPU, serial interface (R/W)

**13.7.16 RQS – Receive Queue Select CPU Address:h323)**

Accessed by CPU, serial interface (RW)

Select which receive queue is used.

7	6	5	4	3	2	1	0
FQ3	FQ2	FQ1	FQ0	SQ3	SQ2	SQ1	SQ0

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

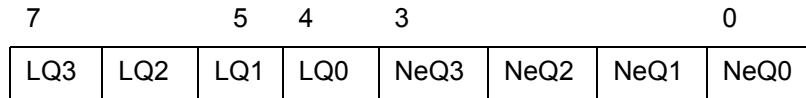
Bit[7]: Enable flush Queue 3

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].

### 13.7.17 RQSS – Receive Queue Status

CPU Address:h324

Accessed by CPU, serial interface (RO)



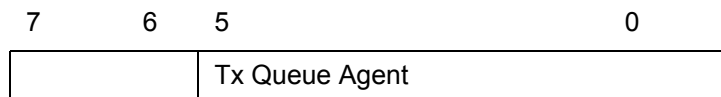
CPU receive queue status

- Bit[3:0]: Queue 3 to 0 not empty
- Bit[4]: Head of line entry for Queue 0 is valid for too long. CPU Queue 0 has no WRED threshold.
- Bit[7:5]: Head of line entry for Queue 3 to 1 is valid for too long or Queue length is longer than WRED threshold.

### 13.7.18 TX\_AGE – Tx Queue Aging timer

I<sup>2</sup>C Address: h07;CPU Address:h324

Accessed by CPU, serial interface (RW)



- Bit[5:0]: Unit of 100ms (**Default 8**)

Disable transmission queue aging if value is zero. Aging timer for all ports and queues.

This register must be set to 0 for 'No Packet Loss Flow Control Test'.

## 13.8 (Group 4 Address) Search Engine Group

### 13.8.1 AGETIME\_LOW – MAC address aging time Low

I<sup>2</sup>C Address h0A8; CPU Address:h400

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

The ZL50416 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 TSTOUT9.

Bit [7:0] Low byte of the MAC address aging timer.



### 13.8.2 AGETIME\_HIGH –MAC address aging time High

I<sup>2</sup>C Address h0A9; CPU Address h401

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

Bit [7:0]: High byte of the MAC address aging timer.

The default setting provide 300 seconds aging time. Aging time is based on the following equation:

{AGETIME\_TIME,AGETIME\_LOW} X (# of MAC entries in the memory X100μsec). Number of MAC entries = 32K when 1 MB is used. Number of entries = 64K when 2 MB is used.

### 13.8.3 V\_AGETIME – VLAN to Port aging time

CPU Address h402

Accessed by CPU (R/W)

Bit [7:0] (Default FF) V\_AGETIME X 256 X 100msec is the age time for the VLAN. This timer is for controlling how long a port is associated to a particular VLAN. It can use dynamic shrinking of a VLAN domain if no packet arrives for the VLAN. The ZL50416 does not remove the port from the VLAN domain. It sends an Age VLAN Port Control Command to the CPU. The CPU has to remove the port.

### 13.8.4 SE\_OPMODE – Search Engine Operation Mode

CPU Address:h403

Accessed by CPU (R/W)

**Note:** ECR2[2] enable/disable learning for each port.

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

### 13.8.5 SCAN – SCAN Control Register (default 00)

CPU Address h404

Accessed by CPU (R/W)

7	6	0
R	Ratio	

SCAN is used when fast learning is enabled (SE\_OPMODE 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 rounds are used for aging. Never scan for new MAC addresses.

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

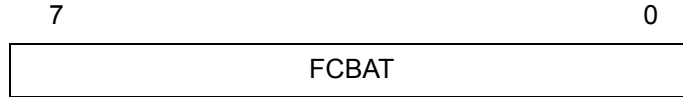
R= 1, Ratio = 7: In eight rounds, one is used for scanning and seven are used for aging

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

### 13.9 (Group 5 Address) Buffer Control/QOS Group

#### 13.9.1 FCBAT – FCB Aging Timer

I<sup>2</sup>C Address h0AA; CPU Address:h500

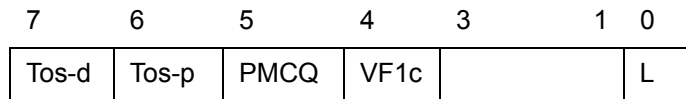


- Bit [7:0]:
- FCB Aging time. Unit of 1ms. (Default FF)
  - This is for buffer aging control. It is used to configure the buffer aging time. This function can be enabled/disabled through bootstrap pin. It is not suggested to use this function for normal operation.

#### 13.9.2 QOSC – QOS Control

I<sup>2</sup>C Address h0AB; CPU Address:h501

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

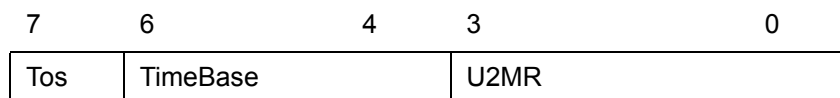


- Bit [0]:
- QoS frame lost is OK. Priority will be available for flow control enabled source only when this bit is set (Default 0)
- Bit [4]:
- Per VLAN Multicast Flow Control (Default 0)
    - 0 – Disable
    - 1 – Enable
- Bit [5]:
- Select processor multicast queue size
    - 0 = 16 entries
    - 1 = 64 entries
- Bit [6]:
- Select TOS bits for Priority (Default 0)
    - 0 – Use TOS [4:2] bits to map the transmit priority
    - 1 – Use TOS [7:5] bits to map the transmit priority
- Bit [7]:
- Select TOS bits for Drop priority(Default 0)
    - 0 – Use TOS [4:2] bits to map the drop priority
    - 1 – Use TOS [7:5] bits to map the drop priority

#### 13.9.3 FCR – Flooding Control Register

I<sup>2</sup>C Address h0AC; CPU Address:h502

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

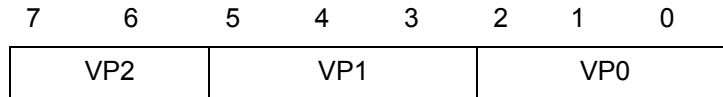


- Bit [3:0]: • U2MR: Unicast to Multicast Rate. Units in terms of time base defined in bits [6:4]. This is used to limit the amount of flooding traffic. The value in U2MR specifies how many packets are allowed to flood within the time specified by bit [6:4]. To disable this function, program U2MR to 0. (Default = 8)
- Bit [6:4]: Time Base: (Default = 000)  
**000 = 100us**  
**001 = 200us**  
**010 = 400us**  
**011 = 800us**  
**100 = 1.6ms**  
**101 = 3.2ms**  
**110 = 6.4ms**  
**111 = 100us, same as 000.**
- Bit [7]: Select VLAN tag or TOS (IP packets) to be preferentially picked to map transmit priority and drop priority (**Default = 0**).  
 0 – Select VLAN Tag priority field over TOS  
 1 – Select TOS over VLAN tag priority field

### 13.9.4 AVPML – VLAN Tag Priority Map

I<sup>2</sup>C Address h0AD; CPU Address:h503

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



Registers AVPML, AVPMM, and AVPMH allow the eight VLAN Tag 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 0 into Internal transmit priority 7. The new priority is used inside the ZL50416. When the packet goes out it carries the original priority.

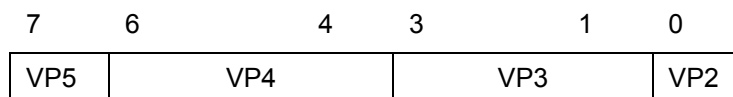
- Bit [2:0]: Priority when the VLAN tag priority field is 0 (**Default 0**)
- Bit [5:3]: Priority when the VLAN tag priority field is 1 (**Default 0**)
- Bit [7:6]: Priority when the VLAN tag priority field is 2 (**Default 0**)

### 13.9.5 AVPMM – VLAN Priority Map

I<sup>2</sup>C Address h0AE, CPU Address:h504

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

Map VLAN priority into eight level transmit priorities:

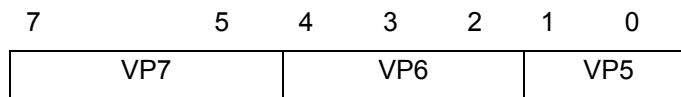


- Bit [0]: Priority when the VLAN tag priority field is 2 **(Default 0)**  
 Bit [3:1]: Priority when the VLAN tag priority field is 3 **(Default 0)**  
 Bit [6:4]: Priority when the VLAN tag priority field is 4 **(Default 0)**  
 Bit [7]: Priority when the VLAN tag priority field is 5 **(Default 0)**

### 13.9.6 AVPMH – VLAN Priority Map

I<sup>2</sup>C Address h0AF, CPU Address:h505

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



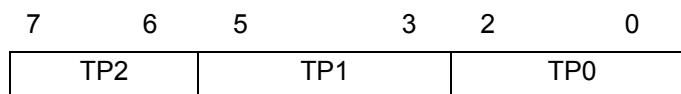
Map VLAN priority into eight level transmit priorities:

- Bit [1:0]: Priority when the VLAN tag priority field is 5 **(Default 0)**  
 Bit [4:2]: Priority when the VLAN tag priority field is 6 **(Default 0)**  
 Bit [7:5]: Priority when the VLAN tag priority field is 7 **(Default 0)**

### 13.9.7 TOSPML – TOS Priority Map

I<sup>2</sup>C Address h0B0, CPU Address:h506

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



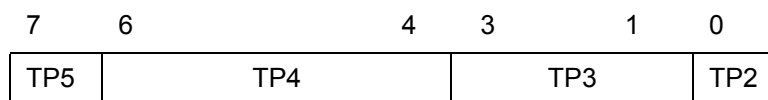
Map TOS field in IP packet into eight level transmit priorities

- Bit [2:0]: Priority when the TOS field is 0 **(Default 0)**  
 Bit [5:3]: Priority when the TOS field is 1 **(Default 0)**  
 Bit [7:6]: Priority when the TOS field is 2 **(Default 0)**

### 13.9.8 TOSPMM – TOS Priority Map

I<sup>2</sup>C Address h0B1, CPU Address:h507

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



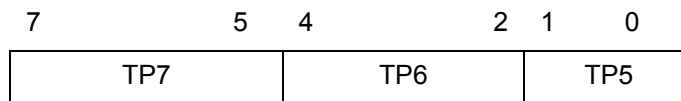
Map TOS field in IP packet into eight level transmit priorities

- Bit [0]: Priority when the TOS field is 2 (**Default 0**)
- Bit [3:1]: Priority when the TOS field is 3 (**Default 0**)
- Bit [6:4]: Priority when the TOS field is 4 (**Default 0**)
- Bit [7]: Priority when the TOS field is 5 (**Default 0**)

### 13.9.9 TOSPMH – TOS Priority Map

I<sup>2</sup>C Address h0B2, CPU Address:h508

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



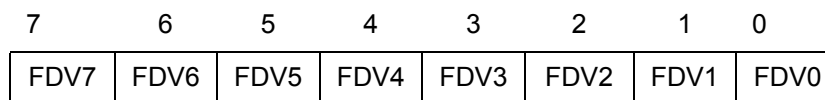
Map TOS field in IP packet into eight level transmit priorities:

- Bit [1:0]: Priority when the TOS field is 5 (**Default 0**)
- Bit [4:2]: Priority when the TOS field is 6 (**Default 0**)
- Bit [7:5]: Priority when the TOS field is 7 (**Default 0**)

### 13.9.10 AVDM – VLAN Discard Map

I<sup>2</sup>C Address h0B3, CPU Address:h509

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



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

- Bit [0]: Frame drop priority when VLAN Tag priority field is 0 (**Default 0**)
- Bit [1]: Frame drop priority when VLAN Tag priority field is 1 (**Default 0**)
- Bit [2]: Frame drop priority when VLAN Tag priority field is 2 (**Default 0**)
- Bit [3]: Frame drop priority when VLAN Tag priority field is 3 (**Default 0**)
- Bit [4]: Frame drop priority when VLAN Tag priority field is 4 (**Default 0**)
- Bit [5]: Frame drop priority when VLAN Tag priority field is 5 (**Default 0**)
- Bit [6]: Frame drop priority when VLAN Tag priority field is 6 (**Default 0**)
- Bit [7]: Frame drop priority when VLAN Tag priority field is 7 (**Default 0**)

### 13.9.11 TOSDML – TOS Discard Map

I<sup>2</sup>C Address h0B4, CPU Address:h50A

Accessed by CPU, serial interface and I<sup>2</sup>C (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 drop priority when TOS field is 0 (**Default 0**)
- Bit [1]: Frame drop priority when TOS field is 1 (**Default 0**)
- Bit [2]: Frame drop priority when TOS field is 2 (**Default 0**)
- Bit [3]: Frame drop priority when TOS field is 3 (**Default 0**)
- Bit [4]: Frame drop priority when TOS field is 4 (**Default 0**)
- Bit [5]: Frame drop priority when TOS field is 5 (**Default 0**)
- Bit [6]: Frame drop priority when TOS field is 6 (**Default 0**)
- Bit [7]: Frame drop priority when TOS field is 7 (**Default 0**)

### 13.9.12 BMRC - Broadcast/Multicast Rate Control

I<sup>2</sup>C Address h0B5, CPU Address:h50B)

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	4	3	0
Broadcast Rate		Multicast Rate	

This broadcast and multicast rate defines for each port, the number of packets 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. Time base is based on register FCR [6:4]

- 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**)

### 13.9.13 UCC – Unicast Congestion Control

I<sup>2</sup>C Address h0B6, CPU Address: 50C

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

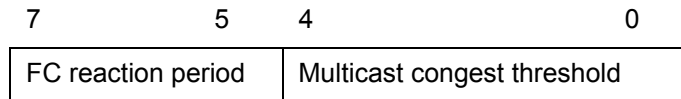
7	0
Unicast congest threshold	

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 1 frame. (Default: h10 for 2 MB or h08 for 1 MB)

### 13.9.14 MCC – Multicast Congestion Control

I<sup>2</sup>C Address h0B7, CPU Address: 50D

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



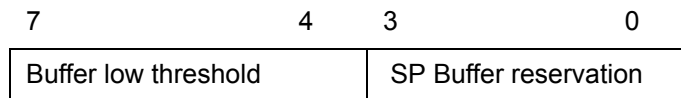
Bit [4:0]: In multiples of two frames (granularity). Used for triggering MC flow control when destination port's multicast best effort queue reaches MCC threshold.(Default 0x10)

Bit [7:5]: Flow control reaction period (Default 2) Granularity 4uSec.

### 13.9.15 PR100 – Port Reservation for 10/100 ports

I<sup>2</sup>C Address h0B8, CPU Address 50E

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



Bit [3:0]: Per source port buffer reservation. Define the space in the FDB reserved for each 10/100 port and CPU. Expressed in multiples of 4 packets. For each packet 1536 bytes are reserved in the memory.

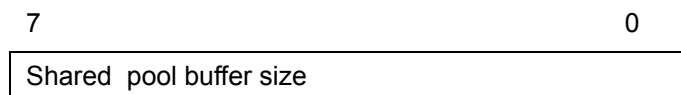
Bits [7:4]: Expressed in multiples of 4 packets. Threshold for dropping all best effort frames when destination port best efforts queues reaches UCC threshold, shared pool is all used and source port reservation is at or below the PR100[7:4] level. Also the threshold for initiating UC flow control.

