

**Features**

April 2006

- Integrated Single-Chip 10/100M Ethernet Switch
  - Sixteen 10/100 Mbps auto-negotiating Fast Ethernet (FE) ports with RMIi or GPSI (7WS) interface options per port
- Supports one Frame Data Buffer (FDB) memory domains (1 MB or 2 MB) with pipelined, sync-burst SRAM at 100 MHz
  - Applies centralized shared memory architecture
- L2 Switching
  - MAC address self learning, up to 64K MAC addresses
  - Supports port-based and tagged-based VLAN (IEEE 802.1Q)
  - Supports up to 255 VLANs and IP multicast groups
  - VLAN tag insertion and stripping selectable on a per port, per VLAN basis
  - Supports spanning tree on per-system (IEEE 802.1D/w) or per-VLAN basis (IEEE 802.1s)
- Supports IP Multicast with IGMP snooping
- High performance packet classification and switching at full-wire speed
- CPU access supports the following interface options:
  - 8/16-bit ISA interface in managed mode
  - Serial interface in unmanaged mode, with optional I<sup>2</sup>C EEPROM support

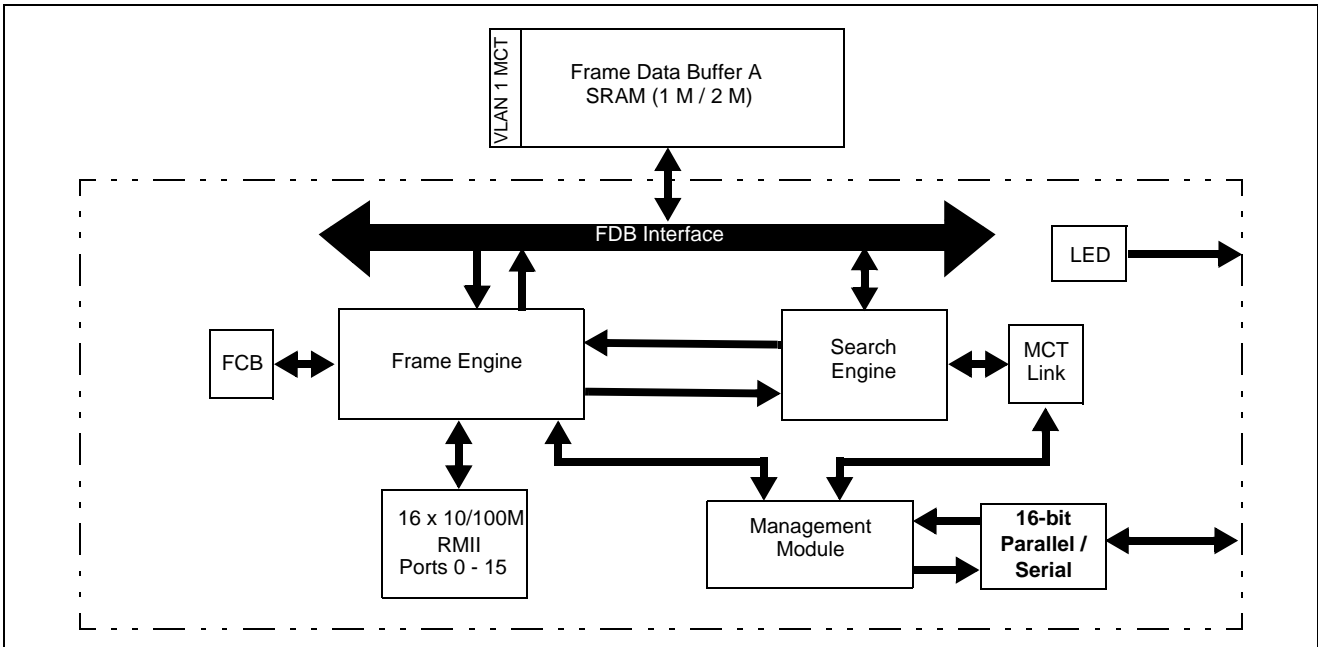
**Ordering Information**

ZL50416/GKC      553 Pin HSBGA  
 ZL50416GKG2    553 Pin HSBGA\*\*

\*\*Pb Free Tin/Silver/Copper

**-40°C to 85°C**

- Packet Filtering and Port Security
  - Static address filtering for source and/or destination MAC
  - Static MAC address not subject to aging
  - Secure mode freezes MAC address learning, each port may independently use this mode
- 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
- QoS Support
  - 4 transmission priorities for Fast Ethernet ports
  - Per-queue weighted random early discard (WRED) with 2 drop precedence levels
  - Scheduling using delay bounded (DB), strict priority (SP), and Weighted Fair Queuing (WFQ) disciplines
  - User controlled WRED thresholds



**Figure 1 - System Block Diagram**

- Buffer management: per-class, shared, and per-port buffer reservations
- Classification based on:
  - Port-based priority: priority in a frame can be overwritten by the priority of port
  - VLAN Priority field in VLAN tagged frame (IEEE 802.1p)
  - DS/TOS field in IP packet
  - UDP/TCP logical ports: 8 hard-wired and 8 programmable ports, including one programmable range
- The drop precedence of the above classifications is programmable
- Supports IEEE 802.3ad link aggregation
  - 2 port trunking groups
  - two groups for 10/100 ports, with up to 4 ports per group
  - Load sharing among trunked ports can be based on:
    - Source and/or destination MAC address
- Port Mirroring
  - supports 2 mirroring ports in managed mode
  - supports a dedicated mirroring port in unmanaged mode
- Built-in MIB statistics counters
- Full Duplex Ethernet IEEE 802.3x Flow Control
- Backpressure flow control for Half Duplex ports
- Full set of LED signals provided by a serial interface
- Recognizes Simple Bandwidth Management (SBM) and Resource Reservation Protocol (RSVP) packets and forwards to CPU
- Built-in reset logic triggered by system malfunction
- Built-in self test (BIST) for internal and external SRAM

## 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 tagged-based Virtual LANs (VLANs). The centralized shared memory architecture permits a very high performance packet forwarding rate at full wire speed. The chip is optimized to provide low-cost, high-performance workgroup switching.

A Frame Buffer Memory domain utilizes 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, or the 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. Two groups are dedicated to 10/100 ports, where 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.

## Changes Summary

The April 2006 issue is the starting point for the change summary section.

Revision Date	Summary of Changes
April 2006	<ul style="list-style-type: none"><li>- Corrected ZL5041x ordering codes (should be /GKC)</li><li>- Added Pb-free order code (ZL50416GKG2)</li><li>- Corrected TSTOUT6 bootstrap description, and clarified only applicable in "managed" mode</li><li>- Corrected ECR1Pn default value (should be 0xC0)</li><li>- Corrected PR100 default value (should be 0x35)</li><li>- Corrected SFCB default value (should be 0x46)</li><li>- Corrected CPU addresses for registers CPUQOSC1,2,3</li></ul>

## Table of Contents

<b>1.0 BGA and Ball Signal Descriptions</b>	<b>12</b>
1.1 BGA Views (Top-View)	12
1.1.1 Encapsulated view in managed mode	12
1.1.2 Encapsulated view in unmanaged mode	13
1.2 Ball – Signal Descriptions	14
1.3 Ball – Signal Name	23
1.4 Signal Mapping and Internal Pull Up/Down Configuration	29
<b>2.0 Block Functionality</b>	<b>31</b>
2.1 Frame Data Buffer (FDB) Interfaces	31
2.2 MAC Modules	31
2.2.1 RMII MAC Module (RMAC)	31
2.2.1.1 GPSI Interface	31
2.2.1.2 SCANLINK and SCANCOL interface	33
2.2.2 CPU MAC Module (CMAC)	33
2.2.3 PHY Addresses	33
2.3 Management Module	33
2.4 Frame Engine	34
2.5 Search Engine	34
2.6 LED Interface	34
2.6.1 Port Status	34
2.6.2 LED Interface Timing Diagram	35
2.7 Internal Memory	35
2.8 Timeout Reset Monitor	35
<b>3.0 System Configuration (Stand-alone and Stacking)</b>	<b>36</b>
3.1 Management and Configuration	36
3.2 Managed Mode	36
3.2.1 Register Configuration, Frame Transmission, and Frame Reception	36
3.2.1.1 Register Configuration	36
3.2.1.2 Rx/Tx of Standard Ethernet Frames	37
3.2.1.3 Control Frames	37
3.3 Unmanaged Mode	38
3.3.1 I2C Interface	38
3.3.1.1 Start Condition	38
3.3.1.2 Address	38
3.3.1.3 Data Direction	38
3.3.1.4 Acknowledgment	39
3.3.1.5 Data	39
3.3.1.6 Stop Condition	39
3.3.2 Synchronous Serial Interface	39
3.3.2.1 Write Command	40
3.3.2.2 Read Command	40
<b>4.0 Data Forwarding Protocol</b>	<b>40</b>
4.1 Unicast Data Frame Forwarding	40
4.2 Multicast Data Frame Forwarding	41
4.3 Frame Forwarding To and From CPU	41
<b>5.0 Memory Interface</b>	<b>42</b>
5.1 Overview	42
5.2 Memory Requirements	42
5.3 Memory Configurations	43
<b>6.0 Search Engine</b>	<b>45</b>
6.1 Search Engine Overview	45

## Table of Contents

6.2 Basic Flow .....	45
6.3 Search, Learning, and Aging .....	46
6.3.1 MAC Search .....	46
6.3.2 Learning .....	46
6.3.3 Aging .....	46
6.4 MAC Address Filtering .....	47
6.5 Port- and Tagged-Based VLAN .....	47
6.5.1 Port-Based VLAN .....	47
6.5.2 Tagged-Based VLAN .....	48
6.6 Quality of Service .....	48
6.6.1 Priority Classification Rule .....	49
<b>7.0 Frame Engine .....</b>	<b>49</b>
7.1 Data Forwarding Summary .....	49
7.2 Frame Engine Details .....	50
7.2.1 FCB Manager .....	50
7.2.2 Rx Interface .....	50
7.2.3 RxDMA .....	50
7.2.4 TxQ Manager .....	50
7.2.5 Port Control .....	50
7.2.6 TxDMA .....	50
<b>8.0 Quality of Service and Flow Control .....</b>	<b>50</b>
8.1 Model .....	50
8.2 Four QoS Configurations .....	52
8.3 Delay Bound .....	52
8.4 Strict Priority and Best Effort .....	52
8.5 Weighted Fair Queuing .....	53
8.6 Rate Control .....	53
8.7 WRED Drop Threshold Management Support .....	54
8.8 Buffer Management .....	54
8.8.1 Dropping When Buffers Are Scarce .....	56
8.9 Flow Control Basics .....	56
8.9.1 Unicast Flow Control .....	56
8.9.2 Multicast Flow Control .....	57
8.10 Mapping to IETF DiffServ Classes .....	57
<b>9.0 Port Trunking .....</b>	<b>58</b>
9.1 Features and Restrictions .....	58
9.2 Unicast Packet Forwarding .....	58
9.3 Multicast Packet Forwarding .....	58
9.4 Unmanaged Trunking .....	59
<b>10.0 Port Mirroring .....</b>	<b>59</b>
10.1 Port Mirroring Features .....	59
10.2 Setting Registers for Port Mirroring .....	59
<b>11.0 Hardware Statistics Counter .....</b>	<b>60</b>
11.1 Hardware Statistics Counters List .....	60
11.2 IEEE 802.3 HUB Management (RFC 1516) .....	61
11.2.1 Event Counters .....	61
11.2.1.1 ReadableTet .....	61
11.2.1.2 ReadableFrame .....	62
11.2.1.3 FCSErrors .....	62
11.2.1.4 AlignmentErrors .....	62
11.2.1.5 FrameTooLongs .....	62

## Table of Contents

11.2.1.6 ShortEvents	62
11.2.1.7 Runts	63
11.2.1.8 Collisions	63
11.2.1.9 LateEvents	63
11.2.1.10 VeryLongEvents	63
11.2.1.11 DataRateMisatches	63
11.2.1.12 AutoPartitions	63
11.2.1.13 TotalErrors	63
11.3 IEEE 802.1 Bridge Management (RFC 1286)	64
11.3.1 Event Counters	64
11.3.1.1 InFrames	64
11.3.1.2 OutFrames	64
11.3.1.3 InDiscards	64
11.3.1.4 DelayExceededDiscards	64
11.3.1.5 MtuExceededDiscards	64
11.4 RMON – Ethernet Statistic Group (RFC 1757)	64
11.4.1 Event Counters	64
11.4.1.1 Drop Events	64
11.4.1.2 Octets	64
11.4.1.3 BroadcastPkts	64
11.4.1.4 MulticastPkts	64
11.4.1.5 CRCAlignErrors	65
11.4.1.6 UndersizePkts	65
11.4.1.7 OversizePkts	65
11.4.1.8 Fragments	65
11.4.1.9 Jabbers	65
11.4.1.10 Collisions	66
11.4.1.11 Packet Count for Different Size Groups	66
11.5 Miscellaneous Counters	66
<b>12.0 Register Definition</b>	<b>67</b>
12.1 Register Description	67
12.2 Directly Accessed Registers (8/16-bit Access Only)	72
12.3 Indirectly Accessed Registers	76
12.3.1 (Group 0 Address) MAC Ports Group	76
12.3.1.1 ECR1Pn: Port n Control Register 1	76
12.3.1.2 ECR2Pn: Port n Control Register 2	76
12.3.2 (Group 1 Address) VLAN Group	78
12.3.2.1 AVTCL – VLAN Type Code Register Low	78
12.3.2.2 AVTCH – VLAN Type Code Register High	78
12.3.2.3 PVMAP00_0 – Port 00 Configuration Register 0	78
12.3.2.4 PVMAP00_1 – Port 00 Configuration Register 1	79
12.3.2.5 PVMAP00_2 – Port 00 Configuration Register 2	79
12.3.2.6 PVMAP00_3 – Port 00 Configuration Register 3	79
12.3.2.7 PVMAPnn_0,1,2,3 – Port nn Configuration Registers	81
12.3.2.8 PVMODE	82
12.3.2.9 PVROUTE0	82
12.3.2.10 PVROUTE1	83
12.3.2.11 PVROUTE2	83
12.3.2.12 PVROUTE3	84
12.3.2.13 PVROUTE4	84
12.3.2.14 PVROUTE5	84
12.3.2.15 PVROUTE6	85

## Table of Contents

12.3.2.16 PVROUTE7	85
12.3.3 (Group 2 Address) Port Trunking Groups	86
12.3.3.1 TRUNK0_L – Trunk group 0 Low (Managed mode only)	86
12.3.3.2 TRUNK0_M – Trunk group 0 Medium (Managed mode only)	86
12.3.3.3 TRUNK0_H – Trunk group 0 High (Managed mode only)	86
12.3.3.4 TRUNK0_MODE – Trunk group 0 mode	86
12.3.3.5 TRUNK0_HASH0 – Trunk group 0 hash result 0 destination port number	87
12.3.3.6 TRUNK0_HASH1 – Trunk group 0 hash result 1 destination port number	87
12.3.3.7 TRUNK0_HASH2 – Trunk group 0 hash result 2 destination port number	87
12.3.3.8 TRUNK0_HASH3 – Trunk group 0 hash result 3 destination port number	87
12.3.3.9 TRUNK1_L – Trunk group 1 Low (Managed mode only)	87
12.3.3.10 TRUNK1_M – Trunk group 1 Medium (Managed mode only)	88
12.3.3.11 TRUNK1_H – Trunk group 1 High (Managed mode only)	88
12.3.3.12 TRUNK1_MODE – Trunk group 1 mode	88
12.3.3.13 TRUNK1_HASH0 – Trunk group 1 hash result 0 destination port number	88
12.3.3.14 TRUNK1_HASH1 – Trunk group 1 hash result 1 destination port number	88
12.3.3.15 TRUNK1_HASH2 – Trunk group 1 hash result 2 destination port number	88
12.3.3.16 TRUNK1_HASH3 – Trunk group 1 hash result 3 destination port number	89
12.3.3.17 TRUNK2_MODE – Trunk group 2 mode	89
12.3.3.18 TRUNK2_HASH0 – Trunk group 2 hash result 0 destination port number	89
12.3.3.19 TRUNK2_HASH1 – Trunk group 2 hash result 1 destination port number	89
12.3.3.20 MULTICAST_HASHn_0 – Multicast hash result 0~3 mask byte 0	90
12.3.3.21 MULTICAST_HASHn_1 – Multicast hash result 0~3 mask byte 1	90
12.3.3.22 MULTICAST_HASHn_2 – Multicast hash result 0~3 mask byte 2	90
12.3.3.23 MULTICAST_HASHn_3 – Multicast hash result 0~3 mask byte 3	90
12.3.4 (Group 3 Address) CPU Port Configuration Group	90
12.3.4.1 MAC0 – CPU Mac address byte 0	90
12.3.4.2 MAC1 – CPU Mac address byte 1	90
12.3.4.3 MAC2 – CPU Mac address byte 2	90
12.3.4.4 MAC3 – CPU Mac address byte 3	91
12.3.4.5 MAC4 – CPU Mac address byte 4	91
12.3.4.6 MAC5 – CPU Mac address byte 5	91
12.3.4.7 INT_MASK0 – Interrupt Mask 0	91
12.3.4.8 INTP_MASK0 – Interrupt Mask for MAC Port 0,1	92
12.3.4.9 INTP_MASKn – Interrupt Mask for MAC Ports	92
12.3.4.10 RQS – Receive Queue Select	93
12.3.4.11 RQSS – Receive Queue Status	93
12.3.4.12 TX_AGE – Tx Queue Aging timer	94
12.3.5 (Group 4 Address) Search Engine Group	94
12.3.5.1 AGETIME_LOW – MAC address aging time Low	94
12.3.5.2 AGETIME_HIGH – MAC address aging time High	94
12.3.5.3 V_AGETIME – VLAN to Port aging time	94
12.3.5.4 SE_OPMODE – Search Engine Operation Mode	95
12.3.5.5 SCAN – SCAN Control Register (default 00)	96
12.3.6 (Group 5 Address) Buffer Control/QOS Group	96
12.3.6.1 FCBAT – FCB Aging Timer	96
12.3.6.2 QOSC – QOS Control	96
12.3.6.3 FCR – Flooding Control Register	97
12.3.6.4 AVPML – VLAN Tag Priority Map	98
12.3.6.5 AVPMM – VLAN Priority Map	98
12.3.6.6 AVPMH – VLAN Priority Map	98
12.3.6.7 TOSPML – TOS Priority Map	99