- Default:
  - h58 for configuration with 2MB;
  - h35 for configuration with 1MB;

### 13.9.16 SFCB – Share FCB Size

I<sup>2</sup>C Address h0BA), CPU Address 510

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



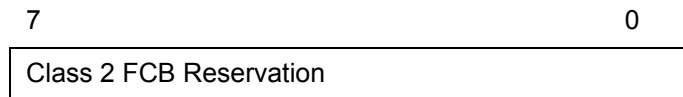


- Bits [7:0]: Expressed in multiples of 4 packets. Buffer reservation for shared pool.
- Default:
    - hE6 for configuration with memory of 2MB;
    - h46 for configuration with memory of 1MB;

### 13.9.17 C2RS – Class 2 Reserve Size

I<sup>2</sup>C Address h0BB, CPU Address 511

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

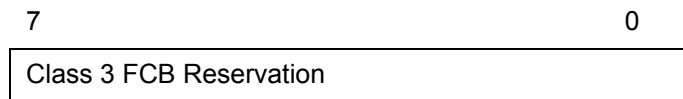


Buffer reservation for class 2 (third lowest priority). Granularity 1. **(Default 0)**

### 13.9.18 C3RS – Class 3 Reserve Size

I<sup>2</sup>C Address h0BC, CPU Address 512

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

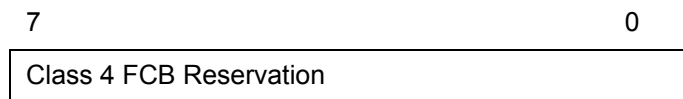


Buffer reservation for class 3. Granularity 1. **(Default 0)**

### 13.9.19 C4RS – Class 4 Reserve Size

I<sup>2</sup>C Address h0BD, CPU Address 513

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

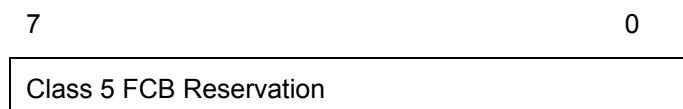


Buffer reservation for class 4. Granularity 1. **(Default 0)**

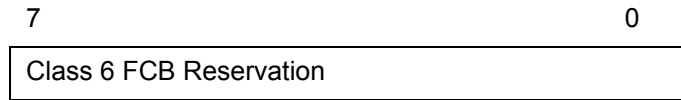
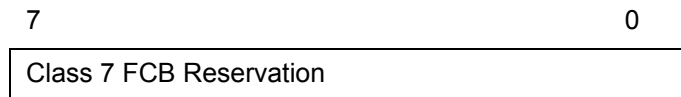
### 13.9.20 C5RS – Class 5 Reserve Size

I<sup>2</sup>C Address h0BE; CPU Address 514

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



Buffer reservation for class 5. Granularity 1. **(Default 0)**

**13.9.21 C6RS – Class 6 Reserve Size**I<sup>2</sup>C Address h0BF; CPU Address 515Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)Buffer reservation for class 6 (second highest priority). Granularity 1. **(Default 0)****13.9.22 C7RS – Class 7 Reserve Size**I<sup>2</sup>C Address h0C0; CPU Address 516Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)Buffer reservation for class 7 (highest priority). Granularity 1. **(Default 0)****13.9.23 QOSCn - Classes Byte Limit Set 0**Accessed by CPU; serial interface and I<sup>2</sup>C (R/W):

- C — QOSC00 – BYTE\_C01 (I<sup>2</sup>C Address h0C1, CPU Address 517)
- B — QOSC01 – BYTE\_C02 (I<sup>2</sup>C Address h0C2, CPU Address 518)
- A — QOSC02 – BYTE\_C03 (I<sup>2</sup>C Address h0C3, CPU Address 519)

QOSC00 through QOSC02 represents one set of values A-C for a 10/100 port when using the Weighted Random Early Drop (WRED) Scheme described in Chapter 7. There are four such sets of values A-C specified in Classes Byte Limit Set 0, 1, 2, and 3. For CPU port A-C values are defined using register CPUQOSC1, 2 and 3.

Each 10/ 100 port can choose one of the four Byte Limit Sets as specified by the QoS Select field located in bits 5 to 4 of the ECR2n register. The values A-C are per-queue byte thresholds for random early drop. QOSC02 represents A, and QOSC00 represents C.

Granularity when Delay bound is used: QOSC02: 128 bytes, QOSC01: 256 bytes, QOSC00: 512 bytes. Granularity when WFQ is used: QOSC02: 512 bytes, QOSC01: 512 bytes, QOSC00: 512 bytes.

**13.9.24 Classes Byte Limit Set 1**Accessed by CPU, serial interface and I<sup>2</sup>C (R/W):

- C - QOSC03 – BYTE\_C11 (I<sup>2</sup>C Address h0C4, CPU Address 51a)
- B - QOSC04 – BYTE\_C12 (I<sup>2</sup>C Address h0C5, CPU Address 51b)
- A - QOSC05 – BYTE\_C13 (I<sup>2</sup>C Address h0C6, CPU Address 51c)

QOSC03 through QOSC05 represents one set of values A-C for a 10/100 port when using the Weighted Random Early Drop (WRED) scheme.

Granularity when Delay bound is used: QOSC05: 128 bytes, QOSC04: 256 bytes, QOSC03: 512 bytes. Granularity when WFQ is used: QOSC05: 512 bytes, QOSC04: 512 bytes, QOSC03: 512 bytes.

### 13.9.25 Classes Byte Limit Set 2

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

- C - QOSC06 – BYTE\_C21 (CPU Address 51d)
- B - QOSC07 – BYTE\_C22 (CPU Address 51e)
- A - QOSC08 – BYTE\_C23 (CPU Address 51f)

QOSC06 through QOSC08 represents one set of values A-C for a 10/100 port when using the Weighted Random Early Drop (WRED) scheme.

Granularity when Delay bound is used: QOSC08: 128 bytes, QOSC07: 256 bytes, QOSC06: 512 bytes.

Granularity when WFQ is used: QOSC08: 512 bytes, QOSC07: 512 bytes, QOSC06: 512 bytes

### 13.9.26 Classes Byte Limit Set 3

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

- C - QOSC09 – BYTE\_C31 (CPU Address 520)
- B - QOSC10 – BYTE\_C32 (CPU Address 521)
- A - QOSC11 – BYTE\_C33 (CPU Address 522)

QOSC09 through QOSC011 represents one set of values A-C for a 10/100 port when using the Weighted Random Early Drop (WRED) scheme.

Granularity when Delay bound is used: QOSC11: 128 bytes, QOSC10: 256 bytes, QOSC09: 512 bytes.

Granularity when WFQ is used: QOSC11: 512 bytes, QOSC10: 512 bytes, QOSC09: 512 bytes

### 13.9.27 Classes WFQ Credit Set 0

Accessed by CPU and serial interface

- W3 - QOSC24[5:0] – CREDIT\_C00 (CPU Address 52f)
- W2 - QOSC25[5:0] – CREDIT\_C01 (CPU Address 530)
- W1 - QOSC26[5:0] – CREDIT\_C02 (CPU Address 531)
- W0 - QOSC27[5:0] – CREDIT\_C03 (CPU Address 532)

QOSC24 through QOSC27 represents one set of WFQ parameters for a 10/100 port. There are four such sets of values. The granularity of the numbers is 1, and their sum must be 64. QOSC27 corresponds to W0, and QOSC24 corresponds to W3.

- QOSC24[7:6]: Priority service type for the ports select this parameter set. Option 1 to option 4.
- QOSC25[7]: Priority service allow flow control for the ports select this parameter set.
- QOSC25[6]: Flow control pause best effort traffic only

Both flow control allow and flow control best effort only can take effect only the priority type is WFQ.

### 13.9.28 Classes WFQ Credit Set 1

Accessed by CPU and serial interface

- W3 - QOSC28[5:0] – CREDIT\_C10 (CPU Address 533)
- W2 - QOSC29[5:0] – CREDIT\_C11 (CPU Address 534)
- W1 - QOSC30[5:0] – CREDIT\_C12 (CPU Address 535)
- W0 - QOSC31[5:0] – CREDIT\_C13 (CPU Address 536)

QOSC28 through QOSC31 represents one set of WFQ parameters for a 10/100 port. There are four such sets of values. The granularity of the numbers is 1, and their sum must be 64. QOSC31 corresponds to W0, and QOSC28 corresponds to W3.

- QOSC28[7:6]: Priority service type for the ports select this parameter set. Option 1 to option 4.
- QOSC29[7]: Priority service allow flow control for the ports select this parameter set.
- QOSC29[6]: Flow control pause best effort traffic only

### 13.9.29 Classes WFQ Credit Set 2

Accessed by CPU and serial interface

- W3 - QOSC32[5:0] – CREDIT\_C20 (CPU Address 537)
- W2 - QOSC33[5:0] – CREDIT\_C21 (CPU Address 538)
- W1 - QOSC34[5:0] – CREDIT\_C22 (CPU Address 539)
- W0 - QOSC35[5:0] – CREDIT\_C23 (CPU Address 53a)

QOSC35 through QOSC32 represents one set of WFQ parameters for a 10/100 port. There are four such sets of values. The granularity of the numbers is 1, and their sum must be 64. QOSC35 corresponds to W0, and QOSC32 corresponds to W3.

- QOSC32[7:6]: Priority service type for the ports select this parameter set. Option 1 to option 4.
- QOSC33[7]: Priority service allow flow control for the ports select this parameter set.
- QOSC33[6]: Flow control pause for best effort traffic only

### 13.9.30 Classes WFQ Credit Set 3

Accessed by CPU and serial interface

- W3 - QOSC36[5:0] – CREDIT\_C30 (CPU Address 53b)
- W2 - QOSC37[5:0] – CREDIT\_C31 (CPU Address 53c)
- W1 - QOSC38[5:0] – CREDIT\_C32 (CPU Address 53d)
- W0 - QOSC39[5:0] – CREDIT\_C33 (CPU Address 53e)

QOSC39 through QOSC36 represents one set of WFQ parameters for a 10/100 port. There are four such sets of values. The granularity of the numbers is 1, and their sum must be 64. QOSC39 corresponds to W0, and QOSC36 corresponds to W3.

- QOSC36[7:6]: Priority service type for the ports select this parameter set. Option 1 to option 4.
- QOSC37[7]: Priority service allow flow control for the ports select this parameter set.
- QOSC37[6]: Flow control pause best effort traffic only

### 13.9.31 RDRC0 – WRED Rate Control 0

I<sup>2</sup>C Address 0FB, CPU Address 553

Accessed by CPU, Serial Interface and I<sup>2</sup>C (R/W)

7	4	3	0
X Rate		Y Rate	

Bits [7:4]: Corresponds to the frame drop percentage X% for WRED. Granularity 6.25%.

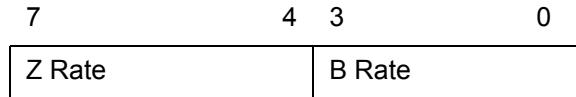
Bits[3:0]: Corresponds to the frame drop percentage Y% for WRED. Granularity 6.25%.

See Programming QoS Registers application note for more information

### 13.9.32 RDRC1 – WRED Rate Control 1

I<sup>2</sup>C Address 0FC, CPU Address 554

Accessed by CPU, Serial Interface and I<sup>2</sup>C (R/W)



Bits [7:4]: Corresponds to the frame drop percentage Z% for WRED. Granularity 6.25%.

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. Granularity 6.25%.

See Programming QoS Registers application note for more information

### 13.9.33 User Defined Logical Ports and Well Known Ports

The ZL50416 supports classifying packet priority through layer 4 logical port information. It can be setup by 8 Well Known Ports, 8 User Defined Logical Ports, and 1 User Defined Range.

The 8 Well Known Ports supported are:

- 0:23
- 1:512
- 2:6000
- 3:443
- 4:111
- 5:22555
- 6:22
- 7:554

Their respective priority can be programmed via Well\_Known\_Port [7:0] priority register. Well\_Known\_Port\_Enable can individually turn on/off each Well Known Port if desired.

Similarly, the User Defined Logical Port provides the user programmability to the priority, plus the flexibility to select specific logical ports to fit the applications. The 8 User Logical Ports can be programmed via User\_Port 0-7 registers. Two registers are required to be programmed for the logical port number. The respective priority can be programmed to the User\_Port [7:0] priority register. The port priority can be individually enabled/disabled via User\_Port\_Enable register.

The User Defined Range provides a range of logical port numbers with the same priority level. Programming is similar to the User Defined Logical Port. Instead of programming a fixed port number, an upper and lower limit need to be programmed, they are: {RHIGHH, RHIGHL} and {RLOWH, RLOWL} respectively. If the value in the upper limit is smaller or equal to the lower limit, the function is disabled. Any IP packet with a logical port that is less than the upper limit and more than the lower limit will use the priority specified in RPRIORITY.

**13.9.33.1 USER\_PORT0\_(0~7) – User Define Logical Port (0~7)**

USER\_PORT\_0 - I<sup>2</sup>C Address h0D6 + 0DE; CPU Address 580(Low) + 581(high)

USER\_PORT\_1 - I<sup>2</sup>C Address h0D7 + 0DF; CPU Address 582 + 583

USER\_PORT\_2 - I<sup>2</sup>C Address h0D8 + 0E0; CPU Address 584 + 585

USER\_PORT\_3 - I<sup>2</sup>C Address h0D9 + 0E1; CPU Address 586 + 587

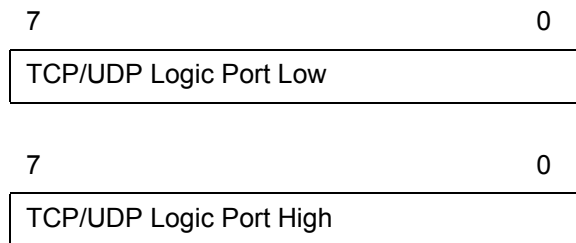
USER\_PORT\_4 - I<sup>2</sup>C Address h0DA + 0E2; CPU Address 588 + 589

USER\_PORT\_5 - I<sup>2</sup>C Address h0DB + 0E3; CPU Address 58A + 58B

USER\_PORT\_6 - I<sup>2</sup>C Address h0DC + 0E4; CPU Address 58C + 58D

USER\_PORT\_7 - I<sup>2</sup>C Address h0DD + 0E5; CPU Address 58E + 58F

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

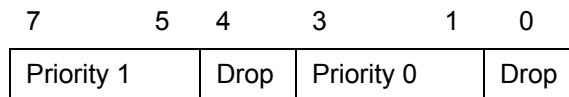


(Default 00) This register is duplicated eight times from PORT 0 through PORT 7 and allows the CPU to define eight separate ports.

**13.9.33.2 USER\_PORT\_[1:0]\_PRIORITY - User Define Logic Port 1 and 0 Priority**

I<sup>2</sup>C Address h0E6, CPU Address 590

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



The chip allows the CPU to define the priority

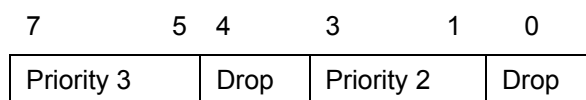
Bits[3:0]: Priority setting, transmission + dropping, for logic port 0

Bits[7:4]: Priority setting, transmission + dropping, for logic port 1 (Default 00)

**13.9.33.3 USER\_PORT\_[3:2]\_PRIORITY - User Define Logic Port 3 and 2 Priority**

I<sup>2</sup>C Address h0E7, CPU Address 591

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



**13.9.33.4 USER\_PORT\_[5:4]\_PRIORITY - User Define Logic Port 5 and 4 Priority**I<sup>2</sup>C Address h0E8, CPU Address 592Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 5	Drop	Priority 4	Drop		

(Default 00)

**13.9.33.5 USER\_PORT\_[7:6]\_PRIORITY - User Define Logic Port 7 and 6 Priority**I<sup>2</sup>C Address h0E9, CPU Address 593Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 7	Drop	Priority 6	Drop		

(Default 00)

**13.9.33.6 USER\_PORT\_ENABLE[7:0] – User Define Logic 7 to 0 Port Enables**I<sup>2</sup>C Address h0EA, CPU Address 594Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

(Default 00)

**13.9.33.7 WELL\_KNOWN\_PORT[1:0] PRIORITY- Well Known Logic Port 1 and 0 Priority**I<sup>2</sup>C Address h0EB, CPU Address 595Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 1	Drop	Priority 0	Drop		

Priority 0 - Well known port 23 for telnet applications.

Priority 1 - Well Known port 512 for TCP/UDP.

(Default 00)

**13.9.33.8 WELL\_KNOWN\_PORT[3:2] PRIORITY- Well Known Logic Port 3 and 2 Priority**I<sup>2</sup>C Address h0EC, CPU Address 596Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 3	Drop	Priority 2	Drop		

Priority 2 - Well known port 6000 for XWIN.

Priority 3 - Well known port 443 for http.sec

(Default 00)

**13.9.33.9 WELL\_KNOWN\_PORT [5:4] PRIORITY- Well Known Logic Port 5 and 4 Priority**I<sup>2</sup>C Address h0ED, CPU Address 597Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 5	Drop	Priority 4	Drop		

Priority 4 - Well Known port 111 for sun remote procedure call.

Priority 5 - Well Known port 22555 for IP Phone call setup.

(Default 00)

**13.9.33.10 WELL\_KNOWN\_PORT [7:6] PRIORITY- Well Known Logic Port 7 and 6 Priority**I<sup>2</sup>C Address h0EE, CPU Address 598Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	1	0
Priority 7	Drop	Priority 6	Drop		

Priority 6 - well know port 22 for ssh.

Priority 7 – well Known port 554 for rtsp.

(Default 00)

**13.9.33.11 WELL\_KNOWN\_PORT\_ENABLE [7:0] – Well Known Logic 7 to 0 Port Enables**I<sup>2</sup>C Address h0EF, CPU Address 599Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0



- 1 – Enable
- 0 - Disable

(Default 00)

### 13.9.33.12 RLOWL – User Define Range Low Bit 7:0

I<sup>2</sup>C Address h0F4, CPU Address: 59a

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

(Default 00)

### 13.9.33.13 RLOWH – User Define Range Low Bit 15:8

I<sup>2</sup>C Address h0F5, CPU Address: 59b

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

(Default 00)

### 13.9.33.14 RHIGHL – User Define Range High Bit 7:0

I<sup>2</sup>C Address h0D3, CPU Address: 59c

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

(Default 00)

### 13.9.33.15 RHIGHH – User Define Range High Bit 15:8

I<sup>2</sup>C Address h0D4, CPU Address: 59d

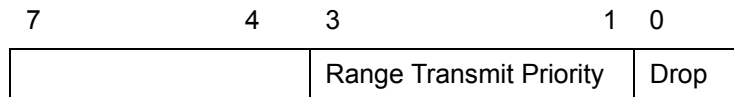
Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

(Default 00)

### 13.9.33.16 RRIORITY – User Define Range Priority

I<sup>2</sup>C Address h0D5, CPU Address: 59e

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)



RLOW and RHIGH form a range for logical ports to be classified with priority specified in RRIORITY.

Bit[3:1]      Transmit Priority

Bits[0]:      Drop Priority

### 13.9.34 CPUQOSC123

CPU Address: 5a0, 5a1, 5a2

Accessed by CPU and serial interface (R/W)

- C - CPUQOSC1 – CPU BYTE\_C1 I<sup>2</sup>C Address h0C1, CPU Address 517)
- B - CPUQOSC2 – CPU BYTE\_C2 I<sup>2</sup>C Address h0C2, CPU Address 518)
- A - CPUQOSC3 – CPU BYTE\_C3 I<sup>2</sup>C Address h0C3, CPU Address 519)

Represents values A-C for a CPU port. The values A-C are per-queue byte thresholds for random early drop.

QOSC3 represents A, and QOSC1 represents C. Granularity: 256 bytes

### 13.10 (Group 6 Address) MISC Group

#### 13.10.1 MII\_OP0 – MII Register Option 0

I<sup>2</sup>C Address F0, CPU Address:h600

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	6	5	4	0
hfc	1prst	DisJ	Vendor Spc. Reg Addr	

Bits [7]: Half duplex flow control feature  
 0 = Half duplex flow control always enable  
 1 = Half duplex flow control by negotiation

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

Bits[5]: Disable jabber detection. This is for HomePNA applications or any serial operation slower than 10Mbps.  
 0 = Enable  
 1 = Disable

Bit[4:0]: Vendor specified link status register address (null value means don't use it) (Default 00). This is used if the Linkup bit position in the PHY is non-standard

#### 13.10.2 MII\_OP1 – MII Register Option 1

I<sup>2</sup>C Address F1, CPU Address:h601

Accessed by CPU, serial interface and I<sup>2</sup>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)

### 13.10.3 FEN – Feature Register

I<sup>2</sup>C Address F2, CPU Address:h602)

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	6	5	4	3	2	1	0
DML	Mii	Rp	IP Mul	V-Sp	DS	RC	SC

- Bits [0]: **Statistic Counter Enable (Default 0)**
- 0 – Disable
  - 1 – Enable (all ports)
- 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]: **Rate Control Enable (Default 0)**
- 0 – Disable
  - 1 – Enable
- This bit enables/disables the rate control for all 10/100 ports. To start rate control in a 10/100 port the rate control memory must be programmed. This feature requires an external CPU. See Programming QoS Registers application note and Processor Interface application note for more information.
- 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 registers ECR1Pn are NOT used to program the port spanning tree status. The port status is programmed using the Control Command Frame.
- 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 to CPU(Default 0)**
- 0 – Disable report to CPU
  - 1 – Enable report to CPU
- When disable new VLAN port association report, new MAC address report or aging reports are disable for all ports. When enable, register SE\_OPEMODE is used to enable/disable selectively each function.
- Bit [6]: **Disable MII Management State Machine (Default 0)**
- 0: Enable MII Management State Machine
  - 1: Disable MII Management State Machine

- Bit [7]: Disable using MCT Link List structure (**Default 0**)
- 0 – Enable using MCT Link structure
  - 1 - Disable using MCT Link List structure

### 13.10.4 MIIC0 – 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.

### 13.10.5 MIIC1 – MII Command Register 1

CPU Address:h604

Accessed by CPU and serial interface only (R/W)

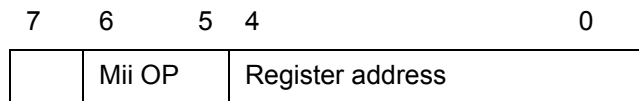
- Bit [7:0] - MII Data [15:8]

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

### 13.10.6 MIIC2 – MII Command Register 2

CPU Address:h605

Accessed by CPU and serial interface only (R/W)



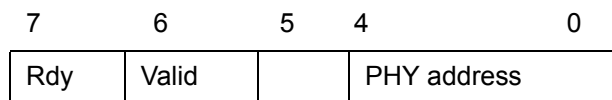
- Bit [4:0] - REG\_AD – Register PHY Address
- Bit [6:5] - OP – Operation code “10” for read command and “01” for write command

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

### 13.10.7 MIIC3 – MII Command Register 3

CPU Address:h606

Accessed by CPU and serial interface only (R/W)



- 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. Writing this register will initiate a serial management cycle to the MII management interface.

### 13.10.8 MIID0 – MII Data Register 0

CPU Address:h607

Accessed by CPU and serial interface only (RO)

- Bit [7:0] - MII Data [7:0]

### 13.10.9 MIID1 – MII Data Register 1

CPU Address:h608

Accessed by CPU and serial interface only (RO)

- Bit [7:0] - MII Data [15:8]

### 13.10.10 OLED Mode – LED Control

CPU Address:h609

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	5	4	3	2	1	0
			Clock rate	Hold Time		

- Bit [0] Reserved(Default 0)
- Bit[2:1]: Hold time for LED signal (**Default 00**)  
00=8msec 01=16msec  
10=32msec 11=64msec
- Bit[4:3]: LED clock frequency (**Default 0**)  
00=100M/8=12.5 MHz 01=100M/16= 25 MHz  
10=100M/32= 125 MHz 11=100M/64=1.5625 MHz
- Bit[6]: Reserved. Must be set to '0' (**Default 0**)

### 13.10.11 CHECKSUM - EEPROM Checksum

I<sup>2</sup>C Address FF, CPU Address:h60b

Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

This register is used in unmanaged mode only. Before requesting that the ZL50416 updates the EEPROM device, the correct checksum needs to be calculated and written into this checksum register. The checksum formula is:

$$\sum_{i=0}^{FF} \text{i}^2\text{C register} = 0$$

When the ZL50416 boots from the EEPROM the checksum is calculated and the value must be zero. If the checksum is not zeroed the ZL50416 does not start and pin CHECKSUM\_OK is set to zero.

## 13.11 (Group 7 Address) Port Mirroring Group

### 13.11.1 MIRROR1\_SRC – Port Mirror source port

CPU Address 700

Accessed by CPU and serial interface (R/W) (Default 7F)

7	6	5	4	0
		I/O	Src Port Select	

- Bit [4:0]: Source port to be mirrored. Use illegal port number to disable mirroring
- Bit [5]: 1 – select ingress data  
0 – select egress data
- Bit [6]: Reserved
- Bit [7]: Reserved must be set to '1'

### 13.11.2 MIRROR1\_DEST – Port Mirror destination

CPU Address 701

Accessed by CPU, serial interface (R/W) (Default 17)

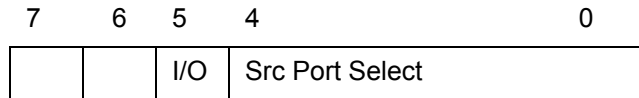
7	5	4	0
			Dest Port Select

- Bit [4:0]: Port Mirror Destination  
When port mirroring is enable, destination port can not serve as a data port.

### 13.11.3 MIRROR2\_SRC – Port Mirror source port

CPU Address 702

Accessed by CPU, serial interface (R/W) (Default FF)

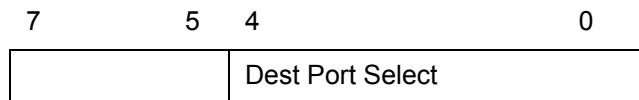


- Bit [4:0]: Source port to be mirrored. Use illegal port number to disable mirroring
- Bit [5]: 1 – select ingress data  
0 – select egress data
- Bit [6] Reserved
- Bit [7] Reserved must be set to '1'

### 13.11.4 MIRROR2\_DEST – Port Mirror destination

CPU Address 703

Accessed by CPU, serial interface (R/W) (Default 00)



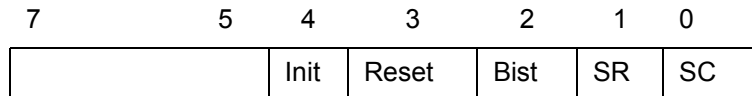
- Bit [4:0]: Port Mirror Destination  
When port mirroring is enable, destination port can not serve as a data port.

## 13.12 (Group F Address) CPU Access Group

### 13.12.1 GCR-Global Control Register

CPU Address: hF00

Accessed by CPU and serial interface. (R/W)



- 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 chip
- Bit[4]: Initialization Done (Default = 0).  
This bit is meaningless in unmanaged mode. 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 complete.

### 13.12.2 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<sup>2</sup>C  
0: Not busy (not writing configuration to I<sup>2</sup>C)
- Bit[1]: 1: Busy reading configuration from I<sup>2</sup>C  
0: Not busy ( not reading configuration from I<sup>2</sup>C)
- Bit[2]: 1: BIST in progress  
0: BIST not running
- Bit[3]: 1: RAM Error  
0: RAM OK
- Bit[5:4]: Device Signature  
11: ZL50416 device
- Bit [7:6]: Revision  
00: Initial Silicon  
01: XA1 Silicon  
10: Production Silicon

### 13.12.3 DCR1-Chip Status

CPU Address: hF02

Accessed by CPU and serial interface. (RO)

7	6	0
CIC		

- Bit [7] Chip initialization completed



### 13.12.4 DPST – Device Port Status Register

CPU Address:hF03

Accessed by CPU and serial interface (R/W)

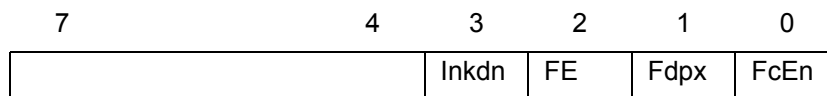
- Bit[4:0]: Read back index register. This is used for selecting what to read back from DTST. (Default 00)
- 5'b00000 - Port 0 Operating mode and Negotiation status
  - 5'b00001 - Port 1 Operating mode and Negotiation status
  - 5'b00010 - Port 2 Operating mode and Negotiation status
  - 5'b00011 - Port 3 Operating mode and Negotiation status
  - 5'b00100 - Port 4 Operating mode and Negotiation status
  - 5'b00101 - Port 5 Operating mode and Negotiation status
  - 5'b00110 - Port 6 Operating mode and Negotiation status
  - 5'b00111 - Port 7 Operating mode and Negotiation status
  - 5'b01000 - Port 8 Operating mode and Negotiation status
  - 5'b01001 - Port 9 Operating mode and Negotiation status
  - 5'b01010 - Port 10 Operating mode and Negotiation status
  - 5'b01011 - Port 11 Operating mode and Negotiation status
  - 5'b01100 - Port 12 Operating mode and Negotiation status
  - 5'b01101 - Port 13 Operating mode and Negotiation status
  - 5'b01110 - Port 14 Operating mode and Negotiation status
  - 5'b01111 - Port 15 Operating mode and Negotiation status
  - 5'b10xxx - Reserved
  - 5'b11000 - Port 24 Operating mode/Neg status (CPU port)

### 13.12.5 DTST – Data Read Back Register

CPU Address: hF04

Accessed by CPU and serial interface (RO)

This register provides various internal information as selected in DPST bit[4:0]. Refer to the PHY Control Application Note.



When bit is 1:

Bit[0] – Flow control enable

Bit[1] – Full duplex port

Bit[2] – Fast Ethernet port

Bit[3] – Link is down

Bit[7:4] – Reserved

### 13.12.6 PLLCR - PLL Control Register

CPU Address: hF05

Accessed by serial interface (RW)

Bit[3] - Must be '1'

Bit[7] - Selects strap option or LCLK/OECLK registers

- 0 - Strap option (default)
- 1 - LCLK/OECLK registers

### 13.12.7 LCLK - LA\_CLK delay from internal OE\_CLK

CPU Address: hF06

Accessed by serial interface (RW)

PD[12:10]	LCLK	Delay
000b	80h	8 Buffers Delay
001b	40h	7 Buffers Delay
010b	20h	6 Buffers Delay
011b	10h	5 Buffers Delay (Recommend)
100b	08h	4 Buffers Delay
101b	04h	3 Buffers Delay
110b	02h	2 Buffers Delay
111b	01h	1 Buffers Delay

The LCLK delay from SCLK is the sum of the delay programmed in here and the delay in OECLK register.

### 13.12.8 OECLK - Internal OE\_CLK delay from SCLK

CPU Address: hF07

Accessed by serial interface (RW)

The OE\_CLK is used for generating the OE0 and OE1 signals.

PD[15:13]	OECLK	Delay
000b	80h	8 Buffers Delay
001b	40h	7 Buffers Delay (Recommend)
010b	20h	6 Buffers Delay
011b	10h	5 Buffers Delay
100b	08h	4 Buffers Delay
101b	04h	3 Buffers Delay
110b	02h	2 Buffers Delay
111b	01h	1 Buffers Delay

### 13.12.9 DA – DA Register

CPU Address: hFFF

Accessed by CPU and serial interface (RO)

Always return 8'h **DA**. Indicate the CPU interface or serial port connection is good.

## 14.0 BGA and Ball Signal Descriptions

### 14.1 BGA Views (TOP-View)

#### 14.1.1 Encapsulated View In Unmanaged Mode

	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	
A			LA_D4	LA_D7	LA_D10	LA_D13	LA_D15	LA_D14	LA_A4	LA_OE0	LA_A8	LA_A13	LA_A16	LA_A19	LA_D33	LA_D36	LA_D39	LA_D42	LA_D45	OE_CLK0	LA_CLK0	TRUNK1	RESE_RVED	RESE_RVED	SCL	SDA	STROBE	TSTOUT7		
B		LA_D1	LA_D3	LA_D6	LA_D9	LA_D12	LA_D14	LA_A DSC	LA_OE1	LA_A7	LA_A12	LA_A15	LA_A18	LA_D32	LA_D35	LA_D38	LA_D41	LA_D44	LA_D47	OE_CLK1	LA_CLK1	LA_D62	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	D0	TSTOUT8	TSTOUT3	
C	LA_CLK	LA_D0	LA_D2	LA_D5	LA_D8	LA_D11	LA_A3	LA_OE	LA_W E	T_M ODE1	LA_A11	LA_A14	LA_A17	LA_D20	LA_D34	LA_D37	LA_D40	LA_D43	OE_CLK2	LA_CLK2	P_D	TRUNK0	RESE_RVED	RESE_RVED	RESE_RVED	AUTO FD	TSTOUT11	TSTOUT9	TSTOUT4	TSTOUT0
D	AGND	LA_D17	LA_D19	LA_D21	LA_D23	LA_D25	LA_D27	LA_D29	LA_D31	LA_A6	LA_A10	LA_W E0	LA_D49	LA_D51	LA_D53	LA_D55	LA_D57	LA_D59	LA_D61	LA_D63	LA_D67	LA_D44	SCAN COL	SCAN CLK	TSTOUT14	TSTOUT13	TSTOUT12	TSTOUT10	TSTOUT5	TSTOUT1
E	SCLK	LA_D18	LA_D11	LA_D22	LA_D24	LA_D26	LA_D28	LA_D30	LA_A5	LA_A9	LA_W E1	LA_D48	LA_D50	LA_D52	LA_D54	LA_D56	LA_D58	LA_D60	RESE_RVED	LA_D46		SCAN LINK	TSTOUT15	RESE_RVED	RESE_RVED	SCAN MODE	TSTOUT6	TSTOUT2		
F	AVCC	RESIN	SCAN EN	RESE_RVED	RESE_RVED								VCC	VCC	VCC	VCC	VCC									RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
G	RESE_RVED	RESE_TOUT	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
H	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
J	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
K	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED							VDD	VDD					VDD	VDD							RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
L	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
M	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED					VDD			VSS	VSS	VSS	VSS	VSS	VSS	VSS			VDD			RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	
N	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	VCC				VDD			VSS	VSS	VSS	VSS	VSS	VSS	VSS					VCC	RESE_RVED	RESE_RVED			RESE_RVED	
P	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	VCC							VSS	VSS	VSS	VSS	VSS	VSS	VSS					VCC	RESE_RVED	RESE_RVED		MDIO	RESE_RVED	
R	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	VCC							VSS	VSS	VSS	VSS	VSS	VSS	VSS					VCC	RESE_RVED	RESE_RVED		MDC	M_CLK	
T	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	VCC							VSS	VSS	VSS	VSS	VSS	VSS	VSS					VCC	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	
U	RESE_RVED	RESE_RVED	T_M ODE0	RESE_RVED	RESE_RVED	VCC				VDD			VSS	VSS	VSS	VSS	VSS	VSS	VSS					VCC	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	
V	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED					VDD			VSS	VSS	VSS	VSS	VSS	VSS	VSS					VDD		RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
W	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
Y	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED							VDD	VDD					VDD	VDD							RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AA	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AB	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AC	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED																					RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AD	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED									VCC	VCC	VCC	VCC	VCC								RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AE	M0_TX EN	M0_TX D0	M0_TX D1	M3_TX D1	M3_TX EN	M3_TX D0	M5_TX D1	M5_TX EN	M5_TX D0	M8_TX D1	M8_TX EN	M8_TX D0	M10_TX D1	M10_TX EN	M10_TX D0	M13_TX D1	M13_TX EN	M13_TX D0	M13_TX D1	M13_TX D0	M14_TX D1	M14_TX D0	M15_TX EN	M15_TX D0	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AF	M0_RX D1	M0_RX D0	M0_C RS	M3_TX D0	M3_C RS	M3_TX D1	M5_TX D0	M5_C RS	M5_TX D1	M8_TX D0	M8_C RS	M8_TX D1	M10_TX D0	M10_C RS	M10_TX D1	M13_TX D0	M13_C RS	M13_TX D1	M13_TX D0	M14_TX D1	M14_C RS	M14_TX D0	M15_TX D1	M15_C RS	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AG	M1_TX EN	M1_TX D0	M1_TX D1	M2_TX D1	M2_C RS	M4_TX D1	M4_C RS	M6_TX D1	M6_C RS	M7_TX D1	M7_C RS	M9_TX D1	M9_C RS	M11_TX D1	M11_C RS	M12_TX D1	M12_C RS	M14_TX D1	M14_TX D0	M15_TX D0	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AH		M1_RX D0	M1_C RS	M2_TX D0	M2_RX D0	M4_TX D0	M4_RX D0	M6_TX D0	M6_RX D0	M7_TX D0	M7_RX D0	M9_TX D0	M9_RX D0	M11_TX D0	M11_RX D0	M12_TX D0	M12_RX D0	M14_TX D0	M14_RX D0	M15_C RS	M15_RX D0	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED
AJ			M1_RX D1	M2_TX EN	M2_RX D1	M4_TX EN	M4_RX D1	M6_TX EN	M6_RX D1	M7_TX EN	M7_RX D1	M9_TX EN	M9_RX D1	M11_TX EN	M11_RX D1	M12_TX EN	M12_RX D1	M14_TX EN	M14_RX D1	RESE_RVED	M13_TX EN	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED	RESE_RVED



## 14.2 Ball – Signal Descriptions in Managed Mode

All pins are CMOS type; all Input Pins are 5 Volt tolerance; and all Output Pins are 3.3 CMOS drive.