## Table of Contents

12.3.6.8 TOSPM – TOS Priority Map	99
12.3.6.9 TOSPMH – TOS Priority Map	99
12.3.6.10 AVDM – VLAN Discard Map	99
12.3.6.11 TOSDML – TOS Discard Map	100
12.3.6.12 BMRC - Broadcast/Multicast Rate Control	100
12.3.6.13 UCC – Unicast Congestion Control	101
12.3.6.14 MCC – Multicast Congestion Control	101
12.3.6.15 PR100 – Port Reservation for 10/100 ports	101
12.3.6.16 SFCB – Share FCB Size	102
12.3.6.17 C2RS – Class 2 Reserve Size	102
12.3.6.18 C3RS – Class 3 Reserve Size	102
12.3.6.19 C4RS – Class 4 Reserve Size	103
12.3.6.20 C5RS – Class 5 Reserve Size	103
12.3.6.21 C6RS – Class 6 Reserve Size	103
12.3.6.22 C7RS – Class 7 Reserve Size	103
12.3.6.23 QOSC00–02 - Classes Byte Limit Set 0	103
12.3.6.24 QOSC03–05 - Classes Byte Limit Set 1	104
12.3.6.25 QOSC06–08 - Classes Byte Limit Set 2	104
12.3.6.26 QOSC09–11 - Classes Byte Limit Set 3	104
12.3.6.27 QOSC24–27 - Classes WFQ Credit Set 0	105
12.3.6.28 QOSC28–31 - Classes WFQ Credit Set 1	105
12.3.6.29 QOSC32–35 - Classes WFQ Credit Set 2	105
12.3.6.30 QOSC36–39 - Classes WFQ Credit Set 3	106
12.3.6.31 RDRC0 – WRED Rate Control 0	106
12.3.6.32 RDRC1 – WRED Rate Control 1	106
12.3.6.33 USER_PORT0~7)_L/H – User Define Logical Port (0~7)	107
12.3.6.34 USER_PORT_[1:0]_PRIORITY - User Define Logic Port 1 and 0 Priority	108
12.3.6.35 USER_PORT_[3:2]_PRIORITY - User Define Logic Port 3 and 2 Priority	108
12.3.6.36 USER_PORT_[5:4]_PRIORITY - User Define Logic Port 5 and 4 Priority	108
12.3.6.37 USER_PORT_[7:6]_PRIORITY - User Define Logic Port 7 and 6 Priority	108
12.3.6.38 USER_PORT_ENABLE[7:0] – User Define Logic 7 to 0 Port Enables	108
12.3.6.39 WELL_KNOWN_PORT[1:0]_PRIORITY- Well Known Logic Port 1 and 0 Priority	109
12.3.6.40 WELL_KNOWN_PORT[3:2]_PRIORITY- Well Known Logic Port 3 and 2 Priority	109
12.3.6.41 WELL_KNOWN_PORT [5:4]_PRIORITY- Well Known Logic Port 5 and 4 Priority	109
12.3.6.42 WELL_KNOWN_PORT [7:6]_PRIORITY- Well Known Logic Port 7 and 6 Priority	109
12.3.6.43 WELL_KNOWN_PORT_ENABLE [7:0] – Well Known Logic 7 to 0 Port Enables	110
12.3.6.44 RLOWL – User Define Range Low Bit 7:0	110
12.3.6.45 RLOWH – User Define Range Low Bit 15:8	110
12.3.6.46 RHIGHL – User Define Range High Bit 7:0	110
12.3.6.47 RHIGHH – User Define Range High Bit 15:8	110
12.3.6.48 RPRIORITY – User Define Range Priority	110
12.3.6.49 CPUQOSC1,2,3	111
12.3.7 (Group 6 Address) MISC Group	111
12.3.7.1 MII_OP0 – MII Register Option 0	111
12.3.7.2 MII_OP1 – MII Register Option 1	111
12.3.7.3 FEN – Feature Register	112
12.3.7.4 MIIC0 – MII Command Register 0	113
12.3.7.5 MIIC1 – MII Command Register 1	113
12.3.7.6 MIIC2 – MII Command Register 2	113
12.3.7.7 MIIC3 – MII Command Register 3	114
12.3.7.8 MIID0 – MII Data Register 0	114
12.3.7.9 MIID1 – MII Data Register 1	114

## Table of Contents

12.3.7.10 LED Mode – LED Control . . . . .	114
12.3.7.11 DEVICE Mode . . . . .	115
12.3.7.12 CHECKSUM - EEPROM Checksum . . . . .	115
12.3.8 (Group 7 Address) Port Mirroring Group . . . . .	115
12.3.8.1 MIRROR1_SRC - Port Mirror source port . . . . .	115
12.3.8.2 MIRROR1_DEST – Port Mirror destination . . . . .	116
12.3.8.3 MIRROR2_SRC – Port Mirror source port . . . . .	116
12.3.8.4 MIRROR2_DEST – Port Mirror destination . . . . .	116
12.3.9 (Group F Address) CPU Access Group . . . . .	117
12.3.9.1 GCR-Global Control Register . . . . .	117
12.3.9.2 DCR - Device Status and Signature Register . . . . .	118
12.3.9.3 DCR1 - Chip Status . . . . .	118
12.3.9.4 DPST – Device Port Status Register . . . . .	119
12.3.9.5 DTST – Data read back register . . . . .	120
12.3.9.6 DA – Dead or Alive Register . . . . .	120
12.4 Characteristics and Timing . . . . .	121
12.4.1 Absolute Maximum Ratings . . . . .	121
12.4.2 DC Electrical Characteristics . . . . .	121
12.4.3 Recommended Operating Conditions . . . . .	122
12.5 AC Characteristics and Timing . . . . .	123
12.5.1 Typical Reset & Bootstrap Timing Diagram . . . . .	123
12.5.2 Typical CPU Timing Diagram for a CPU Write Cycle . . . . .	124
12.5.3 Typical CPU Timing Diagram for a CPU Read Cycle . . . . .	125
12.5.4 Local Frame Buffer SDRAM Memory Interface . . . . .	126
12.5.4.1 Local SDRAM Memory Interface A . . . . .	126
12.5.5 Reduced Media Independent Interface . . . . .	128
12.5.6 LED Interface . . . . .	129
12.5.7 SCANLINK, SCANCOL Interface . . . . .	130
12.6 MDIO Interface . . . . .	131
12.6.1 I2C Interface . . . . .	132
12.6.2 Synchronous Serial Interface . . . . .	133

## List of Figures

Figure 1 - System Block Diagram . . . . .	1
Figure 2 - GPSI (7WS) Mode Connection Diagram . . . . .	32
Figure 3 - SCANLINK and SCANCOL Status Diagram . . . . .	33
Figure 4 - Timing Diagram of LED Interface . . . . .	35
Figure 5 - Overview of the CPU Interface . . . . .	36
Figure 6 - Data Transfer Format for I2C Interface . . . . .	38
Figure 7 - SRAM Interface Block Diagram (DMAs for 10/100 Ports Only) . . . . .	42
Figure 8 - Memory Configuration . . . . .	43
Figure 9 - Memory Configuration For 1 M/bank, 1 Layer . . . . .	44
Figure 10 - Memory Configuration For 2 M/bank, 2 Layers . . . . .	44
Figure 11 - Memory Configuration For 2 M/bank, 1 Layer . . . . .	45
Figure 12 - Priority Classification Rule . . . . .	49
Figure 13 - Buffer Partition Scheme Used to Implement Buffer Management . . . . .	55
Figure 14 - Typical Reset & Bootstrap Timing Diagram . . . . .	123
Figure 15 - Typical CPU Timing Diagram for a CPU Write Cycle . . . . .	124
Figure 16 - Typical CPU Timing Diagram for a CPU Read Cycle . . . . .	125
Figure 17 - Local Memory Interface – Input Setup and Hold Timing . . . . .	126
Figure 18 - Local Memory Interface - Output Valid Delay Timing . . . . .	126
Figure 19 - AC Characteristics – Reduced Media Independent Interface . . . . .	128
Figure 20 - AC Characteristics – Reduced Media Independent Interface . . . . .	128
Figure 21 - AC Characteristics – LED Interface . . . . .	129
Figure 22 - SCANLINK, SCANCOL Output Delay Timing . . . . .	130
Figure 23 - SCANLINK, SCANCOL Setup Timing . . . . .	130
Figure 24 - MDIO Input Setup and Hold Timing . . . . .	131
Figure 25 - MDIO Output Delay Timing . . . . .	131
Figure 26 - I2C Input Setup Timing . . . . .	132
Figure 27 - I2C Output Delay Timing . . . . .	132
Figure 28 - Serial Interface Setup Timing . . . . .	133
Figure 29 - Serial Interface Output Delay Timing . . . . .	133

## 1.0 BGA and Ball Signal Descriptions

### 1.1 BGA Views (Top-View)

#### 1.1.1 Encapsulated view in managed 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_D 4	LA_D 7	LA_D 10	LA_D 13	LA_D 14	LA_D 15	LA_A 4	LA_O E0#	LA_A 8	LA_A 13	LA_A 16	LA_A 19	LA_D 33	LA_D 36	LA_D 39	LA_D 42	LA_D 45	P_DA TA13	P_DA TA10	P_DA TA7	P_DA TA4	P_DA TA1	P_A1	P_A0	P_WE #	TSTO UT7	A				
B		LA_D 1	LA_D 3	LA_D 6	LA_D 9	LA_D 12	LA_D 14	LA_D 15	LA_A DSC#	LA_O E1#	LA_A 7	LA_A 12	LA_A 15	LA_A 18	LA_D 32	LA_A 35	LA_D 38	LA_A 41	LA_D 44	P_DA TA14	P_DA TA11	LA_D 62	P_DA TA5	P_DA TA2	P_DA TA6	P_INT #	P_RD #	TSTO UT8	TSTO UT3	B			
C	LA_C LK 0	LA_D 2	LA_D 5	LA_D 8	LA_D 11	LA_D 13	LA_D 14	LA_A 3	LA_O E#	LA_W E#	T_MO DE1	LA_A 11	LA_A 14	LA_A 17	LA_A 20	LA_D 34	LA_D 37	LA_D 40	LA_D 43	P_DA TA15	P_DA TA12	P_DA TA9	P_A2	P_DA TA3	P_DA TA0	P_CS #	TSTO UT11	TSTO UT9	TSTO UT4	TSTO UT0	C		
D	VSSA	LA_D 17	LA_D 19	LA_D 21	LA_D 23	LA_D 25	LA_D 27	LA_D 29	LA_D 31	LA_A 6	LA_A 10	LA_A 10	LA_W E0#	LD_D 49	LA_D 51	LA_D 53	LA_D 55	LA_D 57	LA_D 59	LA_D 61	LA_D 63	LA_D 47	SCAN COL	SCAN CLK	TSTO UT14	TSTO UT13	TSTO UT12	TSTO UT10	TSTO UT5	TSTO UT1	D		
E	SCLK	LA_D 16	LA_D 18	LA_D 20	LA_D 22	LA_D 24	LA_D 26	LA_D 28	LA_D 30	LA_A 5	LA_A 9	LA_A 9	LA_W E1#	LA_D 48	LA_D 50	LA_D 52	LA_D 54	LA_D 56	LA_D 58	LA_D 60	P_DA TA8	LA_D 46	NC	SCAN LNK	TSTO UT15	RSVD	RSVD	SCAN MD	TSTO UT6	TSTO UT2	E		
F	VDDA	RESI N#	SCAN EN	RSVD	RSVD									VCC	VCC	VCC	VCC	VCC									RSVD	RSVD	RSVD	RSVD	RSVD	F	
G	RSVD	RESO UT#	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	G	
H	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	H	
J	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	J	
K	RSVD	RSVD	RSVD	RSVD	RSVD								VDD	VDD													RSVD	RSVD	RSVD	RSVD	RSVD	K	
L	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	L	
M	RSVD	RSVD	RSVD	RSVD	RSVD						VDD																RSVD	RSVD	RSVD	RSVD	RSVD	M	
N	RSVD	RSVD	RSVD	RSVD	RSVD	VCC					VDD																VCC	RSVD	RSVD	NC	NC	RSVD	N
P	RSVD	RSVD	RSVD	RSVD	RSVD	VCC																					VCC	RSVD	RSVD	NC	MDIO	RSVD	P
R	RSVD	RSVD	RSVD	RSVD	RSVD	VCC																					VCC	RSVD	RSVD	NC	MDC	M_CL K	R
T	RSVD	RSVD	RSVD	RSVD	RSVD	VCC																					VCC	RSVD	RSVD	RSVD	RSVD	RSVD	T
U	RSVD	RSVD	T_MO DE0	RSVD	RSVD	VCC					VDD																VCC	RSVD	RSVD	RSVD	RSVD	RSVD	U
V	RSVD	RSVD	RSVD	RSVD	RSVD						VDD																	RSVD	RSVD	RSVD	RSVD	RSVD	V
W	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	W	
Y	RSVD	RSVD	RSVD	RSVD	RSVD								VDD	VDD													RSVD	RSVD	RSVD	RSVD	RSVD	Y	
AA	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	AA	
AB	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	AB	
AC	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	AC	
AD	RSVD	RSVD	RSVD	RSVD	RSVD																						RSVD	RSVD	RSVD	RSVD	RSVD	AD	
AE	M0_T XEN	M0_T XD0	M0_T XD1	M3_T XD1	M3_T XEN	M3_R XD0	M5_T XD1	M5_T XEN	M5_R XD0	M8_T XD1	M8_T XEN	M8_R XD0	M10_T XD1	M10_T XEN	M10_R XD0	M13_T XD1	RSVD	M15_T XD1	RSVD	M15_T XEN	M15_R XD0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	NC	AE	
AF	M0_R XD1	M0_R XD0	M0_C RS	M3_T XD0	M3_C RS	M3_R XD1	M5_T XD0	M5_C RS	M5_R XD1	M8_T XD0	M8_C RS	M8_R XD1	M10_T XD0	M10_C RS	M10_R XD1	M13_T XD0	M13_C RS	M13_R XD1	M14_C RS	RSVD	M15_R XD1	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AF		
AG	M1_T XEN	M1_T XD0	M1_T XD1	M2_T XD1	M2_C RS	M4_T XD1	M4_C RS	M6_T XD1	M6_C RS	M7_T XD1	M7_C RS	M9_T XD1	M9_C RS	M11_T XD1	M11_C RS	M12_T XD1	M12_C RS	M14_T XD1	M15_T XD0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AG		
AH		M1_R XD0	M1_C RS	M2_T XD0	M2_R XD0	M4_T XD0	M4_R XD0	M6_T XD0	M7_R XD0	M7_T XD0	M7_C RS	M9_T XD0	M9_R XD0	M11_T XD0	M11_R XD0	M12_T XD0	M12_C RS	M14_T XD0	M14_R XD0	M13_R XD0	M15_C RS	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AH		
AJ			M1_R XD1	M2_T XEN	M2_R XD1	M4_T XEN	M4_R XD1	M6_T XEN	M7_R XD1	M7_T XEN	M7_C RS	M9_T XEN	M9_R XD1	M11_T XEN	M11_R XD1	M12_T XEN	M12_C RS	M14_T XEN	M14_R XD1	RSVD	M13_T XEN	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AJ		
	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				