### 14.2.1 Ball Signal Descriptions in Managed Mode

Ball No(s)	Symbol	I/O	Description
<b>CPU BUS Interface in Managed Mode</b>			
C19, B19, A19, C20, B20, A20, C21, E20, A21, B24, B22, A22, C23, B23, A23, C24	P_DATA[15:0]	I/O-TS with pull up Except P_DATA[7:6] I/O-TS with pull down	Processor Bus Data Bit [15:0]. P_DATA[7:0] is used in 8-bit mode.
C22, A24, A25	P_A[2:0]	Input	Processor Bus Address Bit [2:0]
A26	P_WE#	Input with weak internal pull up	CPU Bus-Write Enable
B26	P_RD#	Input with weak internal pull up	CPU Bus-Read Enable
C25	P_CS#	Input with weak internal pull up	Chip Select
B25	P_INT#	Output	CPU Interrupt
<b>Frame Buffer Interface</b>			
D20, B21, D19, E19, D18, E18, D17, E17, D16, E16, D15, E15, D14, E14, D13, E13, D21, E21, A18, B18, C18, A17, B17, C17, A16, B16, C16, A15, B15, C15, A14, B14, D9, E9, D8, E8, D7, E7, D6, E6, D5, E5, D4, E4, D3, E3, D2, E2, A7, B7, A6, B6, C6, A5, B5, C5, A4, B4, C4, A3, B3, C3, B2, C2	LA_D[63:0]	I/O-TS with pullup	Frame Bank A– Data Bit [63:0]
C14, A13, B13, C13, A12, B12, C12, A11, B11, C11, D11, E11, A10, B10, D10, E10, A8, C7	LA_A[20:3]	Output	Frame Bank A – Address Bit [20:3]
B8	LA_ADSC#	Output with pull up	Frame Bank A Address Status Control
C1	LA_CLK	Output	Frame Bank A Clock Input
C9	LA_WE#	Output with pull up	Frame Bank A Write Chip Select for one layer SRAM configuration

Ball No(s)	Symbol	I/O	Description
D12	LA_WE0#	Output with pull up	Frame Bank A Write Chip Select for lower layer of two layers SRAM configuration
E12	LA_WE1#	Output with pull up	Frame Bank A Write Chip Select for upper layer of two layers SRAM configuration
C8	LA_OE#	Output with pull up	Frame Bank A Read Chip Select for one bank SRAM configuration
A9	LA_OE0#	Output with pull up	Frame Bank A Read Chip Select for lower layer of two layers SRAM configuration
B9	LA_OE1#	Output with pull up	Frame Bank A Read Chip Select for upper layer of two layers SRAM configuration
<b>Fast Ethernet Access Ports [15:0] RMII</b>			
R28	M_MDC	Output	MII Management Data Clock – (Common for all MII Ports [15:0])
P28	M_MDIO	I/O-TS with pull up	MII Management Data I/O – (Common for all MII Ports –[15:0])
R29	M_CLKI	Input	Reference Input Clock
AF21, AJ19, AF18, AJ17, AJ15, AF15, AJ13, AF12, AJ11, AJ9, AF9, AJ7, AF6, AJ5, AJ3, AF1	M[15:0]_RXD[1]	Input with weak internal pull up resistors.	Ports [15:0] – Receive Data Bit [1]
AE21, AH19, AH20, AH17, AH15, AE15, AH13, AE12, AH11, AH9, AE9, AH7, AE6, AH5, AH2, AF2	M[15:0]_RXD[0]	Input with weak internal pull up resistors	Ports [15:0] – Receive Data Bit [0]
AH21, AF19, AF17, AG17, AG15, AF14, AG13, AF11, AG11, AG9, AF8, AG7, AF5, AG5, AH3, AF3	M[15:0]_CRS_D V	Input with weak internal pull down resistors.	Ports [15:0] – Carrier Sense and Receive Data Valid
AE20, AJ18, AJ21, AJ16, AJ14, AE14, AJ12, AE11, AJ10, AJ8, AE8, AJ6, AE5, AJ4, AG1, AE1	M[15:0]_TXEN	I/O- TS with pull up , slew	Ports [15:0] – Transmit Enable Strap option for RMII/GPSI
AE18, AG18, AE16, AG16, AG14, AE13, AG12, AE10, AG10, AG8, AE7, AG6, AE4, AG4, AG3, AE3	M[15:0]_TXD[1]	Output, slew	Ports [15:0] – Transmit Data Bit [1]

Ball No(s)	Symbol	I/O	Description
AG19, AH18, AF16, AH16, AH14, AF13, AH12, AF10, AH10, AH8, AF7, AH6, AF4, AH4, AG2, AE2	M[15:0]_TXD[0]	Output, slew	Ports [15:0] – Transmit Data Bit [0]
<b>LED Interface</b>			
C29	LED_CLK/TST OUT0	I/O- TS with pull up	LED Serial Interface Output Clock
D29	LED_SYN/TST OUT1	I/O- TS with pull up	LED Output Data Stream Envelope
E29	LED_BIT/TSTO UT2	I/O- TS with pull up	LED Serial Data Output Stream
C27	INIT_DONE/TS TOUT9	I/O- TS with pull up	System start operation
D27	INIT_START/TS TOUT10	I/O- TS with pull up	Start initialization
C26	CHECKSUM_O K/TSTOUT11	I/O- TS with pull up	EEPROM read OK
D26	FCB_ERR/TST OUT12	I/O- TS with pull up	FCB memory self test fail
D25	MCT_ERR/TST OUT13	I/O- TS with pull up	MCT memory self test fail
D24	BIST_IN_PRC/T STOUT14	I/O- TS with pull up	Processing memory self test
E24	BIST_DONE/TS TOUT15	I/O- TS with pull up	Memory self test done
<b>Test Facility</b>			
U3, C10	T_MODE0, T_MODE1	I/O-TS	Test Pins 00 – Test mode – Set Mode upon Reset, and provides NAND Tree test output during test mode 01 - Reserved - Do not use 10 - Reserved - Do not use 11 – Normal mode. Use external pull up for normal mode
F3	SCAN_EN	Input with pull down	Scan Enable
E27	SCANMODE	Input with pull down	1 – Enable Test mode 0 - Normal mode (open)
<b>System Clock, Power, and Ground Pins</b>			
E1	SCLK	Input	System Clock at 100 MHz

Ball No(s)	Symbol	I/O	Description
K12, K13, K17, K18 M10, N10, M20, N20, U10, V10, U20, V20, Y12, Y13, Y17, Y18	VDD	Power	+2.5 Volt DC Supply
F13, F14, F15, F16, F17, N6, P6, R6, T6, U6, N24, P24, R24, T24, U24, AD13, AD14, AD15, AD16, AD17	VCC	Power	+3.3 Volt DC Supply
M12, M13, M14, M15, M16, M17, M18, N12, N13, N14, N15, N16, N17, N18, P12, P13, P14, P15, P16, P17, P18, R12, R13, R14, R15, R16, R17, R18, T12, T13, T14, T15, T16, T17, T18, U12, U13, U14, U15, U16, U17, U18, V12, V13, V14, V15, V16, V17, V18,	VSS	Power Ground	Ground
F1	AVCC	Analog Power	Analog +2.5 Volt DC Supply
D1	AGND	Analog Ground	Analog Ground
<b>MISC</b>			
D22	SCANCOL	Input/ output	Scans the Collision signal of Home PHY
D23	SCANCLK	Output	Clock for scanning Home PHY collision and link
E23	SCANLINK	Input/ output	Link up signal from Home PHY
F2	RESIN#	Input	Reset Input
G2	RESETOUT#	Output	Reset PHY



Ball No(s)	Symbol	I/O	Description
F4, F5, G4, G5, H4, H5, J4, J5, K4, K5, L4, L5, M4, M5, N4, N5, G3, H1, H2, H3, J1, J2, J3, K1, K2, K3, L1, L2, L3, M1, M2, M3, U4, U5, V4, V5, W4, W5, Y4, Y5, AA4, AA5, AB4, AB5, AC4, AC5, AD4, AD5, W1, Y1, Y2, Y3, AA1, AA2, AA3, AB1, AB2, AB3, AC1, AC2, AC3, AD1, AD2, AD3, N3, N2, N1, P3, P2, P1, R5, R4, R3, R2, R1, T5, T4, T3, T2, T1, W3, W2, V1, G1, V3, P4, P5, V2, U1, U2, U26, U25, V26, V25, W26, W25, Y27, Y26, AA26, AA25, AB26, AB25, AC26, AC25, AD26, AD25, T28, U28, R25, U29, T29, U27, V29, V28, V27, W29, W28, W27, Y29, Y28, Y25, AA29, AA28, AA27, AB29, AB28, AB27, T26, R26, T27, T25, P29, G26, G25, H26, H25, J26, J25, K25, K26, M25, L26, M26, L25, N26, N25, P26, P25, F28, G28, E25, G29, F29, G27, H29, H28, H27, J29, J28, J27, K29, K28, K27, L29, L28, L27, M29, M28, M27, F26, E26, F27, F25, N29, AC29, AE28, AJ27, AF27, AJ25, AF24, AH23, AE19, AC28, AF28, AH27, AE27, AH25, AE24, AF22, AF20, AC27, AF29, AG27, AF26, AG25, AG23, AF23, AG21, AD29, AG28, AJ26, AE26, AJ24, AE23, AJ22, AJ20, AD27, AH28, AG26, AE25, AG24, AE22, AJ23, AG20, AD28, AG29, AH26, AF25, AH24, AG22, AH22, AE17,	RESERVED	NA	Reserved Pins. Leave unconnected.

Ball No(s)	Symbol	I/O	Description
<b>Bootstrap Pins (Default = pull up, 1= pull up 0= pull down)</b>			
After reset TSTOUT0 to TSTOU15 are used by the LED interface.			
C29	TSTOUT0		Reserved
D29	TSTOUT1	Default 1	RMI MAC Power Saving Enable 0 – No power saving 1 – power saving
C28, B28, E29	TSTOUT[4:2]		Reserved
D28	TSTOUT5	Default 1	Scan Speed: ¼ SCLK or SCLK 0 – ¼ SCLK (HPNA) 1 – SCLK
E28	TSTOUT6	Default 1	CPU Port Mode 0 - 8 bit Bus Mode 1 - 16 bit Bus Mode
A27	TSTOUT7	Default 1	Memory Size 0 - 256K x 32 or 256K x 64 (4M total) 1 - 128K x 32 or 128K x 64 (2M total)
B27	TSTOUT8	Default 1	EEPROM Installed 0 – EEPROM installed 1 – EEPROM not installed
C27	TSTOUT9	Default 1	MCT Aging 0 – MCT aging disable 1 – MCT aging enable
D27	TSTOUT10	Default 1	FCB Aging 0 - FCB aging disable 1 – FCB aging enable
C26	TSTOUT11	Default 1	Timeout Reset 0 – Time out reset disable 1 – Time out reset enable. Issue reset if any state machine did not go back to idle for 5sec.
D26	TSTOUT12		Reserved
D25	TSTOUT13	Default 1	FDB RAM depth (1 or 2 layers) 0 – 2 layer 1 – 1 layer
D24	TSTOUT14	Default 1	CPU installed 0 – CPU installed 1 – CPU not installed
E24	TSTOUT15	Default 1	SRAM Test Mode 0 – Enable test mode 1 – Normal operation

Ball No(s)	Symbol	I/O	Description
AE20, AJ18, AJ21, AJ16, AJ14, AE14, AJ12, AE11, AJ10, AJ8, AE8, AJ6, AE5, AJ4, AG1, AE1	M[15:0] TXEN	Default: RMII	0 – GPSI 1 – RMII
C21	P_D[9]	Must be pulled-down	Reserved - Must be pulled-down
C19, B19, A19	P_D[15:13]	Default: 111	Programmable delay for internal OE_CLK from SCLK input. The OE_CLK is used for generating the OE0 and OE1 signals. Suggested value is 001.
C20, B20, A20	P_D[12:10]	Default: 111	Programmable delay for LA_CLK from internal OE_CLK. The LA_CLK delay from SCLK is the sum of the delay programmed in here and the delay in P_D[15:13]. Suggested value is 011.

## 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

### 14.2.2 Ball – Signal Descriptions in Unmanaged Mode

Ball No(s)	Symbol	I/O	Description
<b>I<sup>2</sup>C Interface Note: In unmanaged mode, Use I<sup>2</sup>C and Serial control interface to configure the system</b>			
A24	SCL	Output	I <sup>2</sup> C Data Clock
A25	SDA	I/O-TS with internal pull up	I <sup>2</sup> C Data I/O
<b>Serial Control Interface</b>			
A26	STROBE	Input with weak internal pull up	Serial Strobe Pin
B26	D0	Input with weak internal pull up	Serial Data Input
C25	AUTOFD	Output with pull up	Serial Data Output (AutoFD)
<b>Frame Buffer Interface</b>			
D20, B21, D19, E19, D18, E18, D17, E17, D16, E16, D15, E15, D14, E14, D13, E13, D21, E21, A18, B18, C18, A17, B17, C17, A16, B16, C16, A15, B15, C15, A14, B14, D9, E9, D8, E8, D7, E7, D6, E6, D5, E5, D4, E4, D3, E3, D2, E2, A7, B7, A6, B6, C6, A5, B5, C5, A4, B4, C4, A3, B3, C3, B2, C2	LA_D[63:0]	I/O-TS with pull up	Frame Bank A– Data Bit [63:0]
C14, A13, B13, C13, A12, B12, C12, A11, B11, C11, D11, E11, A10, B10, D10, E10, A8, C7	LA_A[20:3]	Output	Frame Bank A – Address Bit [20:3]
B8	LA_ADSC#	Output with pull up	Frame Bank A Address Status Control
C1	LA_CLK	Output with pull up	Frame Bank A Clock Input
C9	LA_WE#	Output with pull up	Frame Bank A Write Chip Select for one layer SRAM application
D12	LA_WE0#	Output with pull up	Frame Bank A Write Chip Select for lower layer of two bank SRAM application
E12	LA_WE1#	Output with pull up	Frame Bank A Write Chip Select for upper bank of two layer SRAM application
C8	LA_OE#	Output with pull up	Frame Bank A Read Chip Select for one layer SRAM application

Ball No(s)	Symbol	I/O	Description
A9	LA_OE0#	Output with pull up	Frame Bank A Read Chip Select for lower layer of two layers SRAM application
B9	LA_OE1#	Output with pull up	Frame Bank A Read Chip Select for upper layer of two layers SRAM application
<b>Fast Ethernet Access Ports [15:0] RMII</b>			
R28	M_MDC	Output	MII Management Data Clock – (Common for all MII Ports [15:0])
P28	M_MDIO	I/O-TS with pull up	MII Management Data I/O – (Common for all MII Ports –[15:0])
R29	M_CLKI	Input	Reference Input Clock
AF21, AJ19, AF18, AJ17, AJ15, AF15, AJ13, AF12, AJ11, AJ9, AF9, AJ7, AF6, AJ5, AJ3, AF1	M[15:0]_RXD[1]	Input with weak internal pull up resistors.	Ports [15:0] – Receive Data Bit [1]
AE21, AH19, AH20, AH17, AH15, AE15, AH13, AE12, AH11, AH9, AE9, AH7, AE6, AH5, AH2, AF2	M[15:0]_RXD[0]	Input with weak internal pull up resistors	Ports [15:0] – Receive Data Bit [0]
AH21, AF19, AF17, AG17, AG15, AF14, AG13, AF11, AG11, AG9, AF8, AG7, AF5, AG5, AH3, AF3	M[15:0]_CRS_D V	Input with weak internal pull down resistors.	Ports [15:0] – Carrier Sense and Receive Data Valid
AE20, AJ18, AJ21, AJ16, AJ14, AE14, AJ12, AE11, AJ10, AJ8, AE8, AJ6, AE5, AJ4, AG1, AE1	M[15:0]_TXEN	I/O- TS with pull up , slew	Ports [15:0] – Transmit Enable Strap option for RMII/GPSI
AE18, AG18, AE16, AG16, AG14, AE13, AG12, AE10, AG10, AG8, AE7, AG6, AE4, AG4, AG3, AE3	M[15:0]_TXD[1]	Output, slew	Ports [15:0] – Transmit Data Bit [1]
AG19, AH18, AF16, AH16, AH14, AF13, AH12, AF10, AH10, AH8, AF7, AH6, AF4, AH4, AG2, AE2	M[15:0]_TXD[0]	Output, slew	Ports [15:0] – Transmit Data Bit [0]
<b>LED Interface</b>			
C29	LED_CLK/TSTO UT0	I/O- TS with pull up	LED Serial Interface Output Clock
D29	LED_SYN/TST OUT1	I/O- TS with pull up	LED Output Data Stream Envelope