1.1.2 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_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	MIRROR4	MIRROR1	SCL	SDA	STROBE	TSTO UT7		A				
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_A35	LA_D38	LA_A41	LA_D44	OE_CLK1	LA_CLK1	LA_D62	MIRROR5	MIRROR2	RSVD	RSVD	D0	TSTO UT8	TSTO UT3	B				
C	LA_CLK	LA_D0	LA_D2	LA_D5	LA_D8	LA_D11	LA_A3	LA_OE#	LA_WE#	T_MO DE1	LA_A11	LA_A14	LA_A17	LA_A20	LA_D34	LA_D37	LA_D40	LA_D43	OE_CLK2	LA_CLK2	P_D	TRUNK0	MIRROR3	MIRROR0	AUTO FD	TSTO UT11	TSTO UT9	TSTO UT4	TSTO UT0	C		
D	VSSA	LA_D17	LA_D19	LA_D21	LA_D23	LA_D25	LA_D27	LA_D29	LA_D31	LA_A6	LA_A10	LA_WE0#	LD_D49	LA_D51	LA_D53	LA_D55	LA_D57	LA_D59	LA_D61	LA_D63	LA_D47	SCAN COL	SCAN CLK	TSTO UT14	TSTO UT13	TSTO UT12	TSTO UT10	TSTO UT5	TSTO UT1	D		
E	SCLK	LA_D16	LA_D18	LA_D20	LA_D22	LA_D24	LA_D26	LA_D28	LA_D30	LA_A5	LA_A9	LA_WE1#	LA_D48	LA_D50	LA_D52	LA_D54	LA_D56	LA_D58	LA_D60	RSVD	LA_D46	NC	SCAN LNK	TSTO UT15	RSVD	RSVD	SCAN MD	TSTO UT6	TSTO UT2	E		
F	VDDA	RESI N#	SCAN EN	RSVD	RSVD								VCC	VCC	VCC	VCC	VCC								RSVD	RSVD	RSVD	RSVD	RSVD	F		
G	RSVD	RESO UT#	RSVD	RSVD	RSVD																				RSVD	RSVD	RSVD	RSVD	RSVD	G		
H	RSVD	RSVD	RSVD	RSVD	RSVD																				RSVD	RSVD	RSVD	RSVD	RSVD	H		
J	RSVD	RSVD	RSVD	RSVD	RSVD																				RSVD	RSVD	RSVD	RSVD	RSVD	J		
K	RSVD	RSVD	RSVD	RSVD	RSVD						VDD	VDD													RSVD	RSVD	RSVD	RSVD	RSVD	K		
L	RSVD	RSVD	RSVD	RSVD	RSVD																				RSVD	RSVD	RSVD	RSVD	RSVD	L		
M	RSVD	RSVD	RSVD	RSVD	RSVD					VDD		VSS	VSS	VSS	VSS	VSS	VSS	VSS						VDD		RSVD	RSVD	RSVD	RSVD	RSVD	M	
N	RSVD	RSVD	RSVD	RSVD	RSVD	VCC				VDD		VSS	VSS	VSS	VSS	VSS	VSS	VSS						VDD	VCC	RSVD	RSVD	NC	NC	RSVD	N	
P	RSVD	RSVD	RSVD	RSVD	RSVD	VCC						VSS	VSS	VSS	VSS	VSS	VSS	VSS							VCC	RSVD	RSVD	NC	MDIO	RSVD	P	
R	RSVD	RSVD	RSVD	RSVD	RSVD	VCC						VSS	VSS	VSS	VSS	VSS	VSS	VSS							VCC	RSVD	RSVD	NC	MDC	M_CLK	R	
T	RSVD	RSVD	RSVD	RSVD	RSVD	VCC						VSS	VSS	VSS	VSS	VSS	VSS	VSS							VCC	RSVD	RSVD	RSVD	RSVD	RSVD	T	
U	RSVD	RSVD	T_MO DE0	RSVD	RSVD	VCC				VDD		VSS	VSS	VSS	VSS	VSS	VSS	VSS						VDD	VCC	RSVD	RSVD	RSVD	RSVD	RSVD	U	
V	RSVD	RSVD	RSVD	RSVD	RSVD					VDD		VSS	VSS	VSS	VSS	VSS	VSS	VSS							VDD	RSVD	RSVD	RSVD	RSVD	RSVD	V	
W	RSVD	RSVD	RSVD	RSVD	RSVD																					RSVD	RSVD	RSVD	RSVD	RSVD	W	
Y	RSVD	RSVD	RSVD	RSVD	RSVD						VDD	VDD														RSVD	RSVD	RSVD	RSVD	RSVD	Y	
AA	RSVD	RSVD	RSVD	RSVD	RSVD																					RSVD	RSVD	RSVD	RSVD	RSVD	AA	
AB	RSVD	RSVD	RSVD	RSVD	RSVD																					RSVD	RSVD	RSVD	RSVD	RSVD	AB	
AC	RSVD	RSVD	RSVD	RSVD	RSVD																					RSVD	RSVD	RSVD	RSVD	RSVD	AC	
AD	RSVD	RSVD	RSVD	RSVD	RSVD							VCC	VCC	VCC	VCC	VCC										RSVD	RSVD	RSVD	RSVD	RSVD	AD	
AE	M0_T XEN	M0_T XD0	M0_T XD1	M3_T XD1	M3_T XEN	M3_R XD0	M5_T XD1	M5_T XEN	M5_R XD0	M8_T XD1	M8_T XEN	M8_R XD0	M10_T XD1	M10_T XEN	M10_R XD0	M13_T XD1	RSVD	M15_T XD1	RSVD	M15_T XEN	M15_R XD0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	NC	AE
AF	M0_R XD1	M0_R XD0	M0_C RS	M3_T XD0	M3_C RS	M3_R XD1	M5_T XD0	M5_C RS	M5_R XD1	M8_T XD0	M8_C RS	M8_R XD1	M10_T XD0	M10_C RS	M10_R XD1	M13_T XD0	M13_C RS	M13_R XD1	M14_C RS	RSVD	M15_R XD1	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AF	
AG	M1_T XEN	M1_T XD0	M1_T XD1	M2_T XD1	M2_C RS	M4_T XD1	M4_C RS	M6_T XD1	M6_C RS	M7_T XD1	M7_C RS	M9_T XD1	M9_C RS	M11_T XD1	M11_C RS	M12_T XD1	M12_C RS	M14_T XD1	M15_T XD0	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AG	
AH	M1_R XD0	M1_C RS	M2_T XD0	M2_R XD0	M4_T XD0	M4_R XD0	M6_T XD0	M7_R XD0	M7_T XD0	M7_R XD0	M9_T XD0	M9_R XD0	M11_T XD0	M11_R XD0	M12_T XD0	M12_R XD0	M14_T XD0	M14_R XD0	M13_R XD0	M15_C RS	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AH	
AJ		M1_R XD1	M2_T XEN	M2_R XD1	M4_T XEN	M4_R XD1	M6_T XEN	M7_R XD1	M7_T XEN	M7_R XD1	M9_T XEN	M9_R XD1	M11_T XEN	M11_R XD1	M12_T XEN	M12_R XD1	M14_T XEN	M14_R XD1	RSVD	M13_T XEN	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	AJ	
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				

## 1.2 Ball – Signal Descriptions

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

Notes:

# = Active low signal

Weak internal pull-up/down resistors are nominal 100k ohm

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

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 weak internal pull-up (except P_DATA[7:6] with weak internal 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>CPU BUS Interface in Unmanaged Mode - Use I2C and Serial control interface to configure the system</b>			
A24	SCL	Output	I2C Data Clock
A25	SDA	I/O-TS with weak internal pull-up	I2C Data I/O
A26	STROBE	Input with weak internal pull-up	Serial Strobe Pin
B26	DATAIN (D0)	Input with weak internal pull-up	Serial Data Input (D0)
C25	DATAOUT (AUTOFD)	Output with weak internal pull-up	Serial Data Output (AutoFD)

Ball No(s)	Symbol	I/O	Description
<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 weak internal 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 weak internal pull-up	Frame Bank A Address Status Control
C1	LA_CLK	Output	Frame Bank A Clock Input
C9	LA_WE#	Output with weak internal pull-up	Frame Bank A Write Chip Select for one layer SRAM configuration
D12	LA_WE0#	Output with weak internal pull-up	Frame Bank A Write Chip Select for lower layer of two layers SRAM configuration
E12	LA_WE1#	Output with weak internal pull-up	Frame Bank A Write Chip Select for upper layer of two layers SRAM configuration
C8	LA_OE#	Output with weak internal pull-up	Frame Bank A Read Chip Select for one bank SRAM configuration
A9	LA_OE0#	Output with weak internal pull-up	Frame Bank A Read Chip Select for lower layer of two layers SRAM configuration
B9	LA_OE1#	Output with weak internal 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 weak internal pull-up	MII Management Data I/O – (Common for all MII Ports –[15:0])

Ball No(s)	Symbol	I/O	Description
R29	M_CLK	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	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	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_DV	Input with weak internal pull-down	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, slew with weak internal pull-up	Ports [15:0] – Transmit Enable  Bootstrap 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/ TSTOUT0	I/O-TS with weak internal pull-up	LED Serial Interface Output Clock
D29	LED_SYN/ TSTOUT1	I/O-TS with weak internal pull-up	LED Output Data Stream Envelope
E29	LED_BIT/ TSTOUT2	I/O-TS with weak internal pull-up	LED Serial Data Output Stream
B27, A27, E28, D28, C28, B28	TSTOUT[8:3]	I/O- TS with pull up	Reserved
C27	INIT_DONE/ TSTOUT9	I/O-TS with weak internal pull-up	System start operation
D27	INIT_START/ TSTOUT10	I/O-TS with weak internal pull-up	Start initialization

Ball No(s)	Symbol	I/O	Description
C26	CHECKSUM_OK/ TSTOUT11	I/O-TS with weak internal pull-up	EEPROM read OK
D26	FCB_ERR/ TSTOUT12	I/O-TS with weak internal pull-up	FCB memory self test fail
D25	MCT_ERR/ TSTOUT13	I/O-TS with weak internal pull-up	MCT memory self test fail
D24	BIST_IN_PRC/ TSTOUT14	I/O-TS with weak internal pull-up	Processing memory self test
E24	BIST_DONE/ TSTOUT15	I/O-TS with weak internal pull-up	Memory self test done
<b>Test Facility</b>			
U3, C10	T_MODE0, T_MODE1	I/O-TS  Must be externally pulled-up	Test Pins. Manufacturing test option. 00 – Test mode – Set test mode upon reset, and provides NANDTree test output during test mode 01 - Reserved - Do not use 10 - Reserved - Do not use 11 – Normal mode  Use external pull-ups for normal mode
F3	SCAN_EN	Input with weak internal pull-down	Scan Enable. Manufacturing test option.  Should not be connected for proper operation.
E27	SCANMODE	Input with weak internal pull-down	Scan Mode Enable. Manufacturing test option. 1 – Enable Test mode 0 - Normal mode (open)  Should not be connected for proper operation.
<b>System Clock, Power, and Ground Pins</b>			
E1	SCLK	Input	System Clock at 100 MHz
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

Ball No(s)	Symbol	I/O	Description
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	VDDA	Analog Power	Analog +2.5 Volt DC Supply
D1	VSSA	Analog Ground	Analog Ground
<b>MISC</b>			
D22	SCANCOL	I/O	Scans the Collision signal of Home PHY
D23	SCANCLK	Output	Clock for scanning Home PHY collision and link
E23	SCANLINK	I/O	Link up signal from Home PHY
F2	RESIN#	Input	Reset Input
G2	RESETOUT#	Output	Reset PHY
E22, N27, N28, P27, R27, AE29	NC	NC	No Internal Connect

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, U27, V29, V28, V27, W29, W28, G26, G25, H26, H25, J26, J25, G27, H29, H28, H27, J29, J28, 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	RSVD	N/A	Reserved. Leave unconnected.
<b>Bootstrap Pins (1= pull-up 0= pull-down) (Default = 1 due to weak internal pull-ups)</b>			
<b>After reset TSTOUT0 to TSTOU15 are used by the LED interface.</b>			
C29	TSTOUT0	Input (Reset Only) with weak internal pull-up	Reserved
D29	TSTOUT1	Input (Reset Only) with weak internal pull-up	RMII MAC Power Saving Enable 1 – power saving 0 – No power saving

Ball No(s)	Symbol	I/O	Description
E29	TSTOUT2	Input (Reset Only) with weak internal pull-up  Must be externally pulled-down	Manufacturing Option. Must be '0'.
B28	TSTOUT3	Input (Reset Only) with weak internal pull-up	Reserved
C28	TSTOUT4	Input (Reset Only) with weak internal pull-up	Reserved
D28	TSTOUT5	Input (Reset Only) with weak internal pull-up	Scan Speed: ¼ SCLK or SCLK 1 - SCLK 0 – ¼ SCLK (HPNA)
E28	TSTOUT6	Input (Reset Only) with weak internal pull-up	CPU Port Mode 1 - 16 bit Bus Mode 0 - 8 bit Bus Mode  Only applicable in managed mode (TSTOUT14='0').
A27	TSTOUT7	Input (Reset Only) with weak internal pull-up	Memory Size 1 - 128 K x 32 or 128 K x 64 (1 M total) 0 - -256 K x 32 or 256 K x 64 (2 M total)
B27	TSTOUT8	Input (Reset Only) with weak internal pull-up	EEPROM Installed 1 – EEPROM not installed 0 – EEPROM installed  Only applicable in unmanaged mode.
C27	TSTOUT9	Input (Reset Only) with weak internal pull-up	MCT Aging 1 – MCT aging enable 0 – MCT aging disable
D27	TSTOUT10	Input (Reset Only) with weak internal pull-up  Must be externally pulled-down	Manufacturing Option. Must be '0'.
C26	TSTOUT11	Input (Reset Only) with weak internal pull-up	Timeout Reset 1 – Time out reset enable 0 – Time out reset disable  If enabled, issue reset if any state machine did not go back to idle for 5sec.

Ball No(s)	Symbol	I/O	Description
D26	TSTOUT12	Input (Reset Only) with weak internal pull-up	Manufacturing Option. Must be '1'.
D25	TSTOUT13	Input (Reset Only) with weak internal pull-up	FDB RAM depth (1 or 2 layers) 1 – 1 layer 0 – 2 layer
D24	TSTOUT14	Input (Reset Only) with weak internal pull-up	CPU installed 1 – CPU not installed 0 – CPU installed
E24	TSTOUT15	Input (Reset Only) with weak internal pull-up	SRAM Test Mode 1 – Normal operation 0 – Enable test mode
AD29, AG28, AJ26, AE26, AJ24, AE23, AJ22, AJ20, AE20, AJ18, AJ21, AJ16, AJ14, AE14, AJ12, AE11, AJ10, AJ8, AE8, AJ6, AE5, AJ4, AG1, AE1	M[15:0]_TXEN	Input (Reset Only) with weak internal pull-up	1 – RMII 0 – GPSI
C21	P_D[9] (P_D)	Input (Reset Only) with weak internal pull-up  Must be externally pulled-down	Manufacturing Option. Must be '0'.
C19, B19, A19	P_D[15:13] (OE_CLK[2:0])	Input (Reset Only) with weak internal pull-up  Recommend 001 with external pull-downs on P_D[15:14] (OE_CLK[2:1]).	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] (L_CLK[2:0])	Input (Reset Only) with weak internal pull-up  Recommend 011 with external pull-down on P_D[12] (L_CLK[2]).	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] (OE_CLK[2:0]).  Suggested value is 011.
B22, A22, C23, B23, A23, C24	P_D[5:0] (MIRROR[5:0])	Input (Reset Only) with weak internal pull-up	Dedicated Port Mirror Mode.  The first 5 bits ([4:0]) select the port to be mirrored. The last bit ([5]) selects either ingress or egress data.

---

<b>Ball No(s)</b>	<b>Symbol</b>	<b>I/O</b>	<b>Description</b>
C22	P_A[0] (TRUNK0)	Input (Reset Only) with weak internal pull-down	Trunk Group 0 Enable 0 – Disable 1 – Enable
A21	P_D[7] (TRUNK1)	Input (Reset Only) with weak internal pull-down	Trunk Group 1 Enable 0 – Disable 1 – Enable

### 1.3 Ball – Signal Name

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	RSVD
E19	LA_D[60]	E2	LA_D[16]	F5	RSVD
D18	LA_D[59]	A7	LA_D[15]	G4	RSVD
E18	LA_D[58]	B7	LA_D[14]	G5	RSVD
D17	LA_D[57]	A6	LA_D[13]	H4	RSVD
E17	LA_D[56]	B6	LA_D[12]	H5	RSVD
D16	LA_D[55]	C6	LA_D[11]	J4	RSVD
E16	LA_D[54]	A5	LA_D[10]	J5	RSVD
D15	LA_D[53]	B5	LA_D[9]	K4	RSVD
E15	LA_D[52]	C5	LA_D[8]	K5	RSVD
D14	LA_D[51]	A4	LA_D[7]	L4	RSVD
E14	LA_D[50]	B4	LA_D[6]	L5	RSVD
D13	LA_D[49]	C4	LA_D[5]	M4	RSVD
E13	LA_D[48]	A3	LA_D[4]	M5	RSVD
D21	LA_D[47]	B3	LA_D[3]	N4	RSVD
E21	LA_D[46]	C3	LA_D[2]	N5	RSVD
A18	LA_D[45]	B2	LA_D[1]	G3	RSVD
B18	LA_D[44]	C2	LA_D[0]	H1	RSVD
C18	LA_D[43]	C14	LA_A[20]	H2	RSVD
A17	LA_D[42]	A13	LA_A[19]	H3	RSVD
B17	LA_D[41]	B13	LA_A[18]	J1	RSVD
C17	LA_D[40]	C13	LA_A[17]	J2	RSVD
A16	LA_D[39]	A12	LA_A[16]	J3	RSVD
B16	LA_D[38]	B12	LA_A[15]	K1	RSVD
C16	LA_D[37]	C12	LA_A[14]	K2	RSVD
A15	LA_D[36]	A11	LA_A[13]	K3	RSVD
B15	LA_D[35]	B11	LA_A[12]	L1	RSVD
C15	LA_D[34]	C11	LA_A[11]	L2	RSVD
A14	LA_D[33]	D11	LA_A[10]	L3	RSVD
B14	LA_D[32]	E11	LA_A[9]	M1	RSVD

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
D9	LA_D[31]	A10	LA_A[8]	M2	RSVD
E9	LA_D[30]	B10	LA_A[7]	M3	RSVD
D8	LA_D[29]	D10	LA_A[6]	U4	RSVD
E8	LA_D[28]	E10	LA_A[5]	U5	RSVD
D7	LA_D[27]	A8	LA_A[4]	V4	RSVD
E7	LA_D[26]	C7	LA_A[3]	V5	RSVD
D6	LA_D[25]	B8	LA_DSC#	W4	RSVD
E6	LA_D[24]	C1	LA_CLK	W5	RSVD
D5	LA_D[23]	C9	LA_WE#	Y4	RSVD
E5	LA_D[22]	D12	LA_WE0#	Y5	RSVD
D4	LA_D[21]	E12	LA_WE1#	AA4	RSVD
E4	LA_D[20]	C8	LA_OE#	AA5	RSVD
AB4	RSVD	U2	RSVD	AH7	M[4]_RXD[0]
AB5	RSVD	R28	MDC	AE6	M[3]_RXD[0]
AC4	RSVD	P28	MDIO	AH5	M[2]_RXD[0]
AC5	RSVD	R29	M_CLK	AH2	M[1]_RXD[0]
AD4	RSVD	AC29	RSVD	AF2	M[0]_RXD[0]
AD5	RSVD	AE28	RSVD	AC27	RSVD
W1	RSVD	AJ27	RSVD	AF29	RSVD
Y1	RSVD	AF27	RSVD	AG27	RSVD
Y2	RSVD	AJ25	RSVD	AF26	RSVD
Y3	RSVD	AF24	RSVD	AG25	RSVD
AA1	RSVD	AH23	RSVD	AG23	RSVD
AA2	RSVD	AE19	RSVD	AF23	RSVD
AA3	RSVD	AF21	M[15]_RXD[1]	AG21	RSVD
AB1	RSVD	AJ19	M[14]_RXD[1]	AH21	M[15]_CRS_DV
AB2	RSVD	AF18	M[13]_RXD[1]	AF19	M[14]_CRS_DV
AB3	RSVD	AJ17	M[12]_RXD[1]	AF17	M[13]_CRS_DV
AC1	RSVD	AJ15	M[11]_RXD[1]	AG17	M[12]_CRS_DV
AC2	RSVD	AF15	M[10]_RXD[1]	AG15	M[11]_CRS_DV
AC3	RSVD	AJ13	M[9]_RXD[1]	AF14	M[10]_CRS_DV
AD1	RSVD	AF12	M[8]_RXD[1]	AG13	M[9]_CRS_DV
AD2	RSVD	AJ11	M[7]_RXD[1]	AF11	M[8]_CRS_DV