Ball No(s)	Symbol	I/O	Description
E29	LED_BIT/TSTOUT2	I/O- TS with pull up	LED Serial Data Output Stream
C27	INIT_DONE/TSTOUT9	I/O- TS with pull up	System start operation
D27	INIT_START/TSTOUT10	I/O- TS with pull up	Start initialization
C26	CHECKSUM_OK/TSTOUT11	I/O- TS with pull up	EEPROM read OK
D26	FCB_ERR/TSTOUT12	I/O- TS with pull up	FCB memory self test fail
D25	MCT_ERR/TSTOUT13	I/O- TS with pull up	MCT memory self test fail
D24	BIST_IN_PRC/TSTOUT14	I/O- TS with pull up	Processing memory self test
E24	BIST_DONE/TSTOUT15	I/O- TS with pull up	Memory self test done
<b>Trunk Enable</b>			
C22	TRUNK0	Input w/ weak internal pull down resistors	Trunk Port Enable in unmanaged mode In managed mode doesn't care
A21	TRUNK1	Input w/ weak internal pull down resistors	Trunk Port Enable in unmanaged mode In managed mode doesn't care
<b>Test Facility</b>			
U3, C10	T_MODE0, T_MODE1	I/O-TS	Test Pins 00 – Test mode – Set Mode upon Reset, and provides NAND Tree test output during test mode 01 - Reserved - Do not use 10 - Reserved - Do not use 11 – Normal mode. Use external pull up for normal mode
F3	SCAN_EN	Input with pull down	Scan Enable 0 - Normal mode (open)
E27	SCANMODE	Input with pull down	1 – Enable Test mode 0 - Normal mode (open)
<b>System Clock, Power, and Ground Pins</b>			
E1	SCLK	Input	System Clock at 100 MHz

Ball No(s)	Symbol	I/O	Description
K12, K13, K17, K18 M10, N10, M20, N20, U10, V10, U20, V20, Y12, Y13, Y17, Y18	VDD	Power	+2.5 Volt DC Supply
F13, F14, F15, F16, F17, N6, P6, R6, T6, U6, N24, P24, R24, T24, U24, AD13, AD14, AD15, AD16, AD17	VCC	Power	+3.3 Volt DC Supply
M12, M13, M14, M15, M16, M17, M18, N12, N13, N14, N15, N16, N17, N18, P12, P13, P14, P15, P16, P17, P18, R12, R13, R14, R15, R16, R17, R18, T12, T13, T14, T15, T16, T17, T18, U12, U13, U14, U15, U16, U17, U18, V12, V13, V14, V15, V16, V17, V18,	VSS	Power Ground	Ground
F1	AVCC	Analog Power	Analog +2.5 Volt DC Supply
D1	AGND	Analog Ground	Analog Ground
<b>MISC</b>			
D22	SCANCOL	Input	Scans the Collision signal of Home PHY
D23	SCANCLK	Input/ output	Clock for scanning Home PHY collision and link
E23	SCANLINK	Input	Link up signal from Home PHY
F2	RESIN#	Input	Reset Input
G2	RESETOUT#	Output	Reset PHY

Ball No(s)	Symbol	I/O	Description
F4, F5, G4, G5, H4, H5, J4, J5, K4, K5, L4, L5, M4, M5, N4, N5, G3, H1, H2, H3, J1, J2, J3, K1, K2, K3, L1, L2, L3, M1, M2, M3, U4, U5, V4, V5, W4, W5, Y4, Y5, AA4, AA5, AB4, AB5, AC4, AC5, AD4, AD5, W1, Y1, Y2, Y3, AA1, AA2, AA3, AB1, AB2, AB3, AC1, AC2, AC3, AD1, AD2, AD3, N3, N2, N1, P3, P2, P1, R5, R4, R3, R2, R1, T5, T4, T3, T2, T1, W3, W2, V1, G1, V3, P4, P5, V2, U1, U2, U26, U25, V26, V25, W26, W25, Y27, Y26, AA26, AA25, AB26, AB25, AC26, AC25, AD26, AD25, T28, U28, R25, U29, T29, U27, V29, V28, V27, W29, W28, W27, Y29, Y28, Y25, AA29, AA28, AA27, AB29, AB28, AB27, T26, R26, T27, T25, P29, G26, G25, H26, H25, J26, J25, K25, K26, M25, L26, M26, L25, N26, N25, P26, P25, F28, G28, E25, G29, F29, G27, H29, H28, H27, J29, J28, J27, K29, K28, K27, L29, L28, L27, M29, M28, M27, F26, E26, F27, F25, N29, B24, AC29, AE28, AJ27, AF27, AJ25, AF24, AH23, AE19, AC28, AF28, AH27, AE27, AH25, AE24, AF22, AF20, AC27, AF29, AG27, AF26, AG25, AG23, AF23, AG21, AD29, AG28, AJ26, AE26, AJ24, AE23, AJ22, AJ20, AD27, AH28, AG26, AE25, AG24, AE22, AJ23, AG20, AD28, AG29, AH26, AF25, AH24, AG22, AH22, AE17, E20, B25, B22, A22, C23, B23, A23, C24	RESERVED	NA	Reserved Pins. Leave unconnected.
Bootstrap Pins (Default = pull up, 1= pull up 0= pull down)			
After reset TSTOUT0 to TSTOU15 are used by the LED interface.			
C29	TSTOUT0		Reserved



Ball No(s)	Symbol	I/O	Description
D29	TSTOUT1	Default 1	RMII MAC Power Saving Enable 0 – No power saving 1 – power saving
C28, B28, E29	TSTOUT[4:2]		Reserved
D28	TSTOUT5	Default 1	Scan Speed: ¼ SCLK or SCLK 0 – ¼ SCLK (HPNA) 1 - SCLK
E28	TSTOUT6	Default 1	CPU Port Mode 0 - 8 bit Bus Mode 1 - 16 bit Bus Mode
A27	TSTOUT7	Default 1	Memory Size 0 - 256K x 32 or 256K x 64 (4M total) 1 - 128K x 32 or 128K x 64 (2M total)
B27	TSTOUT8	Default 1	EEPROM Installed 0 – EEPROM installed 1 – EEPROM not installed
C27	TSTOUT9	Default 1	MCT Aging 0 – MCT aging disable 1 – MCT aging enable
D27	TSTOUT10	Default 1	FCB Aging 0 - FCB aging disable 1 – FCB aging enable
C26	TSTOUT11	Default 1	Reserved
D26	TSTOUT12		Test Speed Up 0 – Enable test speed up. Do not use 1 – Disable test speed up
D25	TSTOUT13	Default 1	FDB RAM depth (1 or 2 layers) 0 – 2 layer 1 – 1 layer
D24	TSTOUT14	Default 1	CPU installed 0 – CPU installed 1 – CPU not installed
E24	TSTOUT15	Default 1	SRAM Test Mode 0 – Enable test mode 1 – Normal operation
AE20, AJ18, AJ21, AJ16, AJ14, AE14, AJ12, AE11, AJ10, AJ8, AE8, AJ6, AE5, AJ4, AG1, AE1,	M[15:0]_TXEN	Default: RMII	0 – GPSI 1 - RMII
C21	P_D	Must be pulled-down	Reserved - Must be pulled-down

Ball No(s)	Symbol	I/O	Description
C19, B19, A19	OE_CLK[2:0]	Default: 111	Programmable delay for internal OE_CLK from SCLK input. The OE_CLK is used for generating the OE0 and OE1 signals. Suggested value is 001.
C20, B20, A20	LA_CLK[2:0]	Default: 111	Programmable delay for LA_CLK from internal OE_CLK. The LA_CLK delay from SCLK is the sum of the delay programmed in here and the delay in P_D[15:13]. Suggested value is 011.

## 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

### 14.3 Ball – Signal Name in Unmanaged Mode

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
D20	LA_D[63]	D3	LA_D[19]	A9	LA_OE0#
B21	LA_D[62]	E3	LA_D[18]	B9	LA_OE1#
D19	LA_D[61]	D2	LA_D[17]	F4	RESERVED
E19	LA_D[60]	E2	LA_D[16]	F5	RESERVED
D18	LA_D[59]	A7	LA_D[15]	G4	RESERVED
E18	LA_D[58]	B7	LA_D[14]	G5	RESERVED
D17	LA_D[57]	A6	LA_D[13]	H4	RESERVED
E17	LA_D[56]	B6	LA_D[12]	H5	RESERVED
D16	LA_D[55]	C6	LA_D[11]	J4	RESERVED
E16	LA_D[54]	A5	LA_D[10]	J5	RESERVED
D15	LA_D[53]	B5	LA_D[9]	K4	RESERVED
E15	LA_D[52]	C5	LA_D[8]	K5	RESERVED
D14	LA_D[51]	A4	LA_D[7]	L4	RESERVED
E14	LA_D[50]	B4	LA_D[6]	L5	RESERVED
D13	LA_D[49]	C4	LA_D[5]	M4	RESERVED
E13	LA_D[48]	A3	LA_D[4]	M5	RESERVED
D21	LA_D[47]	B3	LA_D[3]	N4	RESERVED
E21	LA_D[46]	C3	LA_D[2]	N5	RESERVED
A18	LA_D[45]	B2	LA_D[1]	G3	RESERVED
B18	LA_D[44]	C2	LA_D[0]	H1	RESERVED
C18	LA_D[43]	C14	LA_A[20]	H2	RESERVED
A17	LA_D[42]	A13	LA_A[19]	H3	RESERVED
B17	LA_D[41]	B13	LA_A[18]	J1	RESERVED
C17	LA_D[40]	C13	LA_A[17]	J2	RESERVED
A16	LA_D[39]	A12	LA_A[16]	J3	RESERVED
B16	LA_D[38]	B12	LA_A[15]	K1	RESERVED
C16	LA_D[37]	C12	LA_A[14]	K2	RESERVED
A15	LA_D[36]	A11	LA_A[13]	K3	RESERVED
B15	LA_D[35]	B11	LA_A[12]	L1	RESERVED

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
C15	LA_D[34]	C11	LA_A[11]	L2	RESERVED
A14	LA_D[33]	D11	LA_A[10]	L3	RESERVED
B14	LA_D[32]	E11	LA_A[9]	M1	RESERVED
D9	LA_D[31]	A10	LA_A[8]	M2	RESERVED
E9	LA_D[30]	B10	LA_A[7]	M3	RESERVED
D8	LA_D[29]	D10	LA_A[6]	U4	RESERVED
E8	LA_D[28]	E10	LA_A[5]	U5	RESERVED
D7	LA_D[27]	A8	LA_A[4]	V4	RESERVED
E7	LA_D[26]	C7	LA_A[3]	V5	RESERVED
D6	LA_D[25]	B8	LA_DSC#	W4	RESERVED
E6	LA_D[24]	C1	LA_CLK	W5	RESERVED
D5	LA_D[23]	C9	LA_WE#	Y4	RESERVED
E5	LA_D[22]	D12	LA_WE0#	Y5	RESERVED
D4	LA_D[21]	E12	LA_WE1#	AA4	RESERVED
E4	LA_D[20]	C8	LA_OE#	AA5	RESERVED
AB4	RESERVED	U2	RESERVED	AH7	M[4]_RXD[0]
AB5	RESERVED	R28	MDC	AE6	M[3]_RXD[0]
AC4	RESERVED	P28	MDIO	AH5	M[2]_RXD[0]
AC5	RESERVED	R29	M_CLK	AH2	M[1]_RXD[0]
AD4	RESERVED	AC29	RESERVED	AF2	M[0]_RXD[0]
AD5	RESERVED	AE28	RESERVED	AC27	RESERVED
W1	RESERVED	AJ27	RESERVED	AF29	RESERVED
Y1	RESERVED	AF27	RESERVED	AG27	RESERVED
Y2	RESERVED	AJ25	RESERVED	AF26	RESERVED
Y3	RESERVED	AF24	RESERVED	AG25	RESERVED
AA1	RESERVED	AH23	RESERVED	AG23	RESERVED
AA2	RESERVED	AE19	RESERVED	AF23	RESERVED
AA3	RESERVED	AF21	M[15]_RXD[1]	AG21	RESERVED
AB1	RESERVED	AJ19	M[14]_RXD[1]	AH21	M[15]_CRS_DV
AB2	RESERVED	AF18	M[13]_RXD[1]	AF19	M[14]_CRS_DV
AB3	RESERVED	AJ17	M[12]_RXD[1]	AF17	M[13]_CRS_DV

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AC1	RESERVED	AJ15	M[11]_RXD[1]	AG17	M[12]_CRS_DV
AC2	RESERVED	AF15	M[10]_RXD[1]	AG15	M[11]_CRS_DV
AC3	RESERVED	AJ13	M[9]_RXD[1]	AF14	M[10]_CRS_DV
AD1	RESERVED	AF12	M[8]_RXD[1]	AG13	M[9]_CRS_DV
AD2	RESERVED	AJ11	M[7]_RXD[1]	AF11	M[8]_CRS_DV
AD3	RESERVED	AJ9	M[6]_RXD[1]	AG11	M[7]_CRS_DV
N3	RESERVED	AF9	M[5]_RXD[1]	AG9	M[6]_CRS_DV
N2	RESERVED	AJ7	M[4]_RXD[1]	AF8	M[5]_CRS_DV
N1	RESERVED	AF6	M[3]_RXD[1]	AG7	M[4]_CRS_DV
P3	RESERVED	AJ5	M[2]_RXD[1]	AF5	M[3]_CRS_DV
P2	RESERVED	AJ3	M[1]_RXD[1]	AG5	M[2]_CRS_DV
P1	RESERVED	AF1	M[0]_RXD[1]	AH3	M[1]_CRS_DV
R5	RESERVED	AC28	RESERVED	AF3	M[0]_CRS_DV
R4	RESERVED	AF28	RESERVED	AD29	RESERVED
R3	RESERVED	AH27	RESERVED	AG28	RESERVED
R2	RESERVED	AE27	RESERVED	AJ26	RESERVED
R1	RESERVED	AH25	RESERVED	AE26	RESERVED
T5	RESERVED	AE24	RESERVED	AJ24	RESERVED
T4	RESERVED	AF22	RESERVED	AE23	RESERVED
T3	RESERVED	AF20	RESERVED	AJ22	RESERVED
T2	RESERVED	AE21	M[15]_RXD[0]	AJ20	RESERVED
T1	RESERVED	AH19	M[14]_RXD[0]	AE20	M[15]_TXEN
W3	RESERVED	AH20	M[13]_RXD[0]	AJ18	M[14]_TXEN
W2	RESERVED	AH17	M[12]_RXD[0]	AJ21	M[13]_TXEN
V1	RESERVED	AH15	M[11]_RXD[0]	AJ16	M[12]_TXEN
G1	RESERVED	AE15	M[10]_RXD[0]	AJ14	M[11]_TXEN
V3	RESERVED	AH13	M[9]_RXD[0]	AE14	M[10]_TXEN
P4	RESERVED	AE12	M[8]_RXD[0]	AJ12	M[9]_TXEN
P5	RESERVED	AH11	M[7]_RXD[0]	AE11	M[8]_TXEN
V2	RESERVED	AH9	M[6]_RXD[0]	AJ10	M[7]_TXEN
U1	RESERVED	AE9	M[5]_RXD[0]	AJ8	M[6]_TXEN

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AE8	M[5]_TXEN	AH8	M[6]_TXD[0]	G27	RESERVED
AJ6	M[4]_TXEN	AF7	M[5]_TXD[0]	H29	RESERVED
AE5	M[3]_TXEN	AH6	M[4]_TXD[0]	H28	RESERVED
AJ4	M[2]_TXEN	AF4	M[3]_TXD[0]	H27	RESERVED
AG1	M[1]_TXEN	AH4	M[2]_TXD[0]	J29	RESERVED
AE1	M[0]_TXEN	AG2	M[1]_TXD[0]	J28	RESERVED
AD27	RESERVED	AE2	M[0]_TXD[0]	J27	RESERVED
AH28	RESERVED	U26	RESERVED	K29	RESERVED
AG26	RESERVED	U25	RESERVED	K28	RESERVED
AE25	RESERVED	V26	RESERVED	K27	RESERVED
AG24	RESERVED	V25	RESERVED	L29	RESERVED
AE22	RESERVED	W26	RESERVED	L28	RESERVED
AJ23	RESERVED	W25	RESERVED	L27	RESERVED
AG20	RESERVED	Y27	RESERVED	M29	RESERVED
AE18	M[15]_TXD[1]	Y26	RESERVED	M28	RESERVED
AG18	M[14]_TXD[1]	AA26	RESERVED	M27	RESERVED
AE16	M[13]_TXD[1]	AA25	RESERVED	G26	RESERVED
AG16	M[12]_TXD[1]	AB26	RESERVED	G25	RESERVED
AG14	M[11]_TXD[1]	AB25	RESERVED	H26	RESERVED
AE13	M[10]_TXD[1]	AC26	RESERVED	H25	RESERVED
AG12	M[9]_TXD[1]	AC25	RESERVED	J26	RESERVED
AE10	M[8]_TXD[1]	AD26	RESERVED	J25	RESERVED
AG10	M[7]_TXD[1]	AD25	RESERVED	K25	RESERVED
AG8	M[6]_TXD[1]	U27	RESERVED	K26	RESERVED
AE7	M[5]_TXD[1]	V29	RESERVED	M25	RESERVED
AG6	M[4]_TXD[1]	V28	RESERVED	L26	RESERVED
AE4	M[3]_TXD[1]	V27	RESERVED	M26	RESERVED
AG4	M[2]_TXD[1]	W29	RESERVED	L25	RESERVED
AG3	M[1]_TXD[1]	W28	RESERVED	N26	RESERVED
AE3	M[0]_TXD[1]	W27	RESERVED	N25	RESERVED
AD28	RESERVED	Y29	RESERVED	P26	RESERVED

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AG29	RESERVED	Y28	RESERVED	P25	RESERVED
AH26	RESERVED	Y25	RESERVED	F28	RESERVED
AF25	RESERVED	AA29	RESERVED	G28	RESERVED
AH24	RESERVED	AA28	RESERVED	E25	RESERVED
AG22	RESERVED	AA27	RESERVED	G29	RESERVED
AH22	RESERVED	AB29	RESERVED	F29	RESERVED
AE17	RESERVED	AB28	RESERVED	F26	RESERVED
AG19	M[15]_TXD[0]	AB27	RESERVED	E26	RESERVED
AH18	M[14]_TXD[0]	R26	RESERVED	F25	RESERVED
AF16	M[13]_TXD[0]	T25	RESERVED	E24	BIST_DONE/TSTOUT[15]
AH16	M[12]_TXD[0]	T26	RESERVED	D24	BIST_IN_PRC/TSTOUT[14]
AH14	M[11]_TXD[0]	T28	RESERVED	D25	MCT_ERR/TSTOUT[13]
AF13	M[10]_TXD[0]	U28	RESERVED	D26	FCB_ERR/TSTOUT[12]
AH12	M[9]_TXD[0]	R25	RESERVED	C26	CHECKSUM_OK/TSTOUT[11]
AF10	M[8]_TXD[0]	U29	RESERVED	D27	INIT_START/TSTOUT[10]
AH10	M[7]_TXD[0]	T29	RESERVED	C27	INIT_DONE/TSTOUT[9]
B27	G2_LINK#/TSTOUT[8]	U18	VSS	N12	VSS
A27	G2_DPCOL#/TSTOUT[7]	V12	VSS	N13	VSS
E28	G2_RXTX#/TSTOUT[6]	V13	VSS	K17	VDD
D28	G1_LINK#/TSTOUT[5]	V14	VSS	K18	VDD
C28	G1_DPCOL#/TSTOUT[4]	V15	VSS	M10	VDD
B28	G1_RXTX#/TSTOUT[3]	V16	VSS	N10	VDD
E29	LED_BIT/TSTOUT[2]	V17	VSS	M20	VDD
D29	LED_SYN/TSTOUT[1]	V18	VSS	N20	VDD
C29	LED_CLK/TSTOUT[0]	N14	VSS	U10	VDD
N29	RESERVED	N15	VSS	V10	VDD
P29	RESERVED	N16	VSS	U20	VDD

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
F3	SCAN_EN	N17	VSS	V20	VDD
E1	SCLK	N18	VSS	Y12	VDD
U3	T_MODE0	P12	VSS	Y13	VDD
C10	T_MODE1	P13	VSS	Y17	VDD
B24	RESERVED	P14	VSS	Y18	VDD
A21	TRUNK1	P15	VSS	K12	VDD
C22	TRUNK0	P16	VSS	K13	VDD
A26	STROBE	C19	OE_CLK2	M16	VSS
B26	D0	B19	OE_CLK1	M17	VSS
C25	AUTOFD	A19	OE_CLK0	M18	VSS
A24	SCL	R13	VSS	F16	VCC
A25	SDA	R14	VSS	F17	VCC
F1	AVCC	R15	VSS	N6	VCC
D1	AGND	R16	VSS	P6	VCC
D22	SCANCOL	R17	VSS	R6	VCC
E23	SCANLINK	R18	VSS	T6	VCC
E27	SCANMODE	T12	VSS	U6	VCC
N28		T13	VSS	N24	VCC
N27		T14	VSS	P24	VCC
F2	RESIN#	T15	VSS	R24	VCC
G2	RESETOUT#	T16	VSS	T24	VCC
B22	Reserved	T17	VSS	U24	VCC
A22	Reserved	T18	VSS	AD13	VCC
C23	Reserved	U12	VSS	AD14	VCC
B23	Reserved	U13	VSS	AD15	VCC
A23	Reserved	U14	VSS	AD16	VCC
C24	RESERVED	U15	VSS	AD17	VCC
D23	SCANCLK	U16	VSS	F13	VCC
T27	RESERVED	U17	VSS	F14	VCC
F27	RESERVED	M12	VSS	F15	VCC
C20	LA_CLK2	M13	VSS		



Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
B20	LA_CLK1	M14	VSS		
A20	LA_CLK0	M15	VSS		
C21	P_D	P17	VSS		
E20	RESERVED	P18	VSS		
B25	RESERVED	R12	VSS		

#### 14.4 Ball – Signal Name in Managed Mode

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
D20	LA_D[63]	D3	LA_D[19]	A9	LA_OE0#
B21	LA_D[62]	E3	LA_D[18]	B9	LA_OE1#
D19	LA_D[61]	D2	LA_D[17]	F4	RESERVED
E19	LA_D[60]	E2	LA_D[16]	F5	RESERVED
D18	LA_D[59]	A7	LA_D[15]	G4	RESERVED
E18	LA_D[58]	B7	LA_D[14]	G5	RESERVED
D17	LA_D[57]	A6	LA_D[13]	H4	RESERVED
E17	LA_D[56]	B6	LA_D[12]	H5	RESERVED
D16	LA_D[55]	C6	LA_D[11]	J4	RESERVED
E16	LA_D[54]	A5	LA_D[10]	J5	RESERVED
D15	LA_D[53]	B5	LA_D[9]	K4	RESERVED
E15	LA_D[52]	C5	LA_D[8]	K5	RESERVED
D14	LA_D[51]	A4	LA_D[7]	L4	RESERVED
E14	LA_D[50]	B4	LA_D[6]	L5	RESERVED
D13	LA_D[49]	C4	LA_D[5]	M4	RESERVED
E13	LA_D[48]	A3	LA_D[4]	M5	RESERVED
D21	LA_D[47]	B3	LA_D[3]	N4	RESERVED
E21	LA_D[46]	C3	LA_D[2]	N5	RESERVED
A18	LA_D[45]	B2	LA_D[1]	G3	RESERVED
B18	LA_D[44]	C2	LA_D[0]	H1	RESERVED
C18	LA_D[43]	C14	LA_A[20]	H2	RESERVED

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
A17	LA_D[42]	A13	LA_A[19]	H3	RESERVED
B17	LA_D[41]	B13	LA_A[18]	J1	RESERVED
C17	LA_D[40]	C13	LA_A[17]	J2	RESERVED
A16	LA_D[39]	A12	LA_A[16]	J3	RESERVED
B16	LA_D[38]	B12	LA_A[15]	K1	RESERVED
C16	LA_D[37]	C12	LA_A[14]	K2	RESERVED
A15	LA_D[36]	A11	LA_A[13]	K3	RESERVED
B15	LA_D[35]	B11	LA_A[12]	L1	RESERVED
C15	LA_D[34]	C11	LA_A[11]	L2	RESERVED
A14	LA_D[33]	D11	LA_A[10]	L3	RESERVED
B14	LA_D[32]	E11	LA_A[9]	M1	RESERVED
D9	LA_D[31]	A10	LA_A[8]	M2	RESERVED
E9	LA_D[30]	B10	LA_A[7]	M3	RESERVED
D8	LA_D[29]	D10	LA_A[6]	U4	RESERVED
E8	LA_D[28]	E10	LA_A[5]	U5	RESERVED
D7	LA_D[27]	A8	LA_A[4]	V4	RESERVED
E7	LA_D[26]	C7	LA_A[3]	V5	RESERVED
D6	LA_D[25]	B8	LA_DSC#	W4	RESERVED
E6	LA_D[24]	C1	LA_CLK	W5	RESERVED
D5	LA_D[23]	C9	LA_WE#	Y4	RESERVED
E5	LA_D[22]	D12	LA_WE0#	Y5	RESERVED
D4	LA_D[21]	E12	LA_WE1#	AA4	RESERVED
E4	LA_D[20]	C8	LA_OE#	AA5	RESERVED
AB4	RESERVED	U2	RESERVED	AH7	M[4]_RXD[0]
AB5	RESERVED	R28	MDC	AE6	M[3]_RXD[0]
AC4	RESERVED	P28	MDIO	AH5	M[2]_RXD[0]
AC5	RESERVED	R29	M_CLK	AH2	M[1]_RXD[0]
AD4	RESERVED	AC29	RESERVED	AF2	M[0]_RXD[0]
AD5	RESERVED	AE28	RESERVED	AC27	RESERVED
W1	RESERVED	AJ27	RESERVED	AF29	RESERVED
Y1	RESERVED	AF27	RESERVED	AG27	RESERVED

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
Y2	RESERVED	AJ25	RESERVED	AF26	RESERVED
Y3	RESERVED	AF24	RESERVED	AG25	RESERVED
AA1	RESERVED	AH23	RESERVED	AG23	RESERVED
AA2	RESERVED	AE19	RESERVED	AF23	RESERVED
AA3	RESERVED	AF21	M[15]_RXD[1]	AG21	RESERVED
AB1	RESERVED	AJ19	M[14]_RXD[1]	AH21	M[15]_CRS_DV
AB2	RESERVED	AF18	M[13]_RXD[1]	AF19	M[14]_CRS_DV
AB3	RESERVED	AJ17	M[12]_RXD[1]	AF17	M[13]_CRS_DV
AC1	RESERVED	AJ15	M[11]_RXD[1]	AG17	M[12]_CRS_DV
AC2	RESERVED	AF15	M[10]_RXD[1]	AG15	M[11]_CRS_DV
AC3	RESERVED	AJ13	M[9]_RXD[1]	AF14	M[10]_CRS_DV
AD1	RESERVED	AF12	M[8]_RXD[1]	AG13	M[9]_CRS_DV
AD2	RESERVED	AJ11	M[7]_RXD[1]	AF11	M[8]_CRS_DV
AD3	RESERVED	AJ9	M[6]_RXD[1]	AG11	M[7]_CRS_DV
N3	RESERVED	AF9	M[5]_RXD[1]	AG9	M[6]_CRS_DV
N2	RESERVED	AJ7	M[4]_RXD[1]	AF8	M[5]_CRS_DV
N1	RESERVED	AF6	M[3]_RXD[1]	AG7	M[4]_CRS_DV
P3	RESERVED	AJ5	M[2]_RXD[1]	AF5	M[3]_CRS_DV
P2	RESERVED	AJ3	M[1]_RXD[1]	AG5	M[2]_CRS_DV
P1	RESERVED	AF1	M[0]_RXD[1]	AH3	M[1]_CRS_DV
R5	RESERVED	AC28	RESERVED	AF3	M[0]_CRS_DV
R4	RESERVED	AF28	RESERVED	AD29	RESERVED
R3	RESERVED	AH27	RESERVED	AG28	RESERVED
R2	RESERVED	AE27	RESERVED	AJ26	RESERVED
R1	RESERVED	AH25	RESERVED	AE26	RESERVED
T5	RESERVED	AE24	RESERVED	AJ24	RESERVED
T4	RESERVED	AF22	RESERVED	AE23	RESERVED
T3	RESERVED	AF20	RESERVED	AJ22	RESERVED
T2	RESERVED	AE21	M[15]_RXD[0]	AJ20	RESERVED
T1	RESERVED	AH19	M[14]_RXD[0]	AE20	M[15]_TXEN
W3	RESERVED	AH20	M[13]_RXD[0]	AJ18	M[14]_TXEN

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
W2	RESERVED	AH17	M[12]_RXD[0]	AJ21	M[13]_TXEN
V1	RESERVED	AH15	M[11]_RXD[0]	AJ16	M[12]_TXEN
G1	RESERVED	AE15	M[10]_RXD[0]	AJ14	M[11]_TXEN
V3	RESERVED	AH13	M[9]_RXD[0]	AE14	M[10]_TXEN
P4	RESERVED	AE12	M[8]_RXD[0]	AJ12	M[9]_TXEN
P5	RESERVED	AH11	M[7]_RXD[0]	AE11	M[8]_TXEN
V2	RESERVED	AH9	M[6]_RXD[0]	AJ10	M[7]_TXEN
U1	RESERVED	AE9	M[5]_RXD[0]	AJ8	M[6]_TXEN
AE8	M[5]_TXEN	AH8	M[6]_TXD[0]	G27	RESERVED
AJ6	M[4]_TXEN	AF7	M[5]_TXD[0]	H29	RESERVED
AE5	M[3]_TXEN	AH6	M[4]_TXD[0]	H28	RESERVED
AJ4	M[2]_TXEN	AF4	M[3]_TXD[0]	H27	RESERVED
AG1	M[1]_TXEN	AH4	M[2]_TXD[0]	J29	RESERVED
AE1	M[0]_TXEN	AG2	M[1]_TXD[0]	J28	RESERVED
AD27	RESERVED	AE2	M[0]_TXD[0]	J27	RESERVED
AH28	RESERVED	U26	RESERVED	K29	RESERVED
AG26	RESERVED	U25	RESERVED	K28	RESERVED
AE25	RESERVED	V26	RESERVED	K27	RESERVED
AG24	RESERVED	V25	RESERVED	L29	RESERVED
AE22	RESERVED	W26	RESERVED	L28	RESERVED
AJ23	RESERVED	W25	RESERVED	L27	RESERVED
AG20	RESERVED	Y27	RESERVED	M29	RESERVED
AE18	M[15]_TXD[1]	Y26	RESERVED	M28	RESERVED
AG18	M[14]_TXD[1]	AA26	RESERVED	M27	RESERVED
AE16	M[13]_TXD[1]	AA25	RESERVED	G26	RESERVED
AG16	M[12]_TXD[1]	AB26	RESERVED	G25	RESERVED
AG14	M[11]_TXD[1]	AB25	RESERVED	H26	RESERVED
AE13	M[10]_TXD[1]	AC26	RESERVED	H25	RESERVED
AG12	M[9]_TXD[1]	AC25	RESERVED	J26	RESERVED
AE10	M[8]_TXD[1]	AD26	RESERVED	J25	RESERVED
AG10	M[7]_TXD[1]	AD25	RESERVED	K25	RESERVED

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AG8	M[6]_TXD[1]	U27	RESERVED	K26	RESERVED
AE7	M[5]_TXD[1]	V29	RESERVED	M25	RESERVED
AG6	M[4]_TXD[1]	V28	RESERVED	L26	RESERVED
AE4	M[3]_TXD[1]	V27	RESERVED	M26	RESERVED
AG4	M[2]_TXD[1]	W29	RESERVED	L25	RESERVED
AG3	M[1]_TXD[1]	W28	RESERVED	N26	RESERVED
AE3	M[0]_TXD[1]	W27	RESERVED	N25	RESERVED
AD28	RESERVED	Y29	RESERVED	P26	RESERVED
AG29	RESERVED	Y28	RESERVED	P25	RESERVED
AH26	RESERVED	Y25	RESERVED	F28	RESERVED
AF25	RESERVED	AA29	RESERVED	G28	RESERVED
AH24	RESERVED	AA28	RESERVED	E25	RESERVED
AG22	RESERVED	AA27	RESERVED	G29	RESERVED
AH22	RESERVED	AB29	RESERVED	F29	RESERVED
AE17	RESERVED	AB28	RESERVED	F26	RESERVED
AG19	M[15]_TXD[0]	AB27	RESERVED	E26	RESERVED
AH18	M[14]_TXD[0]	R26	RESERVED	F25	RESERVED
AF16	M[13]_TXD[0]	T25	RESERVED	E24	BIST_DONE/TSTOUT[15]
AH16	M[12]_TXD[0]	T26	RESERVED	D24	BIST_IN_PRC/TSTOUT[14]
AH14	M[11]_TXD[0]	T28	RESERVED	D25	MCT_ERR/TSTOUT[13]
AF13	M[10]_TXD[0]	U28	RESERVED	D26	FCB_ERR/TSTOUT[12]
AH12	M[9]_TXD[0]	R25	RESERVED	C26	CHECKSUM_OK/TSTOUT[11]
AF10	M[8]_TXD[0]	U29	RESERVED	D27	INIT_START/TSTOUT[10]
AH10	M[7]_TXD[0]	T29	RESERVED	C27	INIT_DONE/TSTOUT[9]
B27	G2_LINK#/TSTOUT[8]	U18	VSS	N12	VSS
A27	G2_DPCOL#/TSTOUT[7]	V12	VSS	N13	VSS
E28	G2_RXTX#/TSTOUT[6]	V13	VSS	K17	VDD
D28	G1_LINK#/TSTOUT[5]	V14	VSS	K18	VDD