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AD3	RSVD	AJ9	M[6]_RXD[1]	AG11	M[7]_CRS_DV
N3	RSVD	AF9	M[5]_RXD[1]	AG9	M[6]_CRS_DV
N2	RSVD	AJ7	M[4]_RXD[1]	AF8	M[5]_CRS_DV
N1	RSVD	AF6	M[3]_RXD[1]	AG7	M[4]_CRS_DV
P3	RSVD	AJ5	M[2]_RXD[1]	AF5	M[3]_CRS_DV
P2	RSVD	AJ3	M[1]_RXD[1]	AG5	M[2]_CRS_DV
P1	RSVD	AF1	M[0]_RXD[1]	AH3	M[1]_CRS_DV
R5	RSVD	AC28	RSVD	AF3	M[0]_CRS_DV
R4	RSVD	AF28	RSVD	AD29	RSVD
R3	RSVD	AH27	RSVD	AG28	RSVD
R2	RSVD	AE27	RSVD	AJ26	RSVD
R1	RSVD	AH25	RSVD	AE26	RSVD
T5	RSVD	AE24	RSVD	AJ24	RSVD
T4	RSVD	AF22	RSVD	AE23	RSVD
T3	RSVD	AF20	RSVD	AJ22	RSVD
T2	RSVD	AE21	M[15]_RXD[0]	AJ20	RSVD
T1	RSVD	AH19	M[14]_RXD[0]	AE20	M[15]_TXEN
W3	RSVD	AH20	M[13]_RXD[0]	AJ18	M[14]_TXEN
W2	RSVD	AH17	M[12]_RXD[0]	AJ21	M[13]_TXEN
V1	RSVD	AH15	M[11]_RXD[0]	AJ16	M[12]_TXEN
G1	RSVD	AE15	M[10]_RXD[0]	AJ14	M[11]_TXEN
V3	RSVD	AH13	M[9]_RXD[0]	AE14	M[10]_TXEN
P4	RSVD	AE12	M[8]_RXD[0]	AJ12	M[9]_TXEN
P5	RSVD	AH11	M[7]_RXD[0]	AE11	M[8]_TXEN
V2	RSVD	AH9	M[6]_RXD[0]	AJ10	M[7]_TXEN
U1	RSVD	AE9	M[5]_RXD[0]	AJ8	M[6]_TXEN
AE8	M[5]_TXEN	AH8	M[6]_TXD[0]	G27	RSVD
AJ6	M[4]_TXEN	AF7	M[5]_TXD[0]	H29	RSVD
AE5	M[3]_TXEN	AH6	M[4]_TXD[0]	H28	RSVD
AJ4	M[2]_TXEN	AF4	M[3]_TXD[0]	H27	RSVD
AG1	M[1]_TXEN	AH4	M[2]_TXD[0]	J29	RSVD
AE1	M[0]_TXEN	AG2	M[1]_TXD[0]	J28	RSVD
AD27	RSVD	AE2	M[0]_TXD[0]	J27	RSVD

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AH28	RSVD	U26	RSVD	K29	RSVD
AG26	RSVD	U25	RSVD	K28	RSVD
AE25	RSVD	V26	RSVD	K27	RSVD
AG24	RSVD	V25	RSVD	L29	RSVD
AE22	RSVD	W26	RSVD	L28	RSVD
AJ23	RSVD	W25	RSVD	L27	RSVD
AG20	RSVD	Y27	RSVD	M29	RSVD
AE18	M[15]_TXD[1]	Y26	RSVD	M28	RSVD
AG18	M[14]_TXD[1]	AA26	RSVD	M27	RSVD
AE16	M[13]_TXD[1]	AA25	RSVD	G26	RSVD
AG16	M[12]_TXD[1]	AB26	RSVD	G25	RSVD
AG14	M[11]_TXD[1]	AB25	RSVD	H26	RSVD
AE13	M[10]_TXD[1]	AC26	RSVD	H25	RSVD
AG12	M[9]_TXD[1]	AC25	RSVD	J26	RSVD
AE10	M[8]_TXD[1]	AD26	RSVD	J25	RSVD
AG10	M[7]_TXD[1]	AD25	RSVD	K25	RSVD
AG8	M[6]_TXD[1]	U27	RSVD	K26	RSVD
AE7	M[5]_TXD[1]	V29	RSVD	M25	RSVD
AG6	M[4]_TXD[1]	V28	RSVD	L26	RSVD
AE4	M[3]_TXD[1]	V27	RSVD	M26	RSVD
AG4	M[2]_TXD[1]	W29	RSVD	L25	RSVD
AG3	M[1]_TXD[1]	W28	RSVD	N26	RSVD
AE3	M[0]_TXD[1]	W27	RSVD	N25	RSVD
AD28	RSVD	Y29	RSVD	P26	RSVD
AG29	RSVD	Y28	RSVD	P25	RSVD
AH26	RSVD	Y25	RSVD	F28	RSVD
AF25	RSVD	AA29	RSVD	G28	RSVD
AH24	RSVD	AA28	RSVD	E25	RSVD
AG22	RSVD	AA27	RSVD	G29	RSVD
AH22	RSVD	AB29	RSVD	F29	RSVD
AE17	RSVD	AB28	RSVD	F26	RSVD
AG19	M[15]_TXD[0]	AB27	RSVD	E26	RSVD
AH18	M[14]_TXD[0]	R26	RSVD	F25	RSVD

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
AF16	M[13]_TXD[0]	T25	RSVD	E24	BIST_DONE/TSTOUT[15]
AH16	M[12]_TXD[0]	T26	RSVD	D24	BIST_IN_PRC/TSTOUT[14]
AH14	M[11]_TXD[0]	T28	RSVD	D25	MCT_ERR/TSTOUT[13]
AF13	M[10]_TXD[0]	U28	RSVD	D26	FCB_ERR/TSTOUT[12]
AH12	M[9]_TXD[0]	R25	RSVD	C26	CHECKSUM_OK/TSTOUT[11]
AF10	M[8]_TXD[0]	U29	RSVD	D27	INIT_START/TSTOUT[10]
AH10	M[7]_TXD[0]	T29	RSVD	C27	INIT_DONE/TSTOUT[9]
B27	TSTOUT[8]	U18	VSS	N12	VSS
A27	TSTOUT[7]	V12	VSS	N13	VSS
E28	TSTOUT[6]	V13	VSS	K17	VDD
D28	TSTOUT[5]	V14	VSS	K18	VDD
C28	TSTOUT[4]	V15	VSS	M10	VDD
B28	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	RSVD	N15	VSS	V10	VDD
P29	RSVD	C19	P_DATA15/OE_CLK2	U20	VDD
F3	SCAN_EN	B19	P_DATA14/OE_CLK1	V20	VDD
E1	SCLK	A19	P_DATA13/OE_CLK0	Y12	VDD
U3	T_MODE0	P12	VSS	Y13	VDD
C10	T_MODE1	P13	VSS	Y17	VDD
B24	P_DATA6/RSVD	P14	VSS	Y18	VDD
A21	P_DATA7/TRUNK1	P15	VSS	K12	VDD
C22	P_A2/TRUNK0	P16	VSS	K13	VDD
A26	P_WE/STROBE	N16	VSS	M16	VSS
B26	P_RD/D0	N17	VSS	M17	VSS
C25	P_CS/AUTOFD	N18	VSS	M18	VSS
A24	P_A1/SCL	R13	VSS	F16	VCC
A25	P_A0/SDA	R14	VSS	F17	VCC

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
F1	VDDA	R15	VSS	N6	VCC
D1	VSSA	R16	VSS	P6	VCC
D22	SCANCOL	R17	VSS	R6	VCC
E23	SCANLINK	R18	VSS	T6	VCC
E27	SCANMODE	T12	VSS	U6	VCC
N28	NC	T13	VSS	N24	VCC
N27	NC	T14	VSS	P24	VCC
F2	RESIN#	T15	VSS	R24	VCC
G2	RESETOUT#	T16	VSS	T24	VCC
B22	P_DATA5/MIRROR5	T17	VSS	U24	VCC
A22	P_DATA4/MIRROR4	T18	VSS	AD13	VCC
C23	P_DATA3/MIRROR3	U12	VSS	AD14	VCC
B23	P_DATA2/MIRROR2	U13	VSS	AD15	VCC
A23	P_DATA1/MIRROR1	U14	VSS	AD16	VCC
C24	P_DATA0/MIRROR0	U15	VSS	AD17	VCC
D23	SCANCLK	U16	VSS	F13	VCC
T27	RSVD	U17	VSS	F14	VCC
F27	RSVD	M12	VSS	F15	VCC
C20	P_DATA12/L_CLK2	M13	VSS		
B20	P_DATA11/L_CLK1	M14	VSS		
A20	P_DATA10/L_CLK0	M15	VSS		
C21	P_DATA9/P_D	P17	VSS		
E20	P_DATA8	P18	VSS		
B25	P_INT	R12	VSS		

## 1.4 Signal Mapping and Internal Pull Up/Down Configuration

The ZL50416 Fast Ethernet ports (0-15) support 2 interface options: RMII & GPSI. The table below summarizes the interface signals required for each interface and how they relate back to the Pin Symbol name shown in “Ball – Signal Descriptions” on page 14.

Notes:

I – Input

O – Output

NC - No Connect

<b>Fast Ethernet Ports Pin Symbol</b>	<b>RMII Mode (Bootstrap Mn_TXEN='1')</b>	<b>GPSI Mode (Bootstrap Mn_TXEN='0')</b>
Mn_RXD0	Mn_RXD0 (I)	Mn_RXD (I)
Mn_RXD1	Mn_RXD1 (I)	Mn_RXCLK (I)
Mn_CRS_DV	Mn_CRS_DV (I)	Mn_CRS (I)
Mn_TXD0	Mn_TXD0 (O)	Mn_TXD (O)
Mn_TXD1	Mn_TXD1 (O)	Mn_TXCLK (I)
Mn_TXEN	Mn_TXEN (O)	Mn_TXEN (O)
M_CLK	M_CLK (I)	M_CLK (I)
SCANCLK	NC	SCANCLK (O)
SCANLINK	NC	SCANLINK (IO)
SCANCOL	NC	SCANCOL (IO)

**Table 1 - Fast Ethernet Ports Signal Mapping In Different Operation Mode**

The ZL50416 CPU access support 3 interface options: 8 or 16-bit parallel and unmanaged serial (with optional EEPROM). The table below summarizes the interface signals required for each interface, and how they relate back to the Pin Symbol name shown in “Ball – Signal Descriptions” on page 14.

**Notes:**

I – Input  
 O – Output  
 U – Pull-up  
 D – Pull-down  
 NC – No Connect

<b>Management Interface Pin Symbol</b>	<b>16-bit CPU</b> (Bootstrap TSTOUT6='1' and TSTOUT14='0')	<b>8-bit CPU</b> (Bootstrap TSTOUT6='0' and TSTOUT14='0')	<b>Serial</b> (Bootstrap TSTOUT14='1' and TSTOUT8='1')	<b>Serial, I<sup>2</sup>C</b> (Bootstrap TSTOUT14='1' and TSTOUT8='0')
P_A[0]	P_A[0] (I)	P_A[0] (I)	NC	SDA (IO)
P_A[1]	P_A[1] (I)	P_A[1] (I)	NC	SCL (O)
P_A[2]	P_A[2] (I)	P_A[2] (I)	NC	NC
P_WE#	P_WE# (I)	P_WE# (I)	STROBE (IU)	STROBE (IU)
P_RD#	P_RD# (I)	P_RD# (I)	DATAOUT (O)	DATAOUT (O)
P_CS#	P_CS# (I)	P_CS# (I)	DATAIN (IU)	DATAIN (IU)
P_INT#	P_INT# (O)	P_INT# (O)	NC (O)	NC (O)
P_DATA0	P_DATA0 (IOU)	P_DATA0 (IOU)	NC (U)	NC (U)
P_DATA1	P_DATA1 (IOU)	P_DATA1 (IOU)	NC (U)	NC (U)
P_DATA2	P_DATA2 (IOU)	P_DATA2 (IOU)	NC (U)	NC (U)
P_DATA3	P_DATA3 (IOU)	P_DATA3 (IOU)	NC (U)	NC (U)
P_DATA4	P_DATA4 (IOU)	P_DATA4 (IOU)	NC (U)	NC (U)
P_DATA5	P_DATA5 (IOU)	P_DATA5 (IOU)	NC (U)	NC (U)
P_DATA6	P_DATA6 (IOD)	P_DATA6 (IOD)	NC (D)	NC (D)
P_DATA7	P_DATA7 (IOD)	P_DATA7 (IOD)	NC (D)	NC (D)
P_DATA8	P_DATA8 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA9	P_DATA9 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA10	P_DATA10 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA11	P_DATA11 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA12	P_DATA12 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA13	P_DATA13 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA14	P_DATA14 (IOU)	NC (U)	NC (U)	NC (U)
P_DATA15	P_DATA15 (IOU)	NC (U)	NC (U)	NC (U)

**Table 2 - CPU Interface Signal Mapping in Different Operation Mode**

## 2.0 Block Functionality

### 2.1 Frame Data Buffer (FDB) Interfaces

The FDB interface supports pipelined synchronous burst SRAM (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 M ports at full wire speed switching.

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

### 2.2 MAC Modules

#### 2.2.1 RMII MAC Module (RMAC)

The 10/100 M Media Access Control (RMAC) module provides the necessary buffers and control interface between the Frame Engine (FE) and the external physical device (PHY).

The ZL50416 RMAC implements two interfaces, RMII or GPSI (7WS) (only for 10 M), and fully 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 RMACs are from 08h to 17h. These sixteen ports are denoted as ports 0 to 15.

##### 2.2.1.1 GPSI Interface

The 10/100 M RMII ethernet port can function in GPSI (7WS) mode when the corresponding TXEN pin is strapped low with a 1 K 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.

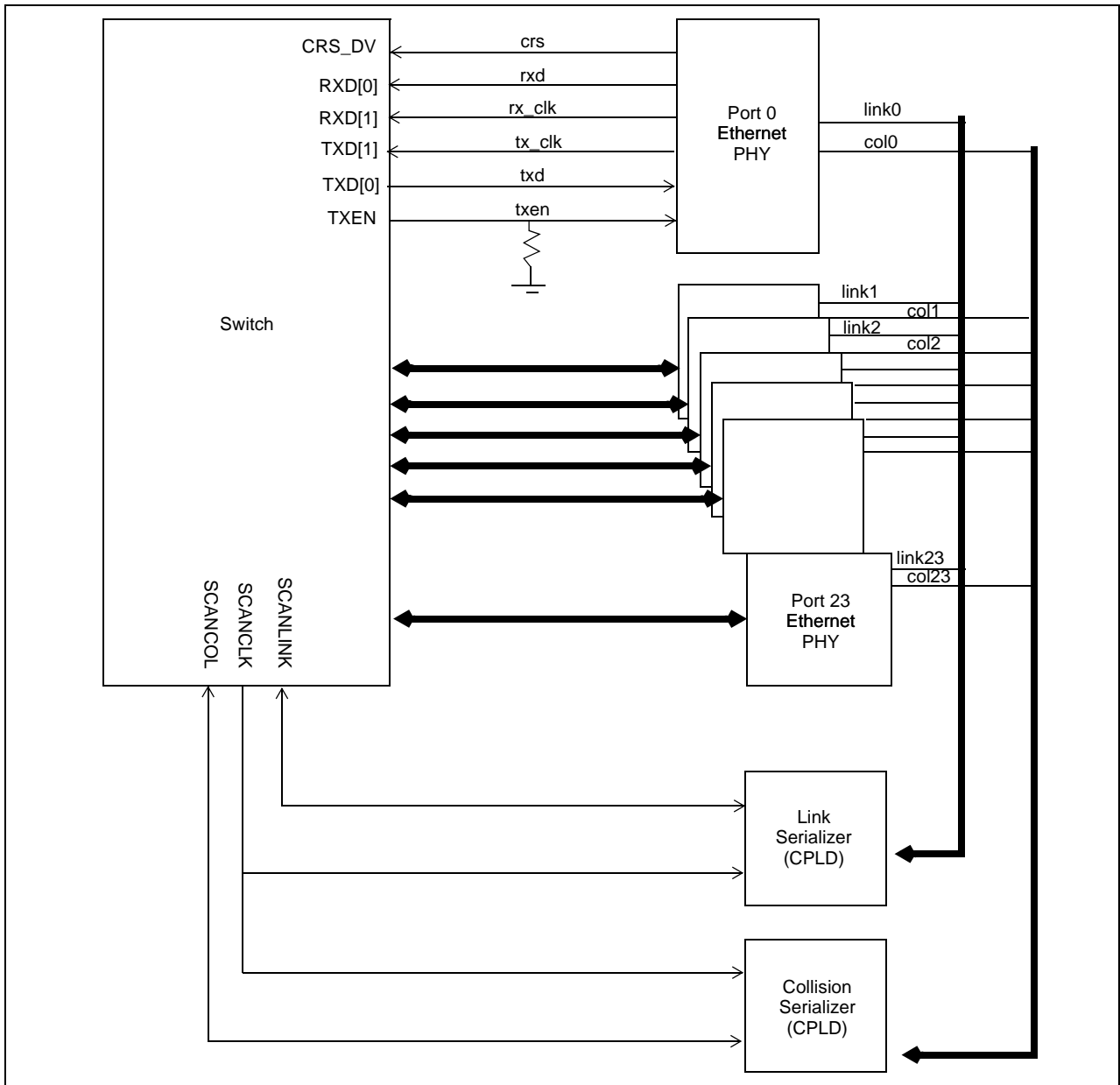


Figure 2 - GPSI (7WS) Mode Connection Diagram

### 2.2.1.2 SCANLINK and SCANCOL 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.

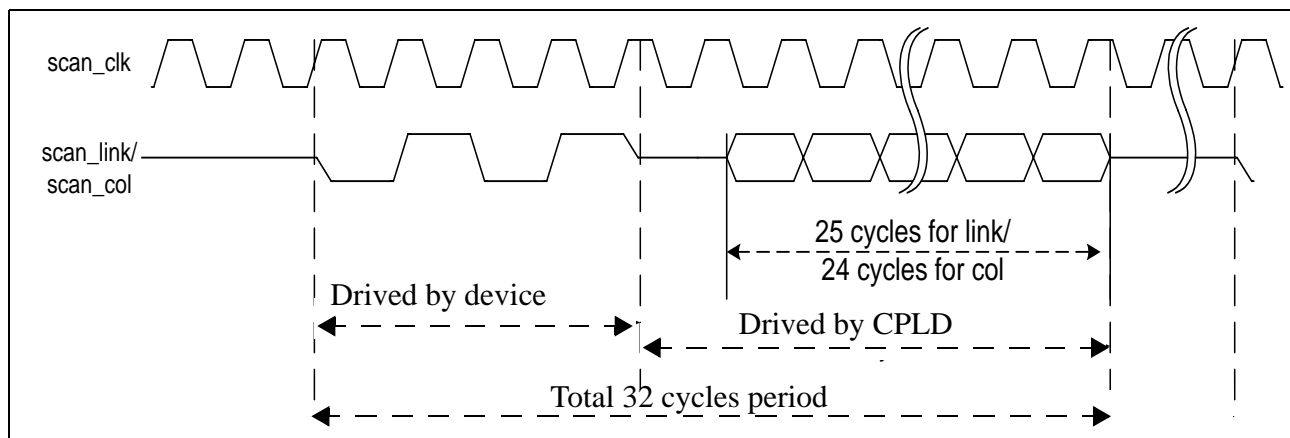


Figure 3 - SCANLINK and SCANCOL Status Diagram

### 2.2.2 CPU MAC Module (CMAC)

The CPU Media Access Control (CMAC) module provides the necessary buffers and control interface between the Frame Engine (FE) and the external CPU device. It support a register access mechanism via the 8/16-bit CPU interface (bootstrap pin TSTOUT6 makes the selection).

The CMAC port is denoted as port 24.

### 2.2.3 PHY Addresses

The table below provides an overview of the PHY addresses required for each port in order for the MDIO auto-negotiation to work between the ZL50416 MAC and the PHY device. If a different PHY address is used, then the port must be manually brought up and the PHY will need to be polled for link status via the MIIC/D registers.

MAC Port	PHY Address
RMAC Port 0	0x08
RMAC Port 1	0x09
...	...
RMAC Port 15	0x17
CMAC Port	N/A

Table 3 - PHY Addresses

## 2.3 Management Module

The CPU can send a control frame to access or configure the internal databases within the ZL50416 device. The Management Module decodes the control frame and executes the functions requested by the CPU. The management module then sends a response or acknowledgment back to the CPU.

This Module is only active in managed mode. In unmanaged mode, no control frame is accepted by the device.

## 2.4 Frame Engine

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

## 2.5 Search Engine

The search engine resolves the frame's destination port or ports by searching the appropriate ZL50416 databases. To achieve its objective, the search engine may use the destination MAC address, IP multicast address (IP multicast packet), and VLAN fields in the packet header. The search engine is also responsible for MAC and VLAN learning, assignment of transmission priority based on IEEE 802.1p or IP TOS/DS fields, and port trunking functions.

## 2.6 LED Interface

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

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.

### 2.6.1 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 (10/100M): cycles #0 to cycle #7

Port 1 (10/100M): cycles#8 to cycle #15

...

Port 14 (10/100M): cycle #112 to cycle #119

Port 15 (10/100M): cycle #120 to cycle #127

RSVD: 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.

The first two bits of byte 26 are reserved while the remainder of byte 26 and byte 27 provides bist status.

26[0]: RSVD

26[1]: RSVD

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

## 2.6.2 LED Interface Timing Diagram

The signal from the ZL50416 to the LED decoder is shown in Figure 7.

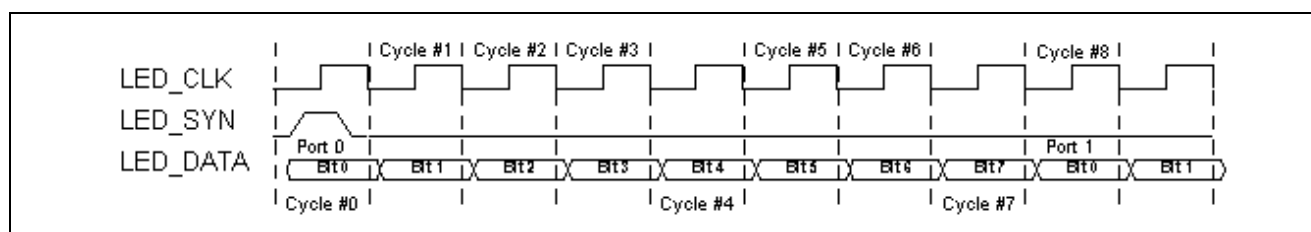


Figure 4 - Timing Diagram of LED Interface

## 2.7 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 SRAM Memory.

## 2.8 Timeout Reset Monitor

The ZL50416 supports a state machine monitoring block which can trigger a reset if any state machine is determined to be stuck in a non-idle state for more than 5 seconds. This feature is enabled via a bootstrap pin (TSTOUT11).

### 3.0 System Configuration (Stand-alone and Stacking)

#### 3.1 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.

#### 3.2 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 5 provides an overview of the CPU interface.

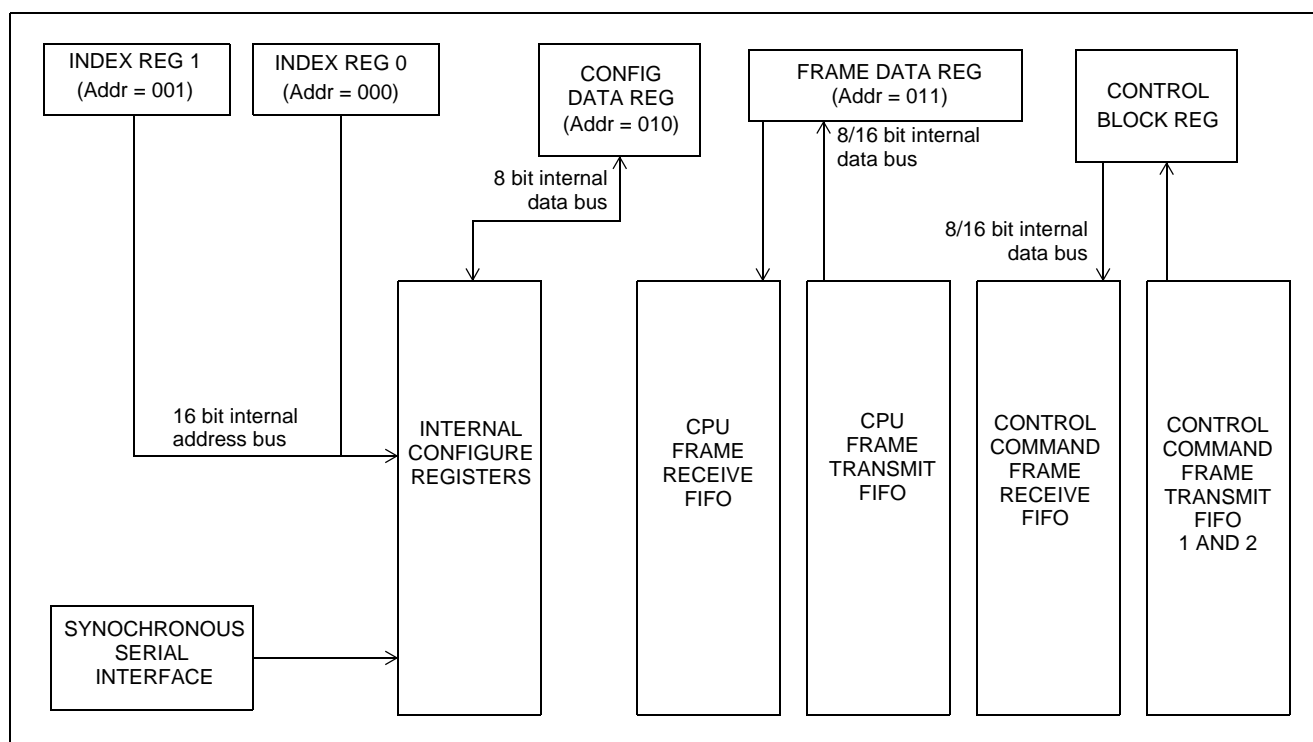


Figure 5 - Overview of the CPU Interface

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

##### 3.2.1.1 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.

### 3.2.1.2 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.

### 3.2.1.3 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.

### 3.3 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.

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

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.

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. The slave address is the memory address of the EEPROM. Refer to "Register Definition" on page 67 for I<sup>2</sup>C address for each register.

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

Figure 6 - Data Transfer Format for I<sup>2</sup>C Interface

##### 3.3.1.1 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.

##### 3.3.1.2 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.

##### 3.3.1.3 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.

### 3.3.1.4 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.

### 3.3.1.5 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.

### 3.3.1.6 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.

## 3.3.2 Synchronous Serial Interface

The synchronous serial interface (SSI) 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

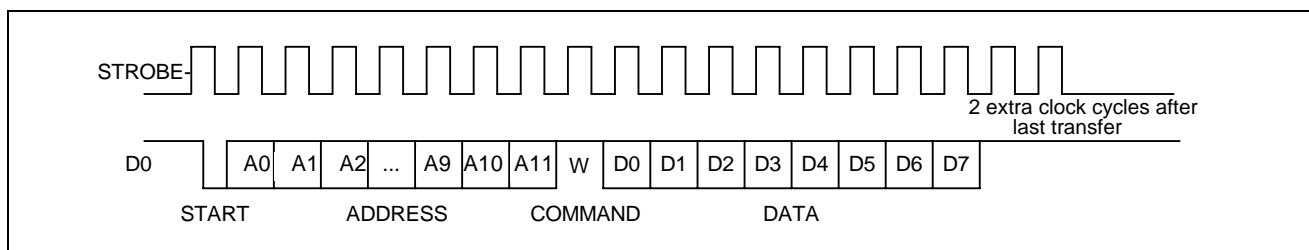
Any command can be aborted in the middle by sending a ABORT pulse to the ZL50416.

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

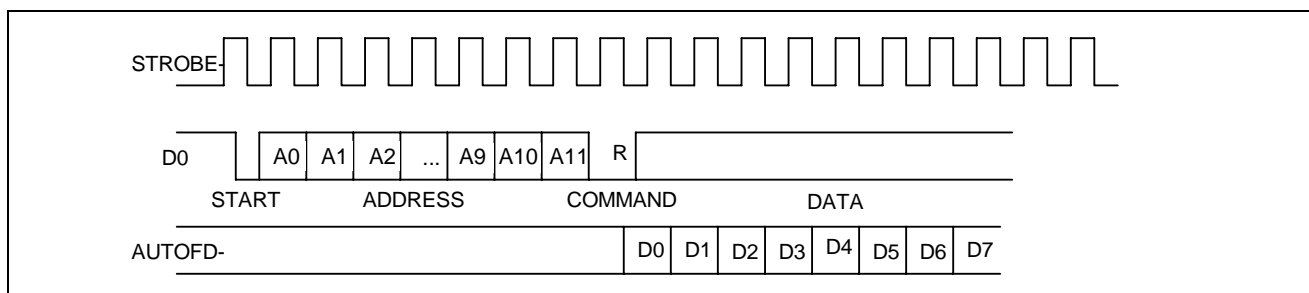
An ABORT command is detected when D0 is sampled low when STROBE rise and D0 is sampled high when STROBE fall.

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

### 3.3.2.1 Write Command



### 3.3.2.2 Read Command



## 4.0 Data Forwarding Protocol

### 4.1 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 consists of one 64-bit bus, connected to one SRAM bank, A. 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 M ports – a total of 64 unicast queues.

The TxQ manager is responsible for scheduling transmission among the queues representing different classes for a port. When the port control module determines that there is room in the MAC Transmission FIFO (Tx FIFO) 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.

## 4.2 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 M ports. The queue with higher priority has room for 32 entries and the queue with lower priority has room for 64 entries. For the 10/100 M 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.

## 4.3 Frame Forwarding To and From CPU

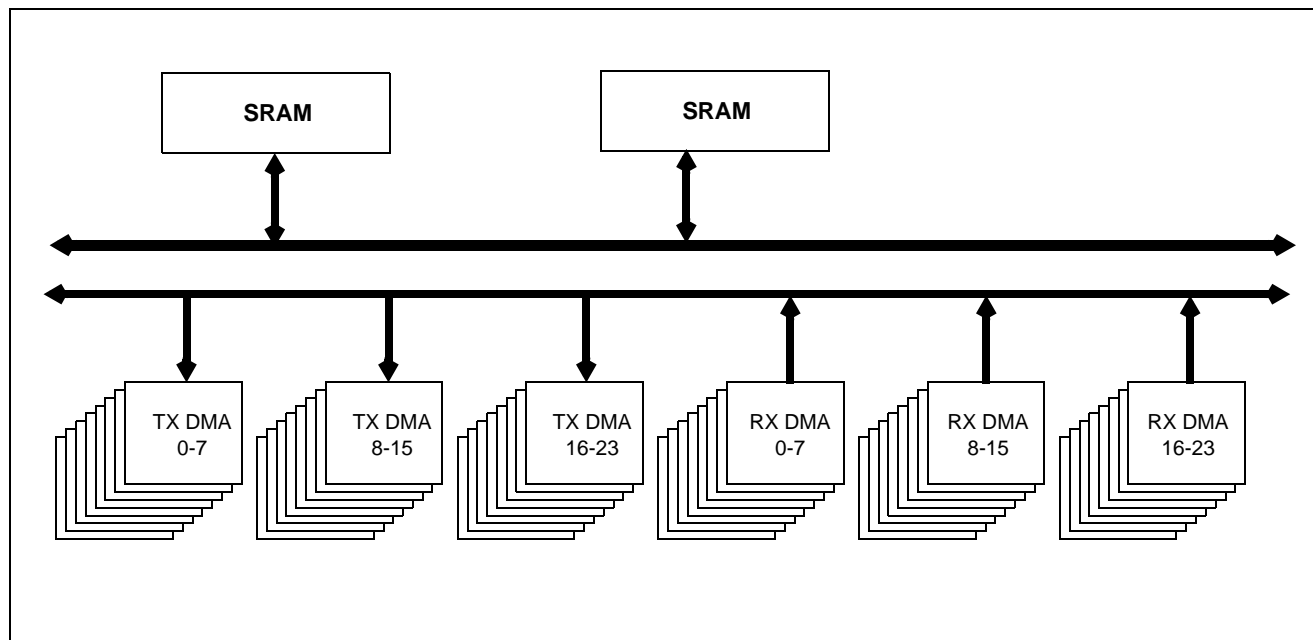
Frame forwarding from the CPU port to a regular transmission port is nearly the same as forwarding between transmission ports. The only difference is that the physical destination port must be indicated in addition to the destination MAC address.

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.

## 5.0 Memory Interface

### 5.1 Overview

The ZL50416 provides one 64-bit wide SRAM bank, SRAM Bank A. Each DMA can read and write from bank A. The following figure provides an overview of the ZL50416 SRAM banks.



**Figure 7 - SRAM Interface Block Diagram (DMAs for 10/100 Ports Only)**

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

### 5.2 Memory Requirements

To support 64 K 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 in the 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.

Bank A	Tagged-based VLAN	Max. Frame Buffers	Max MAC Address
1 M	Disable	0.5 K	32 K
1 M	Enable	0.5 K	31.5 K
2 M	Disable	1 K	64 K
2 M	Enable	1 K	63.5 K

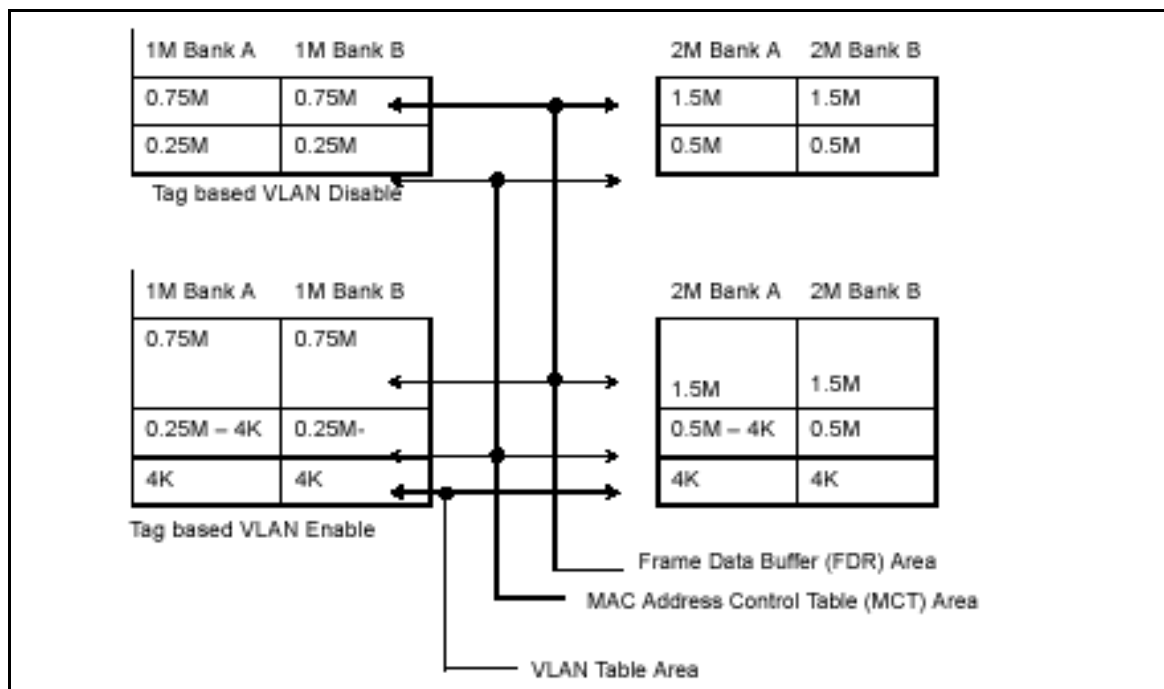


Figure 8 - Memory Configuration

### 5.3 Memory Configurations

The ZL50416 supports pipelined SBRAM with 1 M and 2 M per bank configurations. For detail connection information, please reference the Memory Interface Application Note, MSAN-211.