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
C28	G1_DPCOL#/TSTOUT[4]	V15	VSS	M10	VDD
B28	G1_RXTX#/TSTOUT[3]	V16	VSS	N10	VDD
E29	LED_BIT/TSTOUT[2]	V17	VSS	M20	VDD
D29	LED_SYN/TSTOUT[1]	V18	VSS	N20	VDD
C29	LED_CLK/TSTOUT[0]	N14	VSS	U10	VDD
N29	RESERVED	N15	VSS	V10	VDD
P29	RESERVED	C19	P_DATA15	U20	VDD
F3	SCAN_EN	B19	P_DATA14	V20	VDD
E1	SCLK	A19	P_DATA13	Y12	VDD
U3	T_MODE0	P12	VSS	Y13	VDD
C10	T_MODE1	P13	VSS	Y17	VDD
B24	P_DATA6	P14	VSS	Y18	VDD
A21	P_DATA7	P15	VSS	K12	VDD
C22	P_A2	P16	VSS	K13	VDD
A26	P_WE	N16	VSS	M16	VSS
B26	P_RD	N17	VSS	M17	VSS
C25	P_CS	N18	VSS	M18	VSS
A24	P_A1	R13	VSS	F16	VCC
A25	P_A0	R14	VSS	F17	VCC
F1	AVCC	R15	VSS	N6	VCC
D1	AGND	R16	VSS	P6	VCC
D22	SCANCOL	R17	VSS	R6	VCC
E23	SCANLINK	R18	VSS	T6	VCC
E27	SCANMODE	T12	VSS	U6	VCC
N28		T13	VSS	N24	VCC
N27		T14	VSS	P24	VCC
F2	RESIN#	T15	VSS	R24	VCC
G2	RESETOUT#	T16	VSS	T24	VCC
B22	P_DATA5	T17	VSS	U24	VCC
A22	P_DATA4	T18	VSS	AD13	VCC
C23	P_DATA3	U12	VSS	AD14	VCC

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
B23	P_DATA2	U13	VSS	AD15	VCC
A23	P_DATA1	U14	VSS	AD16	VCC
C24	P_DATA0	U15	VSS	AD17	VCC
D23	SCANCLK	U16	VSS	F13	VCC
T27	RESERVED	U17	VSS	F14	VCC
F27	RESERVED	M12	VSS	F15	VCC
C20	P_DATA12	M13	VSS		
B20	P_DATA11	M14	VSS		
A20	P_DATA10	M15	VSS		
C21	P_DATA9	P17	VSS		
E20	P_DATA8	P18	VSS		
B25	P_INT	R12	VSS		

## 14.5 AC/DC Timing

### 14.5.1 Absolute Maximum Ratings

Storage Temperature	-40C to +85°C
Operating Temperature	-40C to +85°C
Maximum Junction Temperature	+125°C
Supply Voltage VCC with Respect to $V_{SS}$	+3.0V to +3.6 V
Supply Voltage VDD with Respect to $V_{SS}$	+2.38V to +2.75V
Voltage on 5V Tolerant Input Pins	-0.5V to ( $V_{DD2_1} + 3.3V$ )
Voltage on 5V Tolerant Input Pins	-0.5V to ( $V_{DD2_2} + 2.5V$ )
Voltage on Other Pins	-0.5V to ( $V_{DD2} + 0.3V$ )

**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.

### 14.5.2 DC Electrical Characteristics

VCC = 3.0 V to 3.6 V (3.3v +/- 10%)  $T_{AMBIENT} = -40C$  to +85C

VDD = 2.5V +10% - 5%

### 14.5.3 Recommended Operating Conditions

Symbol	Parameter Description				Unit
		Min	Typ	Max	
$f_{osc}$	Frequency of Operation ( -50)		100		MHz
$I_{CC}$	Supply Current – @ 100 MHz ( $V_{CC}=3.3\text{ V}$ )			250	mA
$I_{DD}$	Supply Current – @ 100 MHz ( $V_{DD}=2.5\text{ V}$ )			1350	mA
$V_{OH}$	Output High Voltage (CMOS)	2.4			V
$V_{OL}$	Output Low Voltage (CMOS)			0.4	V
$V_{IH-TTL}$	Input High Voltage (TTL 5V tolerant)	2.0		$V_{CC} + 2.0$	V
$V_{IL-TTL}$	Input Low Voltage (TTL 5V tolerant)			0.8	V
$I_{IL}$	Input Leakage Current ( $0.1\text{ V} < V_{IN} < V_{CC}$ ) (all pins except those with internal pull-up/pull-down resistors)			10	$\mu\text{A}$
$I_{OL}$	Output Leakage Current ( $0.1\text{ V} < V_{OUT} < V_{CC}$ )			10	$\mu\text{A}$
$C_{IN}$	Input Capacitance			5	pF
$C_{OUT}$	Output Capacitance			5	pF
$C_{I/O}$	I/O Capacitance			7	pF
$\theta_{ja}$	Thermal resistance with 0 air flow			11.2	C/W
$\theta_{ja}$	Thermal resistance with 1 m/s air flow			10.2	C/W
$\theta_{ja}$	Thermal resistance with 2m/s air flow			8.9	C/W
$\theta_{jc}$	Thermal resistance between junction and case			3.1	C/W
$\theta_{jb}$	Thermal resistance between junction and board			6.6	C/W



14.5.4 Typical Reset & Bootstrap Timing Diagram

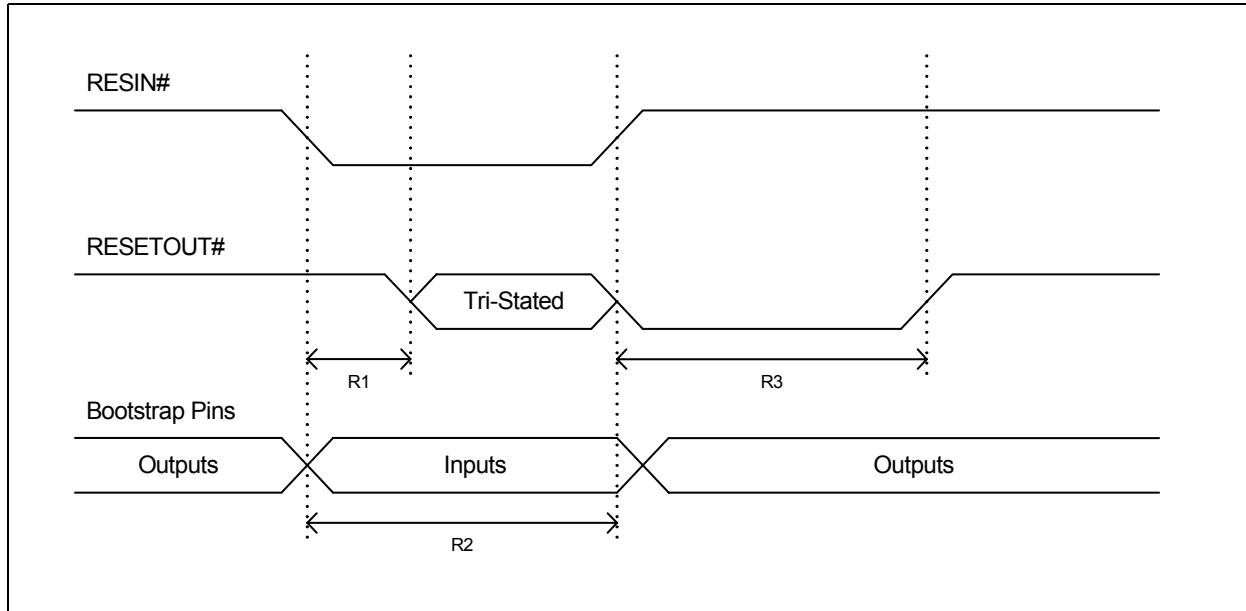


Figure 16 - Typical Reset & Bootstrap Timing Diagram

Symbol	Parameter	Min	Typ	Note:
R1	Delay until RESETOUT# is tri-stated		10ns	RESETOUT# state is then determined by the external pull-up/down resistor
R2	Bootstrap stabilization	1μs	10μs	Bootstrap pins sampled on rising edge of RESIN# <sup>1</sup>
R3	RESETOUT# assertion		2ms	

Table 15 - Reset & Bootstrap Timing

1. The TSTOUT[8:0] pins will switch over to the LED interface functionality in 3 SCLK cycles after RESIN# goes high

14.5.5 Typical CPU Timing Diagram for a CPU Write Cycle

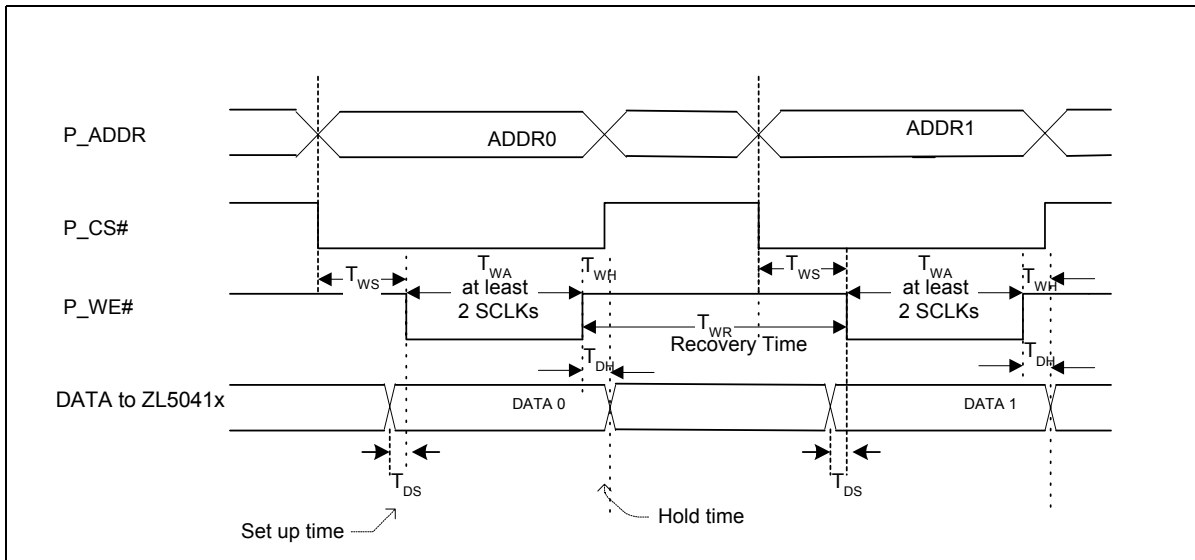


Figure 17 - Typical CPU Timing Diagram for a CPU Write Cycle

Description		(SCLK=100Mhz)		(SCLK=125Mhz)		Refer to Figure
Write Cycle	Symbol	Min	Max	Min	Max	
Write Set up Time	T <sub>WS</sub>	10		10		
Write Active Time	T <sub>WA</sub>	20		16		At least 2 SCLK
Write Hold Time	T <sub>WH</sub>	2		2		
Write Recovery time	T <sub>WR</sub>	30		24		At least 3 SCLK
Data Set Up time	T <sub>DS</sub>	10		10		
Data Hold time	T <sub>DH</sub>	2		2		

Table 16 - Write Cycle

14.5.6 Typical CPU Timing Diagram for a CPU Read Cycle

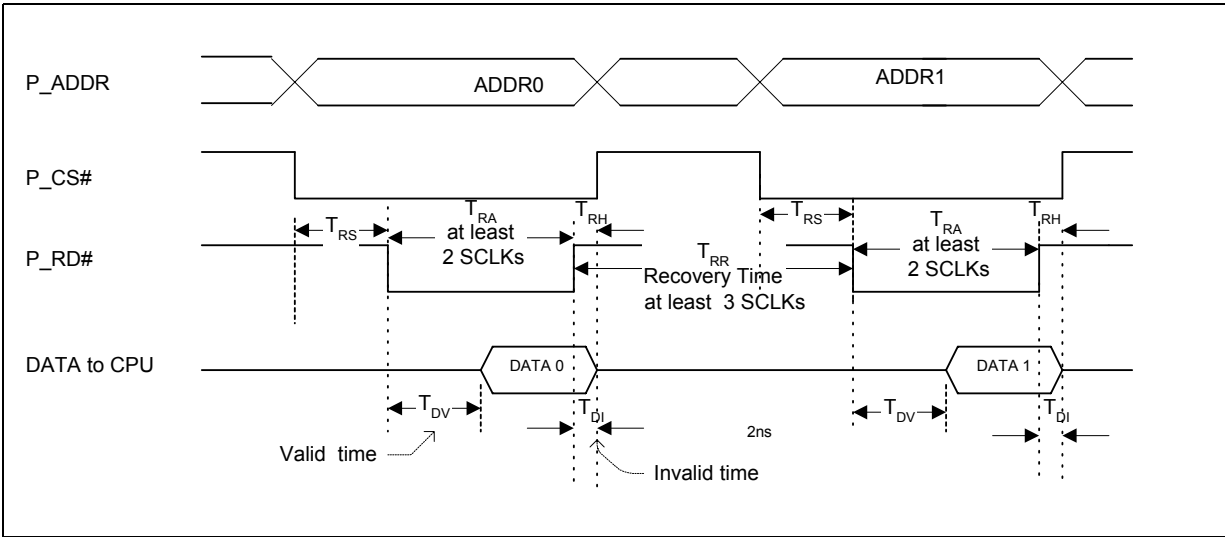


Figure 18 - Typical CPU Timing Diagram for a CPU Read Cycle

Description		(SCLK=100Mhz)		(SCLK=125Mhz)		Refer to Figure
Read Cycle	Symbol	Min	Max	Min	Max	
Read Set up Time	$T_{RS}$	10		10		
Read Active Time	$T_{RA}$	20		16		At least 2 SCLK
Read Hold Time	$T_{RH}$	2		2		
Read Recovery time	$T_{RR}$	30		24		At least 3 SCLK
Data Valid time	$T_{DV}$		10		10	
Data Invalid time	$T_{DI}$		6		6	

Table 17 - Read Cycle

14.6 Local Frame Buffer SBRAM Memory Interface

14.6.1 Local SBRAM Memory Interface:

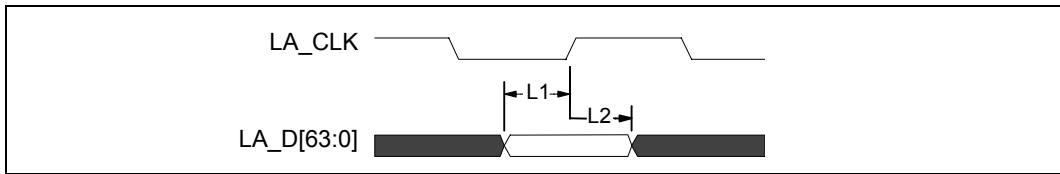


Figure 19 - Local Memory Interface – Input Setup and Hold Timing

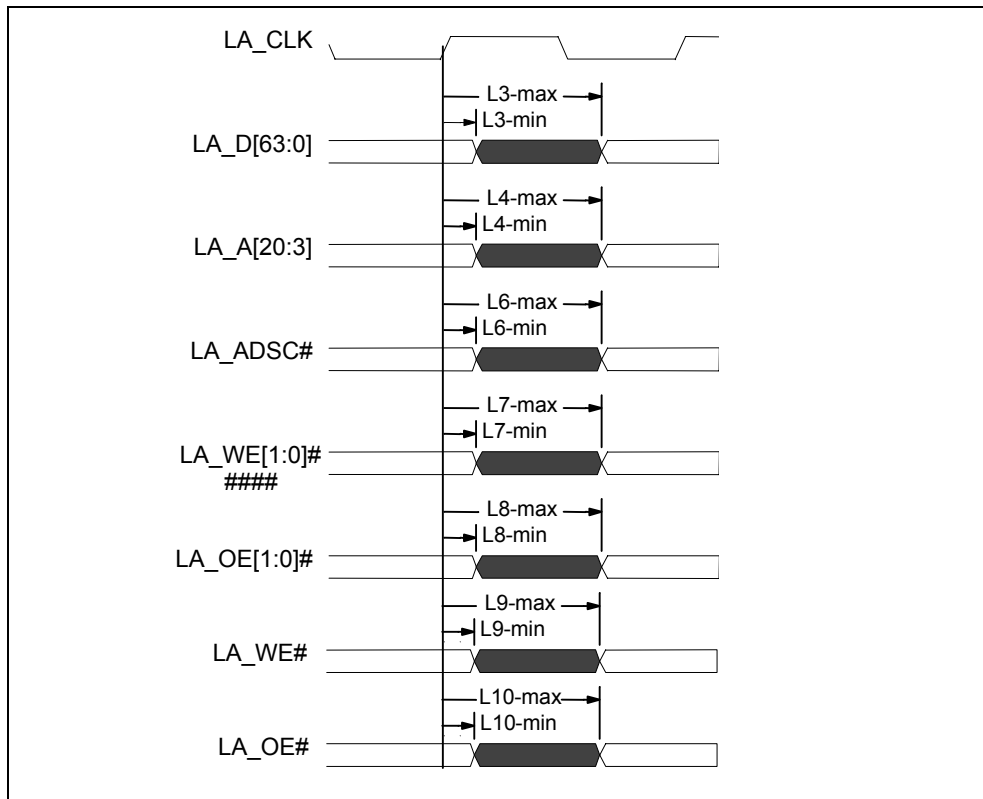


Figure 20 - Local Memory Interface - Output Valid Delay Timing

Symbol	Parameter	-100MHz		Note:
		Min (ns)	Max (ns)	
L1	LA_D[63:0] input set-up time	4		
L2	LA_D[63:0] input hold time	1.5		
L3	LA_D[63:0] output valid delay	1.5	7	C <sub>L</sub> = 25pf
L4	LA_A[20:3] output valid delay	2	7	C <sub>L</sub> = 30pf
L6	LA_ADSC# output valid delay	1	7	C <sub>L</sub> = 30pf
L7	LA_WE[1:0]#output valid delay	1	7	C <sub>L</sub> = 25pf
L8	LA_OE[1:0]# output valid delay	-1	1	C <sub>L</sub> = 25pf
L9	LA_WE# output valid delay	1	7	C <sub>L</sub> = 25pf
L10	LA_OE# output valid delay	1	5	C <sub>L</sub> = 25pf