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

Table 4 - Supported Memory Configurations (SBRAM Mode)

Table 5 - Options for Memory Configuration

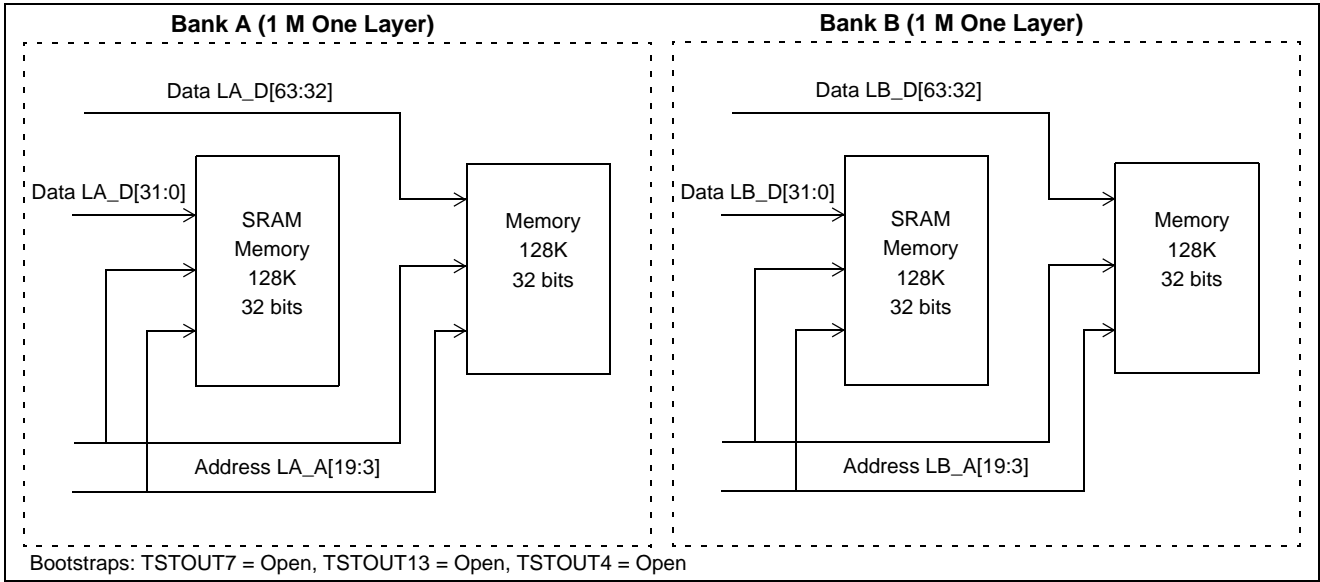


Figure 9 - Memory Configuration For 1 M/bank, 1 Layer

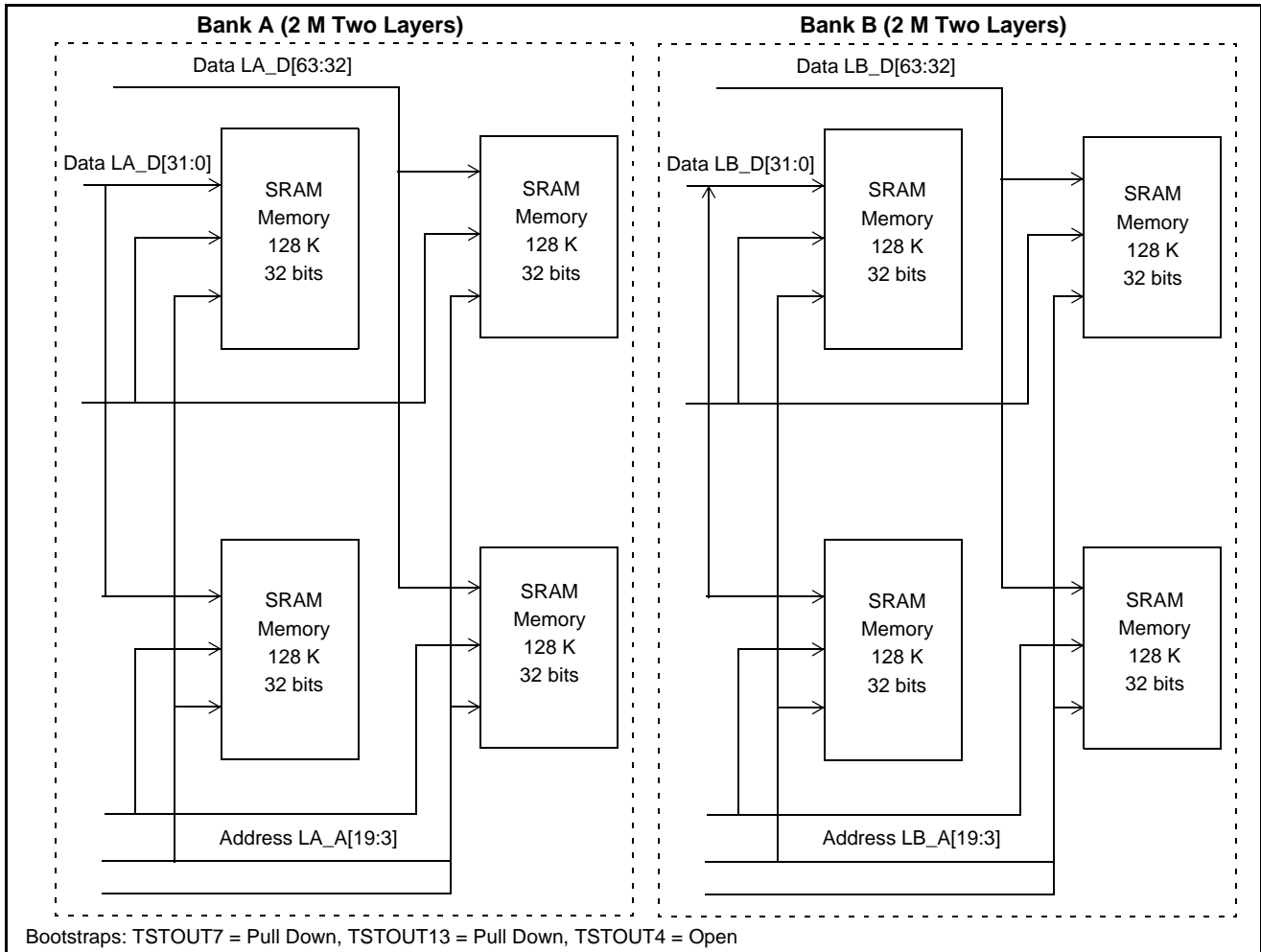
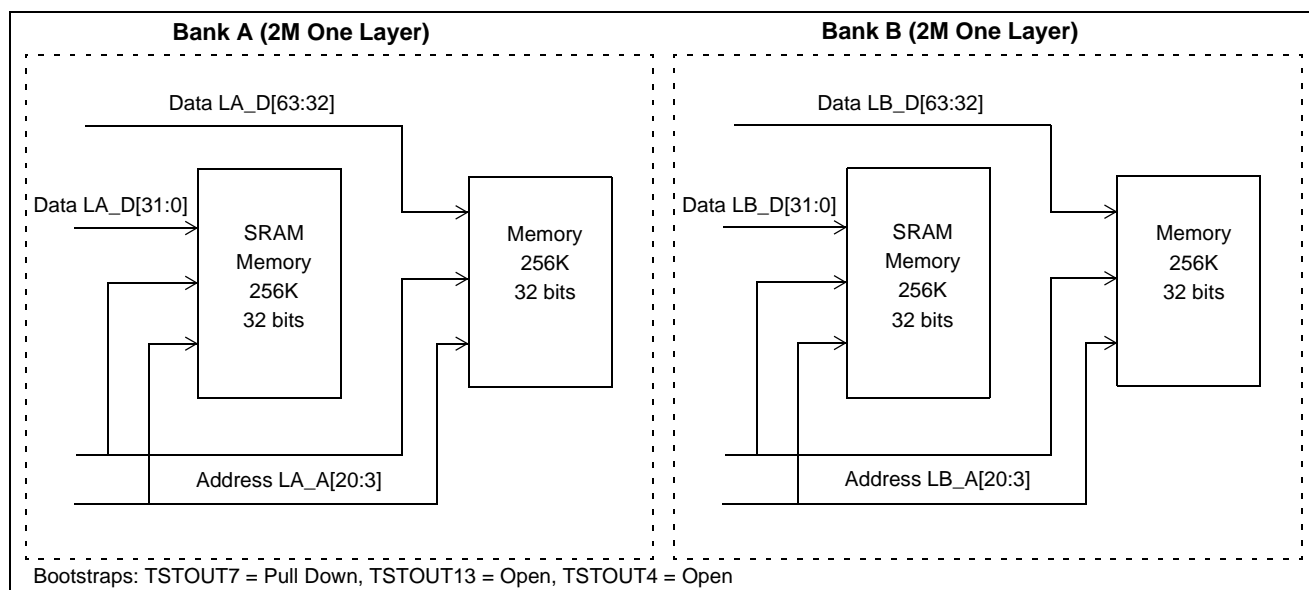


Figure 10 - Memory Configuration For 2 M/bank, 2 Layers



**Figure 11 - Memory Configuration For 2 M/bank, 1 Layer**

## 6.0 Search Engine

### 6.1 Search Engine Overview

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

- Up to 64 K MAC addresses
- Up to 255 tagged-based 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

### 6.2 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.

## **6.3 Search, Learning, and Aging**

### **6.3.1 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.

### **6.3.2 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.

### **6.3.3 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.

## 6.4 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.

## 6.5 Port- and Tagged-Based VLAN

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

### 6.5.1 Port-Based VLAN

An administrator can use the PVMAP registers to configure the ZL50416 for port-based VLAN (See “Register Definition” on page 67.). 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				
	26	...	2	1	0
Register for Port #0 PVMAP00_0[7:0] to PVMAP00_3[2:0]	0		1	1	0
Register for Port #1 PVMAP01_0[7:0] to PVMAP01_3[2:0]	0		1	0	1
Register for Port #2 PVMAP02_0[7:0] to PVMAP02_3[2:0]	0		0	0	0
...					
Register for Port #26 PVMAP26_0[7:0] to PVMAP26_3[2:0]	0		0	0	0

**Table 6 - PVMAP Register**

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.

## 6.5.2 Tagged-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 within the VLAN index (VIX) mapping table. Based on the 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.

For more information on VLANs and details of the VLAN tables, please refer to the IEEE 802.1Q VLAN Setup Application Note, ZLAN-06.

## 6.6 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.

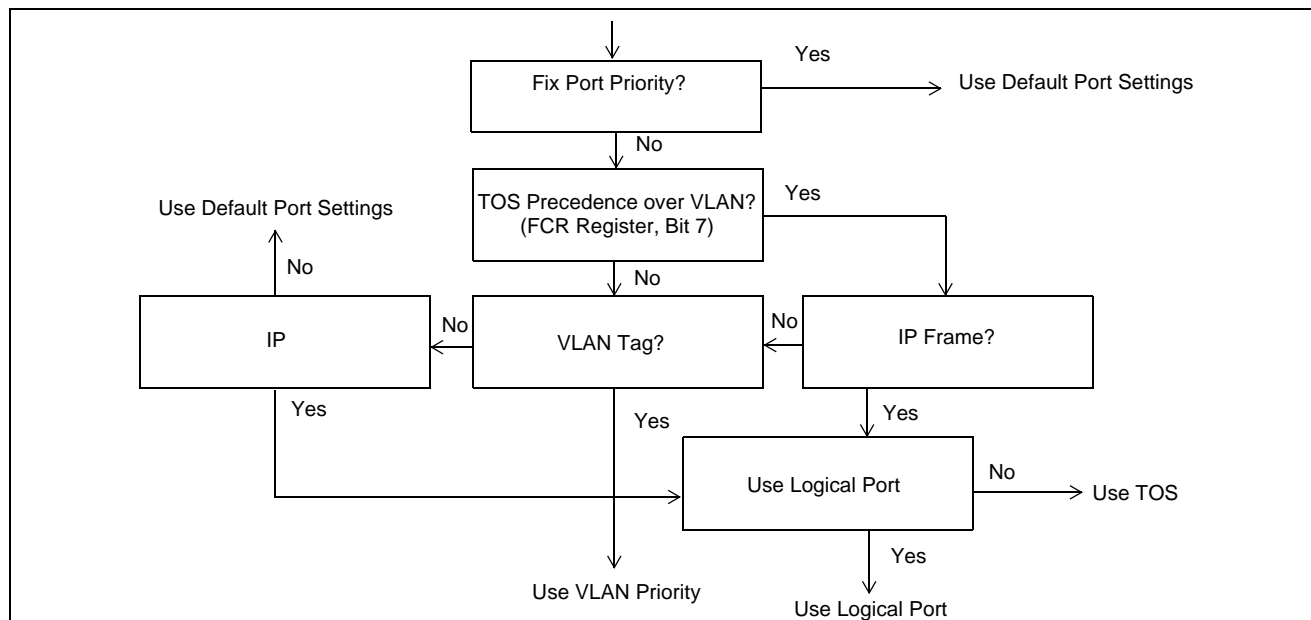
The ZL50416 supports the following QoS techniques:

- 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 behaviours 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.

### 6.6.1 Priority Classification Rule

Figure 12 shows the ZL50416 priority classification rule.



**Figure 12 - Priority Classification Rule**

## 7.0 Frame Engine

### 7.1 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 and a switch response is sent back to the Frame Engine. This response 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 M port, one for each priority. Creation of a queue entry either involves linking a new job to the appropriate linked list if unicast or adding an entry to a physical queue if multicast.

When the port is ready to accept the next frame, the TxQ manager will get the head-of-line (HOL) entry of one of the TxSch Qs, according to the transmission scheduling algorithm (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 M 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.

## 7.2 Frame Engine Details

This section briefly describes the functions of each of the modules of the ZL50416 frame engine.

### 7.2.1 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. 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.

### 7.2.2 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.

### 7.2.3 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.

### 7.2.4 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.

### 7.2.5 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.

### 7.2.6 TxDMA

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

## 8.0 Quality of Service and Flow Control

### 8.1 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 M port will support four total 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.

## 8.2 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 M 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 M Port**

The default configuration for a 10/100 M 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 M 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 an adverse effect on all other classes' performance.

The third configuration for a 10/100 M 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.

## 8.3 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.

## 8.4 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.

## 8.5 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 M 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.

## 8.6 Rate Control

The ZL50416 provides a rate control function on its 10/100 M ports. This rate control function applies to the outgoing traffic aggregate on each 10/100 M 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 M 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 M 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 M 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 5000. 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.

## 8.7 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 $N \geq 120$	$P3 \geq AKB$	$P2 \geq BKB$	$P1 \geq CKB$	X%	0%
Level 2 $N \geq 140$				Y%	Z%
Level 3 $N \geq 160$				100%	100%

**Table 9 - WRED Drop Thresholds**

$P_x$  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  $P3*16+P2*4+P1$ . If using WFQ scheduling,  $N$  equals  $P3+P2+P1$ . 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 RDRC0, RDRC1. 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, ZLAN-05, for more information.

## 8.8 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 13 on page 55.

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 10/100M 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 M 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 ports — 16 ports for Ethernet and one CPU port (port number 24). One parameter can be set for the source port reservation for 10/100 M ports and CPU port. These 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 M and CPU Ports

SFCB- Share FCB Size

C2RS- Class 2 Reserve Size

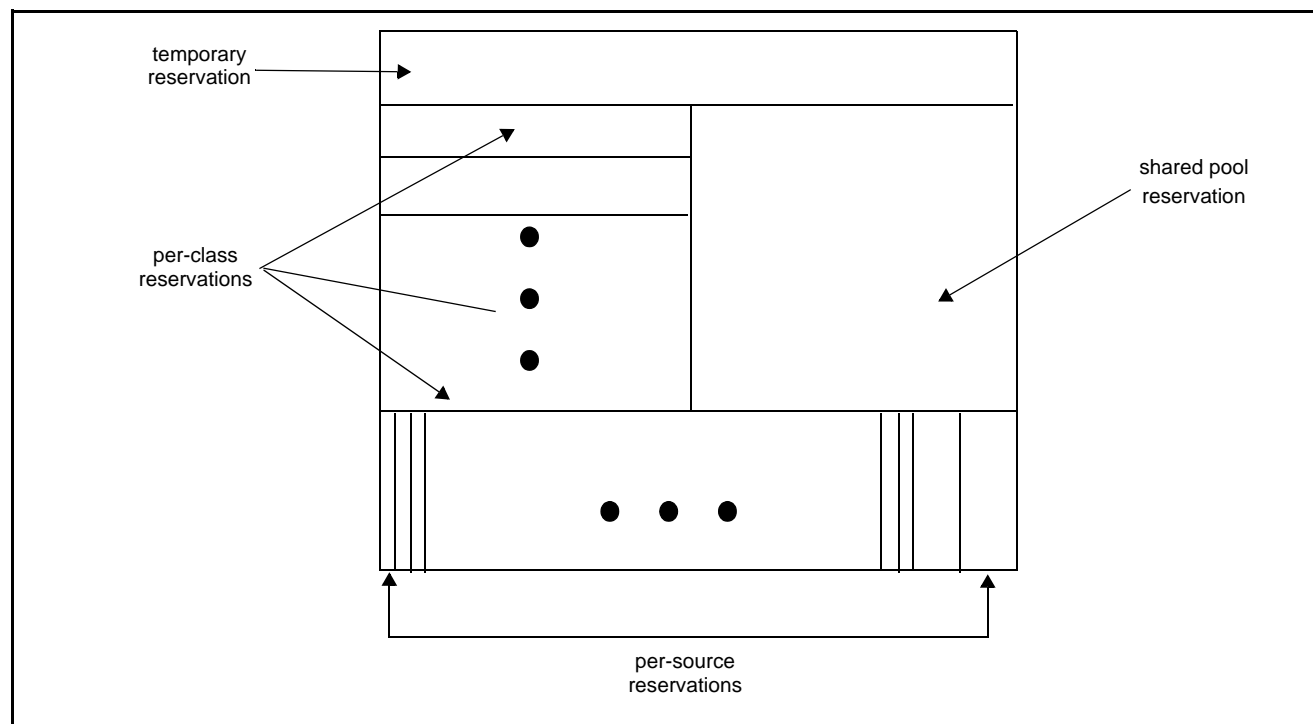
C3RS- Class 3 Reserve Size

C4RS- Class 4 Reserve Size

C5RS- Class 5 Reserve Size

C6RS- Class 6 Reserve Size

C7RS- Class 7 Reserve Size



**Figure 13 - Buffer Partition Scheme Used to Implement Buffer Management**

### 8.8.1 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 multi-level 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 multi-level 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.

### 8.9 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. This means 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.

#### 8.9.1 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.

## 8.9.2 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.

## 8.10 Mapping to IETF DiffServ Classes

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

ZL50416	P3	P2	P1	P0
IETF	NM+EF	AF0	AF1	BE0

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

As Table 10 illustrates, P3 is used for network management (NM) frames and for expedited forwarding service (EF). Classes P2 and P1 correspond to an assured forwarding (AF) group of size 2. Finally, P0 is for best effort (BE) class.

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)	<ul style="list-style-type: none"> <li>Global buffer reservation for NM and EF</li> <li>Option of strict priority scheduling</li> <li>No dropping if admission controlled</li> </ul>
Assured forwarding (AF)	<ul style="list-style-type: none"> <li>Programmable bandwidth partition, with option of WFQ service</li> <li>Option of delay-bounded service keeps delay under fixed levels even if not admission-controlled</li> <li>Random early discard, with programmable levels</li> <li>Global buffer reservation for each AF class</li> </ul>

Best effort (BE)	<ul style="list-style-type: none"> <li>• Service only when other queues are idle means that QoS not adversely affected</li> <li>• Random early discard, with programmable levels</li> <li>• Traffic from flow control enabled ports automatically classified as BE</li> </ul>
------------------	---

**Table 11 - ZL50416 Features Enabling IETF DiffServ Standards**

## 9.0 Port Trunking

### 9.1 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.

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

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

The 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.

### 9.2 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.

### 9.3 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.

## 9.4 Unmanaged Trunking

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

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

Select via trunk0\_mode register

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

Select via trunk1\_mode register

In unmanaged mode, the trunks are individually enabled/disabled by controlling pin TRUNK0,1.

## 10.0 Port Mirroring

### 10.1 Port Mirroring Features

The received or transmitted data of any 10/100 M 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.

Please refer to the Port Mirroring Application Note, MSAN-210, for further details.

### 10.2 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. The default is port 23.

**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.

## 11.0 Hardware Statistics Counter

### 11.1 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 2 2 0  
6 5

	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-l	Flow Control Frames Sent
B[4]	2-u	Non-Unicast Frames Sent
B[5]	3-d	Bytes Received (Good and Bad) (D)
B[6]	4-d	Frames Received (Good and Bad) (D)
B[7]	5-d	Total Bytes Received (D)
B[8]	6-L	Total Frames Received
B[9]	6-U	Flow Control Frames Received
B[10]	7-l	Multicast Frames Received
B[11]	7-u	Broadcast Frames Received
B[12]	8-L	Frames with Length of 64 Bytes
B[13]	8-U	Jabber Frames
B[14]	9-L	Frames with Length Between 65-127 Bytes
B[15]	9-U	Oversize Frames
B[16]	A-l	Frames with Length Between 128-255 Bytes
B[17]	A-u	Frames with Length Between 256-511 Bytes
B[18]	B-l	Frames with Length Between 512-1023 Bytes
B[19]	B-u	Frames with Length Between 1024-1528 Bytes
B[20]	C-l	Fragments
B[21]	C-U1	Alignment Error
B[22]	C-U	Undersize Frames
B[23]	D-l	CRC

B[24]	D-u	Short Event
B[25]	E-l	Collision
B[26]	E-u	Drop
B[27]	F-l	Filtering Counter
B[28]	F-U1	Delay Exceed Discard Counter
B[29]	F-U	Late Collision
B[30]		Link Status Change
B[31]		Current link status

Notation: X-Y

X: Address in the contain memory

Y: Size and bits for the counter

d: D Word counter

L: 24 bits counter bit[23:0]

U: 8 bits counter bit[31:24]

U1: 8 bits counter bit[23:16]

l: 16 bits counter bit[15:0]

u: 16 bits counter bit[31:16]

## 11.2 IEEE 802.3 HUB Management (RFC 1516)

### 11.2.1 Event Counters

#### 11.2.1.1 Readablectet

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

**11.2.1.2 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

**11.2.1.3 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

**11.2.1.4 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

**11.2.1.5 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

**11.2.1.6 ShortEvents**

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

Frame size:  $< 10$  bytes

FCS error: don't care

---

Framing error: don't care

No collisions

#### 11.2.1.7 Runts

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

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

FCS error: don't care

Framing error: don't care

No collisions

#### 11.2.1.8 Collisions

Counts number of collision events.

Frame size: any size

#### 11.2.1.9 LateEvents

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

Frame size: any size

Events are also counted by collision counter

#### 11.2.1.10 VeryLongEvents

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

Frame size: > Jabber

#### 11.2.1.11 DataRateMisatches

For repeaters or HUB application only.

#### 11.2.1.12 AutoPartitions

For repeaters or HUB application only.

#### 11.2.1.13 TotalErrors

Sum of the following errors:

FCS errors

Alignment errors

Frame too long

Short events

Late events

Very long events

### 11.3 IEEE 802.1 Bridge Management (RFC 1286)

#### 11.3.1 Event Counters

##### 11.3.1.1 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.

##### 11.3.1.2 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.

##### 11.3.1.3 InDiscards

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

##### 11.3.1.4 DelayExceededDiscards

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

##### 11.3.1.5 MtuExceededDiscards

Counts number of frames discarded due to excessive size.

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

#### 11.4.1 Event Counters

##### 11.4.1.1 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.

##### 11.4.1.2 Octets

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

##### 11.4.1.3 BroadcastPkts

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

Does not include non-broadcast multicast frames.

##### 11.4.1.4 MulticastPkts

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

Does not include broadcast frames.

**11.4.1.5 CRCAlignErrors**

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

No collisions:

Counts number of frames received with FCS or alignment errors

**11.4.1.6 UndersizePkts**

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

Frame size:  $< 64$  bytes,

No FCS error

No framing error

No collisions

**11.4.1.7 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

**11.4.1.8 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

**11.4.1.9 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

#### 11.4.1.10 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

#### 11.4.1.11 Packet Count for Different Size Groups

Six different size groups – one counter for each:

Pkts64Octets for any packet with size = 64 bytes

Pkts65to127Octets for any packet with size from 65 bytes to 127 bytes

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

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

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

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

Counts both good and bad packets.

### 11.5 Miscellaneous Counters

In addition to the statistics groups defined in previous sections, the ZL50416 has other statistics counters for its own purposes. We have two counters for flow control – one counting the number of flow control frames received, and another counting the number of flow control frames sent. We also have two counters, one for unicast frames sent and one for non-unicast frames sent. A broadcast or multicast frame qualifies as non-unicast. Furthermore, we have a counter called “frame send fail.” This keeps track of FIFO under-runs, late collisions and collisions that have occurred 16 times.

## 12.0 Register Definition

### 12.1 Register Description

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
0. Ethernet Port Control Registers (substitute n with port number (0..Fh))						
ECR1Pn	Port Control Register 1 for Port n	000+2n	R/W	000+n	0C0	
ECR2Pn	Port Control Register 2 for Port n	001+2n	R/W	01B+n	000	
1. VLAN Control Registers (substitute n with port number (0..Fh))						
AVTCL	VLAN Type Code Register Low	100	R/W	036	000	
AVTCH	VLAN Type Code Register High	101	R/W	037	081	
PVMAPn_0	Port n Configuration Register 0	102+4n	R/W	038+n	OFF	
PVMAPn_1	Port n Configuration Register 1	103+4n	R/W	053+n	OFF	
PVMAPn_2	Port n Configuration Register 2	104+4n	R/W	06E+n	OFF	
PVMAPn_3	Port n Configuration Register 3	105+4n	R/W	089+n	007	
PVMODE	VLAN Operating Mode	170	R/W	0A4	000	
PVROUTE[7:0]	VLAN Router Group Enable	171-178	R/W	NA	000	
2. TRUNK Control Registers						
TRUNK0_L	Trunk Group 0 Low	200	R/W	NA	000	
TRUNK0_M	Trunk Group 0 Medium	201	R/W	NA	000	
TRUNK0_H	Trunk Group 0 High	202	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_H	Trunk Group 1 High	20A	R/W	NA	000	
TRUNK1_MODE	Trunk Group 1 Mode	20B	R/W	0A6	003	

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
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	
TRUNK2_MODE	Trunk Group 2 Mode	210	R/W	NA	003	
TRUNK2_HASH0	Trunk Group 2 Hash 0 Destination Port	211	R/W	NA	019	
TRUNK2_HASH1	Trunk Group 2 Hash 1 Destination Port	212	R/W	NA	01A	
MULTICAST_HASHn-0	Multicast hash result n mask byte 0	220+4n	R/W	NA	OFF	n = hash result (0..3)
MULTICAST_HASHn-1	Multicast hash result n mask byte 1	221+4n	R/W	NA	OFF	
MULTICAST_HASHn-2	Multicast hash result n mask byte 2	222+4n	R/W	NA	OFF	
MULTICAST_HASHn-3	Multicast hash result n mask byte 3	223+4n	R/W	NA	OFF	
<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_MASKn	Interrupt Mask for MAC Port 2n, 2n+1	310+n	R/W	NA	000	(n=0..7)
RQS	Receive Queue Select	323	R/W	NA	000	
RQSS	Receive Queue Status	324	RO	NA	NA	
TX_AGE	Transmission Queue Aging Time	325	R/W	0A7	008	

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
4. Search Engine Configurations						
AGETIME_LOW	MAC Address Aging Time Low	400	R/W	0A8	2M:05C/ 4M: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	
SE_OPMODE	Search Engine Operating Mode	403	R/W	NA	000	
SCAN	Scan control register	404	R/W	NA	000	
5. Buffer Control and QOS Control						
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	2M:008/ 4M:010	
MCC	Multicast Congestion Control	50D	R/W	0B7	050	
PR100	Port Reservation for 10/100 Ports	50E	R/W	0B8	2M:035/ 4M:036	
SFCB	Share FCB Size	510	R/W	0BA	2M:046/ 4M:064	
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	

Register	Description	CPU Addr (Hex)	R/W	I <sup>2</sup> C Addr (Hex)	Default	Notes
C7RS	Class 7 Reserve Size	516	R/W	0C0	000	
QOSCn	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=24 - 59)	52F-552	R/W	NA	000	
RDR0	WRED Drop Rate Control 0	553	R/W	0FB	08F	
RDR1	WRED Drop Rate Control 1	554	R/W	0FC	088	
USER_PORTn_LOW	User Define Logical Port n Low	580+2n	R/W	0D6+n	000	(n=0-7)
USER_PORTn_HIGH	User Define Logical Port n High	581+2n	R/W	0DE+n	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	

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

## 12.2 Directly Accessed Registers (8/16-bit Access Only)

<b>INDEX_REG0</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
Used to write the address of the indirect register to be accessed. Data is read from/written to register	8/16-bit	W	0
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
<b>16-bit or serial CPU Interface</b>			
[15:0]	INDEX	W	16-bit address of the indirect register
<b>8-bit CPU Interface</b>			
[7:0]	INDEX_L	W	LSB [7:0] of the 16-bit address of the indirect register

**Register Table 1 - 0, INDEX\_REG0**

<b>INDEX_REG1</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
8-bit CPU Interface Only.	8-bit	W	1
Used to write the address of the indirect register to be accessed. Data is read from/written to register	<b>Default: 00</b>		

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
[7:0]	INDEX_H	W	MSB [15:8] of the 16-bit address of the indirect register

**Register Table 2 - 1, INDEX\_REG1**

<b>DATA_REG</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
Data of indirectly accessed registers	8-bit	R/W	2
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
[7:0]	DATA	R/W	8-bit indirect register data

**Register Table 3 - 2, DATA\_REG**

<b>CPU_FRAME_REG</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
CPU transmit/receive Ethernet frames.	8/16-bit	R/W	3
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
<b>16-bit or serial CPU Interface</b>			
[15:0]	CPU_FRAME	W	Send frame from CPU.  In sequence format: <ul style="list-style-type: none"> <li>• Frame Data (size should be in multiple of 8-byte)</li> <li>• 8-byte of Frame status (Frame size, Destination port #, Frame O.K. status)</li> </ul>
		R	CPU Received frame.  In sequence format: <ul style="list-style-type: none"> <li>• 8-byte of Frame status (Frame size, Source port #, VLAN tag)</li> <li>• Frame Data (size should be in multiple of 8-byte)</li> </ul>
<b>8-bit CPU Interface</b>			
[7:0]	CPU_FRAME	W	Send frame from CPU.  In sequence format: <ul style="list-style-type: none"> <li>• Frame Data (size should be in multiple of 8-byte)</li> <li>• 8-byte of Frame status (Frame size, Destination port #, Frame O.K. status)</li> </ul>
		R	CPU Received frame.  In sequence format: <ul style="list-style-type: none"> <li>• 8-byte of Frame status (Frame size, Source port #, VLAN tag)</li> <li>• Frame Data (size should be in multiple of 8-byte)</li> </ul>

Register Table 4 - 3, CPU\_FRAME\_REG

<b>CMD_STATUS_REG</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
CPU interface commands and status	8-bit	R/W	4
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
[0]	CMD_CONTROL_FRAME_TX_DONE	W	Set Control Frame Receive buffer ready after CPU writes a complete frame into the buffer. This bit is self-cleared.
	STATUS_CONTROL_FRAME_TX_RDY	R	Control Frame receive buffer ready, CPU can write a new frame 1 – CPU can write a new control command 0 – CPU has to wait until this bit is 1 to write a new control command
[1]	CMD_CONTROL_FRAME_BUF1_RX_DONE	W	Set Control Frame Transmit buffer1 ready after CPU reads out a complete frame from the buffer. This bit is self-cleared.
	STATUS_CONTROL_FRAME_RX_BUF1_RDY	R	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
[2]	CMD_CONTROL_FRAME_BUF2_RX_DONE	W	Set Control Frame Transmit buffer2 ready after CPU reads out a complete frame from the buffer. This bit is self-cleared.
	STATUS_CONTROL_FRAME_RX_BUF2_RDY	R	Control Frame transmit buffer2 ready for CPU to read 1 – CPU can read a new control command 2 0 – CPU has to wait until this bit is 1 to read a new control command
[3]	CMD_CPU_FRAME_TX_DONE_AND_FLUSH	W	Set this bit to indicate that the CPU received a whole Ethernet frame (transmit FIFO frame receive done), and flushed the rest of frame fragment, if occurs. This bit is self-cleared.
	STATUS_CPU_FRAME_TX_BUF_RDY	R	Transmit FIFO has data for CPU to read (TXFIFO_RDY)
[4]	CMD_LAST_BYTE_WRITE	W	Set this bit to indicate that the following Write to the Receive FIFO is the last one (EOF). This bit is self-cleared.
	STATUS_CPU_FRAME_RX_BUF_RDY	R	Receive FIFO has space for incoming CPU Ethernet frame (RXFIFO_SPOK)
[5]	CMD_RESTART_RX_FIFO	W	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'.
	STATUS_CPU_FRAME_TX_EOF	R	Transmit FIFO End Of Frame (TXFIFO_EOF)
[7:6]	RSVD	R/W	Reserved

Register Table 5 - 4, CMD\_STATUS\_REG

<b>INT_REG</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
Interrupt sources	8-bit	R/W	5
Note: This register is not self-cleared. After reading CPU has to clear the bit writing 0 to it.	<b>Default: N/A</b>		

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
[0]	INT_CPU_FRAME	R/W	Ethernet frame interrupt. Ethernet Frame receive buffer has data for CPU to read
[1]	INT_CONTROL_FRAME1	R/W	Control Frame 1 interrupt. Control Frame receive buffer1 has data for CPU to read
[2]	INT_CONTROL_FRAME2	R/W	Control Frame 2 interrupt. Control Frame receive buffer2 has data for CPU to read
[7:3]	RSVD	R/W	Reserved

Register Table 6 - 5, INT\_REG

<b>CONTROL_FRM_BUFFER1</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
CPU transmit/receive control frames to/from buffer 1.	8/16-bit	R/W	6
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
<b>16-bit or serial CPU Interface</b>			
[15:0]	CTRL_FRAME1	W	Send frame from CPU.
		R	CPU received frame from buffer 1
<b>8-bit CPU Interface</b>			
[7:0]	CTRL_FRAME1	W	Send frame from CPU.
		R	CPU received frame from buffer 1

Register Table 7 - 6, CONTROL\_FRM\_BUFFER1

<b>CONTROL_FRM_BUFFER2</b>	<b>Width</b>	<b>Access</b>	<b>Address</b>
CPU receive control frames from buffer 2.	8/16-bit	R	7
<b>Default: 00</b>			

<b>Bit #</b>	<b>Name</b>	<b>Type</b>	<b>Description</b>
<b>16-bit or serial CPU Interface</b>			
[15:0]	CTRL_FRAME2	R	CPU received frame from buffer 2
<b>8-bit CPU Interface</b>			
[7:0]	CTRL_FRAME2	R	CPU received frame from buffer 2

Register Table 8 - 7, CONTROL\_FRM\_BUFFER2

## 12.3 Indirectly Accessed Registers

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

#### 12.3.1.1 ECR1Pn: Port n Control Register 1

I<sup>2</sup>C Address 000+n; CPU Address:0000+2n (n = port number)

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

7	6	5	4	0
SS	A-FC	Port Mode		

- Bit [0] 1 - Flow Control Disabled  
0 - Flow Control Enabled (Default)
- Bit [1] 1 - Half Duplex  
0 - Full Duplex (Default)
- Bit [2] 1 - 10 Mbps  
0 - 100 Mbps (Default)
- Bit [4:3] 00 - Enable Auto-Negotiation  
This enables hardware state machine for auto-negotiation. (Default)  
01 - Limited Disable Auto-Negotiation  
This disables hardware state machine for speed auto-negotiation (use ECR1Pn[2:0] for configuration). Hardware will still poll PHY for link status.  
10 - Force Link Down  
Disable the port. Hardware does not talk to PHY.  
11 - Force Link Up  
The configuration in ECR1Pn[2:0] is used for (speed/duplex/flow control) setup. Hardware does not talk to PHY.
- Bit [5] Asymmetric Flow Control Enable.  
0 - Disable asymmetric flow control (Default)  
1 - Enable Asymmetric flow control
- When this bit is set and flow control is on (bit [0] = 0), the device does not send out flow control frames, but it's receiver interprets and processes flow control frames.
- Bit [7:6] SS - Spanning tree state (IEEE 802.1D spanning tree protocol)  
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. (Default)

#### 12.3.1.2 ECR2Pn: Port n Control Register 2

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

Accessed by CPU and serial interface (R/W)

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

- Bit [0]: Filter untagged frame  
 0: Disable (Default)  
 1: All untagged frames from this port are discarded or follow security option when security is enable
- Bit [1]: Filter Tag frame  
 0: Disable (Default)  
 1: All tagged frames from this port are discarded or follow security option when security is enable
- Bit [2]: Learning Disable  
 0: Learning is enabled on this port (Default)  
 1: Learning is disabled on this port
- Bit [3]: Must be '1'
- Bit [5:4] QOS mode selection. Determines which of the 4 sets of QoS settings is used for 10/100 ports.
- 00: select class byte limit set 0 and classes WFQ credit set 0 (Default)
  - 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

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 Programming QOS Registers Application Note, ZLAN-05.

- Bit[7:6] Security Enable. The ZL50416 checks the incoming data for one of the following conditions:
- 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.
  - 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 or the MAC address is not associated to the ingress port.
  - If the port is configured to filter untagged frames and an untagged frame arrives
  - If the port is configured to filter tagged frames and a tagged frame arrives
- If any one of the conditions is met, the packet is forwarded based on these setting.
- 00 – Disable port security, forward packets as usual. (Default)
  - 01 – Discard violating packets
  - 10 – Forward violating packets as usual and also to the CPU for inspection
  - 11 – Forward violating packets to the CPU for inspection

## 12.3.2 (Group 1 Address) VLAN Group

### 12.3.2.1 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**)

### 12.3.2.2 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 0x81**)

### 12.3.2.3 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 0xFF)

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, 2 and 3 to form a 27 bit mask to all egress ports.

In Tagged-based VLAN Mode

Bit [7:0]: PVID [7:0] (Default is 0xFF)

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.