Table 18 - AC Characteristics – Local frame buffer SBRAM Memory Interface

14.7 AC Characteristics

14.7.1 Reduced Media Independent Interface

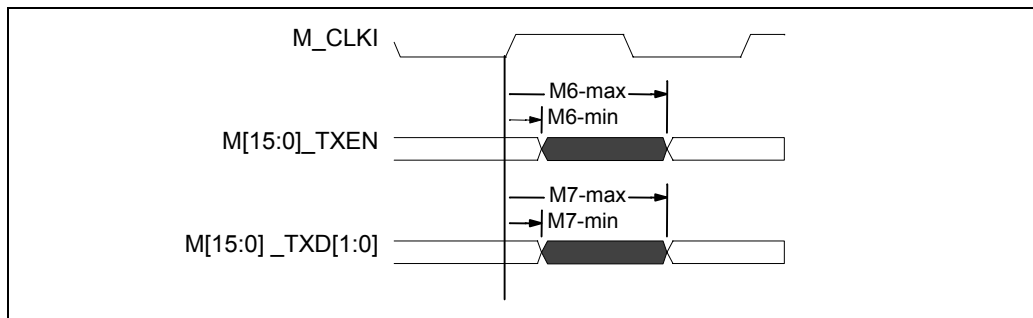


Figure 21 - AC Characteristics – Reduced media independent Interface

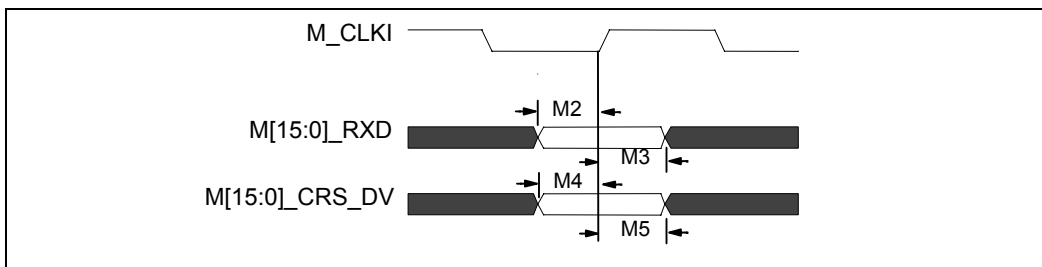
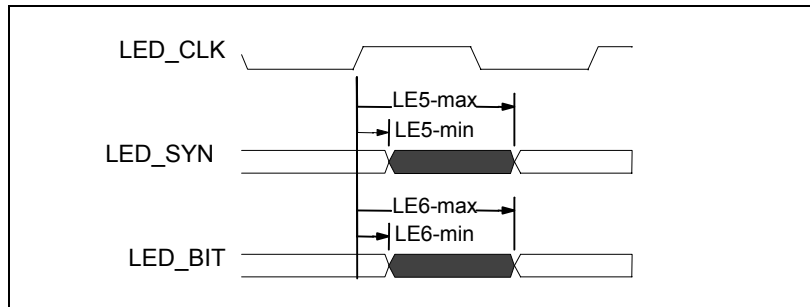


Figure 22 - AC Characteristics – Reduced Media Independent Interface

Symbol	Parameter	-50MHz		Note:
		Min (ns)	Max (ns)	
M2	M[15:0]_RXD[1:0] Input Setup Time	4		
M3	M[15:0]_RXD[1:0] Input Hold Time	1		
M4	M[15:0]_CRS_DV Input Setup Time	4		
M5	M[15:0]_CRS_DV Input Hold Time	1		
M6	M[15:0]_TXEN Output Delay Time	2	11	C <sub>L</sub> = 20 pF
M7	M[15:0]_TXD[1:0] Output Delay Time	2	11	C <sub>L</sub> = 20 pF

**Table 19 - AC Characteristics – Reduced Media Independent Interface**

**14.7.2 LED Interface**



**Figure 23 - AC Characteristics – LED Interface**

Symbol	Parameter	Variable FREQ.		Note:
		Min (ns)	Max (ns)	
LE5	LED_SYN Output Valid Delay	-1	7	C <sub>L</sub> = 30pf
LE6	LED_BIT Output Valid Delay	-1	7	C <sub>L</sub> = 30pf

**Table 20 - AC Characteristics – LED Interface**

14.7.3 SCANLINK SCANCOL Output Delay Timing

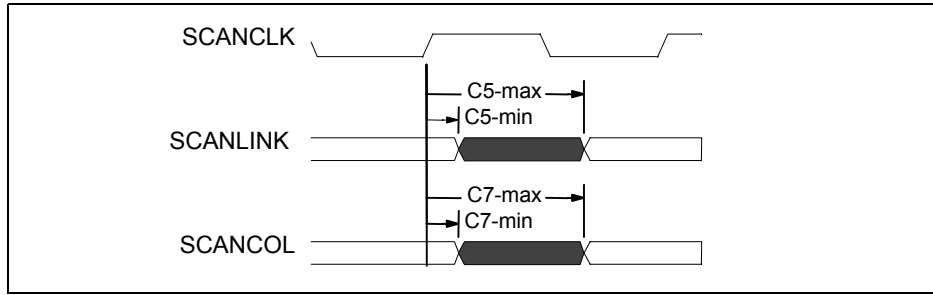


Figure 24 - SCANLINK SCANCOL Output Delay Timing

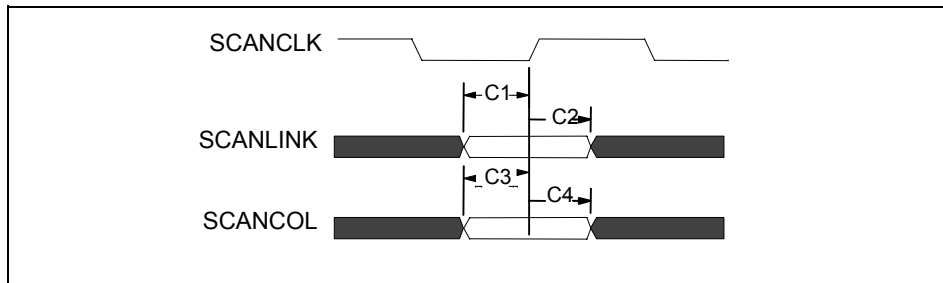


Figure 25 - SCANLINK, SCANCOL Setup Timing

Symbol	Parameter	-25MHz		Note:
		Min (ns)	Max (ns)	
C1	SCANLINK input set-up time	20		
C2	SCANLINK input hold time	2		
C3	SCANCOL input setup time	20		
C4	SCANCOL input hold time	1		
C5	SCANLINK output valid delay	0	10	$C_L = 30\text{pf}$
C7	SCANCOL output valid delay	0	10	$C_L = 30\text{pf}$

Table 21 - TSCANLINK, SCANCOL Timing

14.7.4 MDIO Input Setup and Hold Timing

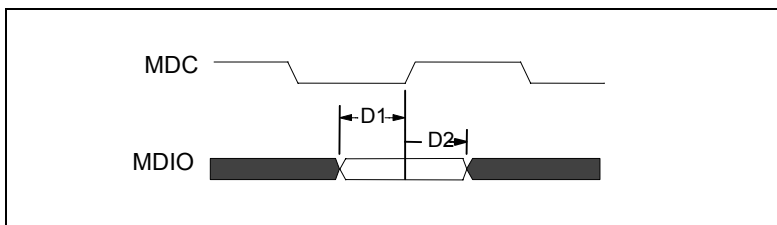


Figure 26 - MDIO Input Setup and Hold Timing

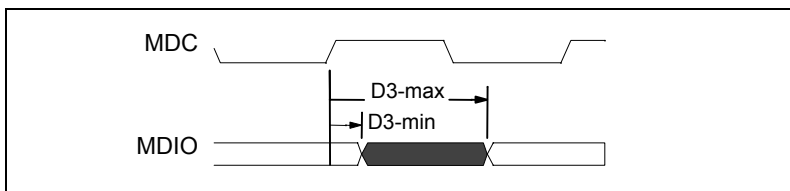


Figure 27 - MDIO Output Delay Timing

Symbol	Parameter	1MHz		Note:
		Min (ns)	Max (ns)	
D1	MDIO input setup time	10		
D2	MDIO input hold time	2		
D3	MDIO output delay time	1	20	C <sub>L</sub> = 50pf

Table 22 - MDIO Timing



14.7.5 I<sup>2</sup>C Input Setup Timing

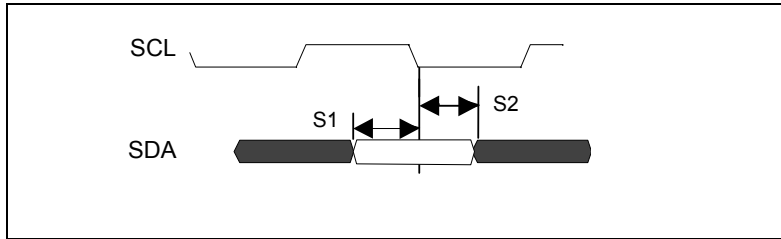


Figure 28 - I<sup>2</sup>C Input Setup Timing

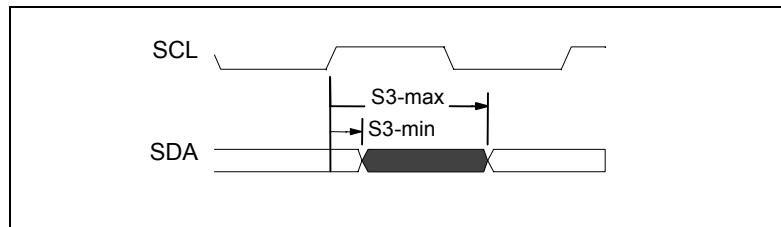


Figure 29 - I<sup>2</sup>C Output Delay Timing

Symbol	Parameter	50KHz		Note:
		Min (ns)	Max (ns)	
S1	SDA input setup time	20		
S2	SDA input hold time	1		
S3*	SDA output delay time	4 usec	6 usec	C <sub>L</sub> = 30pf
* Open Drain Output. Low to High transistor is controlled by external pullup resistor.				

Table 23 - I<sup>2</sup>C Timing

14.7.6 Serial Interface Setup Timing

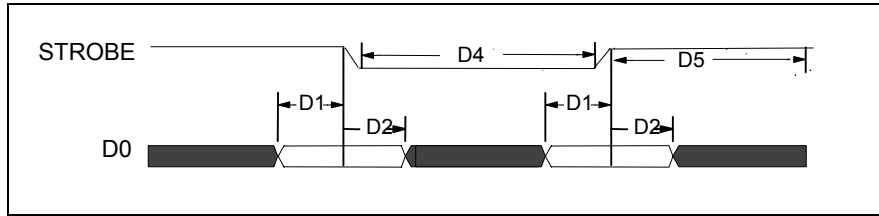


Figure 30 - Serial Interface Setup Timing

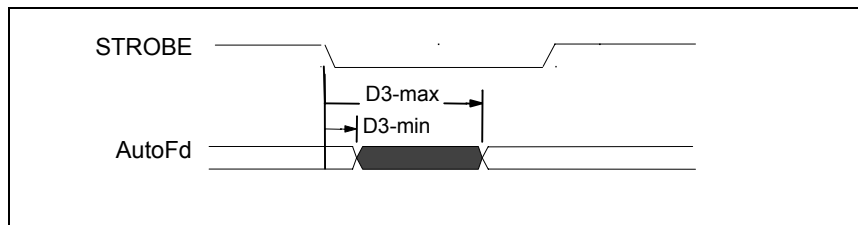
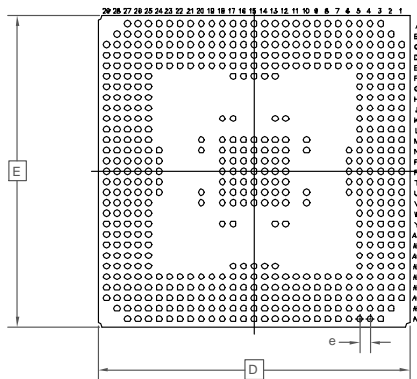
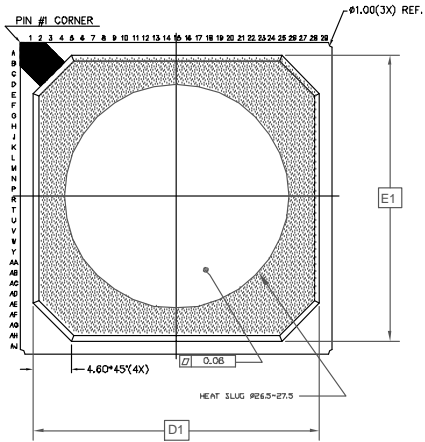


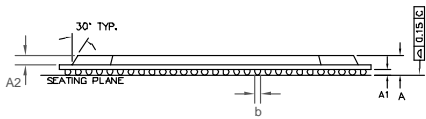
Figure 31 - Serial Interface Output Delay Timing

Symbol	Parameter	Min (ns)	Max (ns)	Note:
D1	D0 setup time	20		
D2	D0 hold time	3 $\mu$ s		
D3	AutoFd output delay time	1	50	C <sub>L</sub> = 100pf
D4	Strobe low time	5 $\mu$ s		
D5	Strobe high time	5 $\mu$ s		

Table 24 - Serial Interface Timing



DIMENSION	MIN	MAX
A	2.20	2.46
A1	0.50	0.70
A2	1.17 REF	
D	37.30	37.70
D1	34.50 REF	
E	37.30	37.70
E1	34.50 REF	
b	0.60	0.90
e	1.27	
N	553	
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

© Zarlink Semiconductor 2002 All rights reserved.					Package Code	GK
ISSUE	1				Previous package codes:	BH / G
ACN	213932				Package Outline for 553 Ball HSBGA (37.5x37.5x2.33mm)	
DATE	20Jan03					GPD00818
APPRD.						



**For more information about all Zarlink products  
visit our Web Site at  
[www.zarlink.com](http://www.zarlink.com)**

Information relating to products and services furnished herein by Zarlink Semiconductor Inc. trading as Zarlink Semiconductor or its subsidiaries (collectively "Zarlink") is believed to be reliable. However, Zarlink assumes no liability for errors that may appear in this publication, or for liability otherwise arising from the application or use of any such information, product or service or for any infringement of patents or other intellectual property rights owned by third parties which may result from such application or use. Neither the supply of such information or purchase of product or service conveys any license, either express or implied, under patents or other intellectual property rights owned by Zarlink or licensed from third parties by Zarlink, whatsoever. Purchasers of products are also hereby notified that the use of product in certain ways or in combination with Zarlink, or non-Zarlink furnished goods or services may infringe patents or other intellectual property rights owned by Zarlink.

This publication is issued to provide information only and (unless agreed by Zarlink in writing) may not be used, applied or reproduced for any purpose nor form part of any order or contract nor to be regarded as a representation relating to the products or services concerned. The products, their specifications, services and other information appearing in this publication are subject to change by Zarlink without notice. No warranty or guarantee express or implied is made regarding the capability, performance or suitability of any product or service. Information concerning possible methods of use is provided as a guide only and does not constitute any guarantee that such methods of use will be satisfactory in a specific piece of equipment. It is the user's responsibility to fully determine the performance and suitability of any equipment using such information and to ensure that any publication or data used is up to date and has not been superseded. Manufacturing does not necessarily include testing of all functions or parameters. These products are not suitable for use in any medical products whose failure to perform may result in significant injury or death to the user. All products and materials are sold and services provided subject to Zarlink's conditions of sale which are available on request.

Purchase of Zarlink's I<sup>2</sup>C components conveys a licence under the Philips I<sup>2</sup>C Patent rights to use these components in an I<sup>2</sup>C System, provided that the system conforms to the I<sup>2</sup>C Standard Specification as defined by Philips.

Zarlink, ZL and the Zarlink Semiconductor logo are trademarks of Zarlink Semiconductor Inc.

Copyright 2003, Zarlink Semiconductor Inc. All Rights Reserved.

**TECHNICAL DOCUMENTATION - NOT FOR RESALE**