### 12.3.2.4 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 0xFF)

In Tagged-based VLAN Mode

7	5	4	3	0
Unitag Port Priority		Untrust	PVID	

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

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 0x7)**

### 12.3.2.5 PVMAP00\_2 – Port 00 Configuration Register 2

I<sup>2</sup>C Address h6E, CPU Address:h104

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

In Port-based VLAN Mode

Bit [7:0]: • Reserved **(Default FF)**

In Tagged-based VLAN Mode

This register is unused

### 12.3.2.6 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

Bit [0]: VLAN Mask for port 24 (CPU) **(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  
 0 - Disable Ingress Filter. Packets with VLAN not belonging to source port are forwarded, if destination port belongs to the VLAN. Symmetric VLAN. (Default)  
 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 IEEE 802.1Q rule).  
 0 - Disable (Default)  
 1 - Force untagged output. All packets transmitted from this port are untagged.

This bit 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  
 0 – Discard Priority Level 0 (Lowest) (Default)  
 1 – Discard Priority Level 1(Highest)

Bit [7]: Enable Fix Priority (Default 0)  
0 - Disable. All frames are analysed. Transmit Priority and Discard Priority are based on VLAN Tag, TOS or Logical Port.  
1 - Enable. Transmit Priority and Discard Priority are based on values programmed in bit [6:3]

### 12.3.2.7 PVMAPnn\_0,1,2,3 – Port nn Configuration Registers

**PVMAP01\_0,1,2,3** I<sup>2</sup>C Address h39,54,6F,8A; CPU Address:h106,107,108,109 (Port 1)

**PVMAP02\_0,1,2,3** I<sup>2</sup>C Address h3A,55,70,8B; CPU Address:h10A, 10B, 10C, 10D (Port 2)

**PVMAP03\_0,1,2,3** I<sup>2</sup>C Address h3B,56,71,8C; CPU Address:h10E, 10F, 110, 111 (Port 3)

**PVMAP04\_0,1,2,3** I<sup>2</sup>C Address h3C,57,72,8D; CPU Address:h112, 113, 114, 115 (Port 4)

**PVMAP05\_0,1,2,3** I<sup>2</sup>C Address h3D,58,73,8E; CPU Address:h116, 117, 118, 119 (Port 5)

**PVMAP06\_0,1,2,3** I<sup>2</sup>C Address h3E,59,74,8F; CPU Address:h11A, 11B, 11C, 11D (Port 6)

**PVMAP07\_0,1,2,3** I<sup>2</sup>C Address h3F,5A,75,90; CPU Address:h11E, 11F, 120, 121 (Port 7)

**PVMAP08\_0,1,2,3** I<sup>2</sup>C Address h40,5B,76,91; CPU Address:h122, 123, 124, 125 (Port 8)

**PVMAP09\_0,1,2,3** I<sup>2</sup>C Address h41,5C,77,92; CPU Address:h126, 127, 128, 129 (Port 9)

**PVMAP10\_0,1,2,3** I<sup>2</sup>C Address h42,5D,78,93; CPU Address:h12A, 12B, 12C, 12D (Port 10)

**PVMAP11\_0,1,2,3** I<sup>2</sup>C Address h43,5E,79,94; CPU Address:h12E, 12F, 130, 131 (Port 11)

**PVMAP12\_0,1,2,3** I<sup>2</sup>C Address h44,5F,7A,95; CPU Address:h132, 133, 134, 135 (Port 12)

**PVMAP13\_0,1,2,3** I<sup>2</sup>C Address h45,60,7B,96; CPU Address:h136, 137, 138, 139 (Port 13)

**PVMAP14\_0,1,2,3** I<sup>2</sup>C Address h46,61,7C,97; CPU Address:h13A, h13B, 13C, 13D (Port 14)

**PVMAP15\_0,1,2,3** I<sup>2</sup>C Address h47,62,7D,98; CPU Address:h13E, 13F, 140, 141 (Port 15)

**PVMAP24\_0,1,2,3** I<sup>2</sup>C Address h50,6B,86,A1; CPU Address:h162, 163, 164, 165 (Port 24 - CPU port)

### 12.3.2.8 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	MMA	STP	SM0		DF	SL	Vmod

- Bit [0]:
- VLAN Mode (Default = 0)
    - 1 Tagged-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 0180C200000F) 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

### 12.3.2.9 PVROUTE0

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

### 12.3.2.10 PVRROUTE1

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

### 12.3.2.11 PVRROUTE2

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

### 12.3.2.12 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

### 12.3.2.13 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

### 12.3.2.14 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

### 12.3.2.15 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

### 12.3.2.16 PVROUTE7

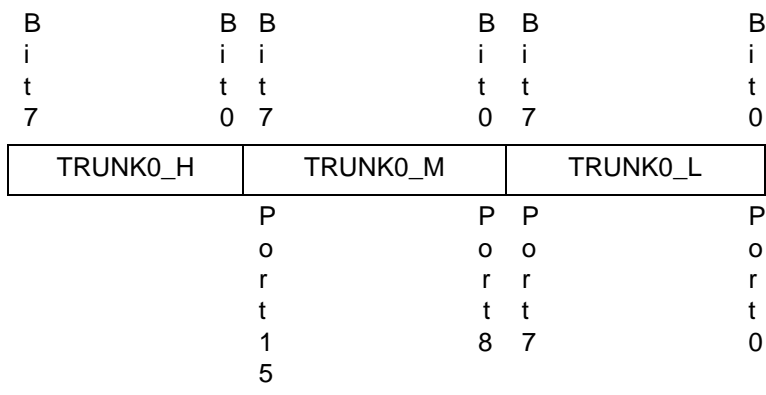
CPU Address:h178

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

### 12.3.3 (Group 2 Address) Port Trunking Groups

Up to four 10/100 ports can be selected for trunk group 0 and 1. TRUNK0\_H, 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.



#### 12.3.3.1 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)

#### 12.3.3.2 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)

#### 12.3.3.3 TRUNK0\_H – Trunk group 0 High (Managed mode only)

CPU Address:h202

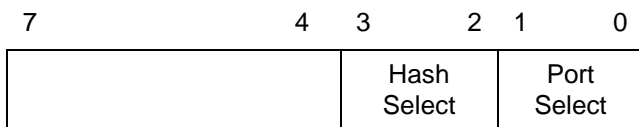
Accessed by CPU, serial interface (R/W)

Bit [7:0]: Reserved (Default 00)

#### 12.3.3.4 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

### 12.3.3.5 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)

### 12.3.3.6 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)

### 12.3.3.7 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)

### 12.3.3.8 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)

### 12.3.3.9 TRUNK1\_L – Trunk group 1 Low (Managed mode only)

CPU Address:h208

Accessed by CPU, serial interface (R/W)

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

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

CPU Address:h209

Accessed by CPU, serial interface (R/W)

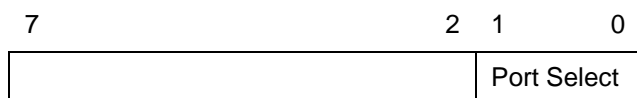
Bit [7:0]: Port15-8 bit map of trunk 1. (Default 00)

**12.3.3.11 TRUNK1\_H – Trunk group 1 High (Managed mode only)**

CPU Address:h20A

Accessed by CPU, serial interface (R/W)

Bit [7:0]: Reserved (Default 00)

**12.3.3.12 TRUNK1\_MODE – Trunk group 1 mode**I<sup>2</sup>C Address h0A6; CPU Address:20BAccessed 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

**12.3.3.13 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)

**12.3.3.14 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)

**12.3.3.15 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)

**12.3.3.16 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)

**12.3.3.17 TRUNK2\_MODE – Trunk group 2 mode**

CPU Address:210

Accessed by CPU, serial interface (R/W)

Bit [:0] Reserved

**12.3.3.18 TRUNK2\_HASH0 – Trunk group 2 hash result 0 destination port number**

CPU Address:h211

Accessed by CPU, serial interface (R/W)

Bit [7:0] Reserved

**12.3.3.19 TRUNK2\_HASH1 – Trunk group 2 hash result 1 destination port number**

CPU Address:h211

Accessed by CPU, serial interface (R/W)

Bit [7:0] Reserved

**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_2	HASH0_1	HASH0_0
Hash Value =1	HASH1_3	HASH1_2	HASH1_1	HASH1_0
Hash Value =2	HASH2_3	HASH2_2	HASH2_1	HASH2_0
Hash Value =3	HASH3_3	HASH3_2	HASH3_1	HASH3_0
	P		P	P P
	o		o	o o
	r		r	r r
	t		t	t t
	2		1	8 7
	4		5	0
	(C			
	P			
	U)			

**12.3.3.20 MULTICAST\_HASHn\_0 – Multicast hash result 0~3 mask byte 0**

CPU Address:h220+n (n=Hash Number)

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**12.3.3.21 MULTICAST\_HASHn\_1 – Multicast hash result 0~3 mask byte 1**

CPU Address:h221+n (n=Hash Number)

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**12.3.3.22 MULTICAST\_HASHn\_2 – Multicast hash result 0~3 mask byte 2**

CPU Address:h222+n (n=Hash Number)

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**12.3.3.23 MULTICAST\_HASHn\_3 – Multicast hash result 0~3 mask byte 3**

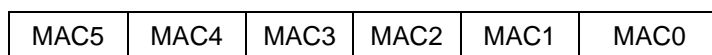
CPU Address:h223+n (n=Hash Number)

Accessed by CPU, serial interface (R/W)

Bit [7:0] (Default FF)

**12.3.4 (Group 3 Address) CPU Port Configuration Group****12.3.4.1 MAC0 – CPU Mac address byte 0**

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.



CPU Address:h300

Accessed by CPU

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

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

CPU Address:h301

Accessed by CPU

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

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

CPU Address:h302

Accessed by CPU

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

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

CPU Address:h303

Accessed by CPU

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

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

CPU Address:h304

Accessed by CPU

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

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

CPU Address:h305

Accessed by CPU

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

**12.3.4.7 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**)

- 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

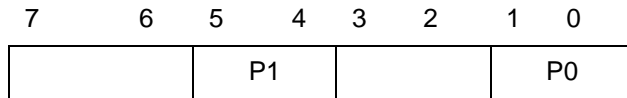
Bit [7:3]: Reserved

### 12.3.4.8 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

### 12.3.4.9 INTP\_MASKn – Interrupt Mask for MAC Ports

INTP\_MASK1 CPU Address:h311 (Ports 2,3)

INTP\_MASK2 CPU Address:h312 (Ports 4,5)

INTP\_MASK3 CPU Address:h313 (Ports 6,7)

INTP\_MASK4 CPU Address:h314 (Ports 8,9)

INTP\_MASK5 CPU Address:h315 (Ports 10,11)

INTP\_MASK6 CPU Address:h316 (Ports 12,13)

INTP\_MASK7 CPU Address:h317 (Ports 14,15)

### 12.3.4.10 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].

### 12.3.4.11 RQSS – Receive Queue Status

CPU Address:h324

Accessed by CPU, serial interface (RO)

7	6	5	4	3	2	1	0
LQ3	LQ2	LQ1	LQ0	NeQ3	NeQ2	NeQ1	NeQ0

CPU receive queue status

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

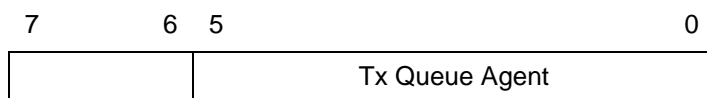
Bit [4]: Head of line entry for Queue 0 is valid for too long. CPU Queue 0 has no WRED threshold.

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.

### 12.3.4.12 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'.

## 12.3.5 (Group 4 Address) Search Engine Group

### 12.3.5.1 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.

### 12.3.5.2 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 X 100µsec). Number of MAC entries = 32K when 1 MB is used per Bank. Number of entries = 64K when 2 MB is used per Bank.

### 12.3.5.3 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 100 msec 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.

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

CPU Address:h403

Accessed by CPU (R/W)

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

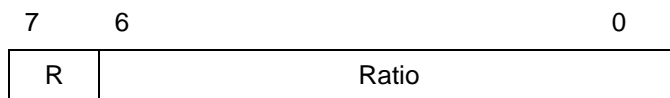
7	6	5	4	3	2	1	0
SL	DMS	ARP	DRA	DA	DRD	DRN	FL

- 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

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

CPU Address h404

Accessed by CPU (R/W)



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

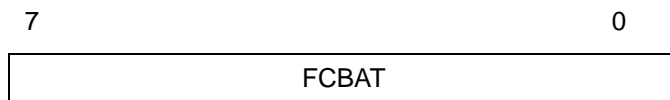
R= 0, Ratio = 7: In eight rounds, one is used for aging and seven are used for scanning

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

#### 12.3.6.1 FCBAT – FCB Aging Timer

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

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

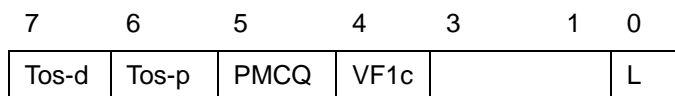


- Bit [7:0]:
- FCB Aging time. Unit of 1ms. (Default FF)
  - This is for buffer aging control. It is used to configure the buffer aging time. This function can be enabled/disabled through bootstrap pin. It is not suggested to use this function for normal operation.

#### 12.3.6.2 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

### 12.3.6.3 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)

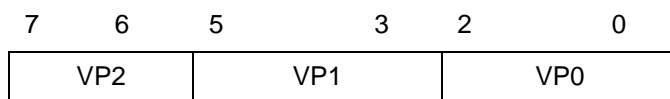
7	6	4	3	0
Tos	TimeBase	U2MR		

- Bit [3:0]:
  - U2MR: Unicast to Multicast Rate. Units in terms of time base defined in bits [6:4]. This is used to limit the amount of flooding traffic. The value in U2MR specifies how many packets are allowed to flood within the time specified by bit [6:4]. To disable this function, program U2MR to 0. (Default = 8)
- Bit [6:4]: Time Base: (Default = 000)
  - 000 = 100 us
  - 001 = 200 us
  - 010 = 400 us
  - 011 = 800 us
  - 100 = 1.6 ms
  - 101 = 3.2 ms
  - 110 = 6.4 ms
  - 111 = 100 us, 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

### 12.3.6.4 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.

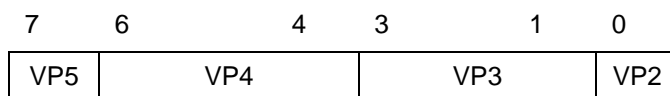
- Bit [2:0]: Priority when the VLAN tag priority field is 0 (**Default 0**)
- Bit [5:3]: Priority when the VLAN tag priority field is 1 (**Default 0**)
- Bit [7:6]: Priority when the VLAN tag priority field is 2 (**Default 0**)

### 12.3.6.5 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:

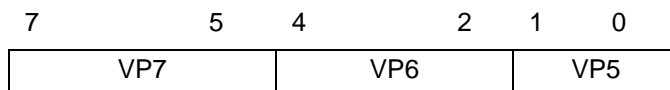


- Bit [0]: Priority when the VLAN tag priority field is 2 (**Default 0**)
- Bit [3:1]: Priority when the VLAN tag priority field is 3 (**Default 0**)
- Bit [6:4]: Priority when the VLAN tag priority field is 4 (**Default 0**)
- Bit [7]: Priority when the VLAN tag priority field is 5 (**Default 0**)

### 12.3.6.6 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**)

### 12.3.6.7 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)

7	6	5	3	2	0
TP2		TP1		TP0	

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)**

### 12.3.6.8 TOSPM – 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)

7	6	4	3	1	0
TP5	TP4		TP3		TP2

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)**

### 12.3.6.9 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)

7	5	4	2	1	0
TP7		TP6		TP5	

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)**

### 12.3.6.10 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)

7	6	5	4	3	2	1	0
FDV7	FDV6	FDV5	FDV4	FDV3	FDV2	FDV1	FDV0

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 <b>(Default 0)</b>
Bit [1]:	Frame drop priority when VLAN Tag priority field is 1 <b>(Default 0)</b>
Bit [2]:	Frame drop priority when VLAN Tag priority field is 2 <b>(Default 0)</b>
Bit [3]:	Frame drop priority when VLAN Tag priority field is 3 <b>(Default 0)</b>
Bit [4]:	Frame drop priority when VLAN Tag priority field is 4 <b>(Default 0)</b>
Bit [5]:	Frame drop priority when VLAN Tag priority field is 5 <b>(Default 0)</b>
Bit [6]:	Frame drop priority when VLAN Tag priority field is 6 <b>(Default 0)</b>
Bit [7]:	Frame drop priority when VLAN Tag priority field is 7 <b>(Default 0)</b>

#### 12.3.6.11 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 <b>(Default 0)</b>
Bit [1]:	Frame drop priority when TOS field is 1 <b>(Default 0)</b>
Bit [2]:	Frame drop priority when TOS field is 2 <b>(Default 0)</b>
Bit [3]:	Frame drop priority when TOS field is 3 <b>(Default 0)</b>
Bit [4]:	Frame drop priority when TOS field is 4 <b>(Default 0)</b>
Bit [5]:	Frame drop priority when TOS field is 5 <b>(Default 0)</b>
Bit [6]:	Frame drop priority when TOS field is 6 <b>(Default 0)</b>
Bit [7]:	Frame drop priority when TOS field is 7 <b>(Default 0)</b>

#### 12.3.6.12 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	

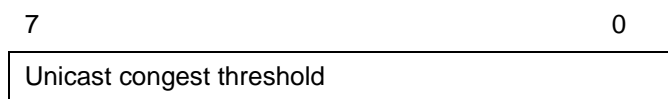
This broadcast and multicast rate defines for each port, the number of packets allowed to be forwarded within a specified time. Once the packet rate is reached, packets will be dropped. To turn off the rate limit, program the field to 0. Time base is based on register FCR [6:4]

- 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)**

### 12.3.6.13 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)

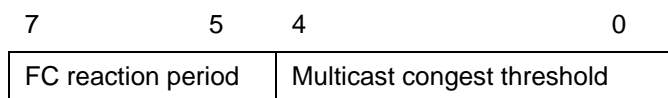


- Bit [7:0] : Number of frame count. Used for best effort dropping at B% when destination port's best effort queue reaches UCC threshold and shared pool is all in use. Granularity 1 frame. (Default: h10 for 2 MB/bank or h08 for 1 MB/bank)

### 12.3.6.14 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)

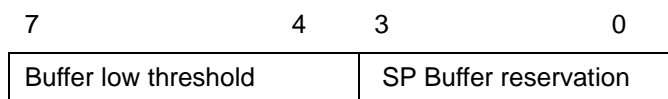


- Bit [4:0]: In multiples of two frames (granularity). Used for triggering MC flow control when destination port's multicast best effort queue reaches MCC threshold.(Default 0x10)
- Bit [7:5]: Flow control reaction period (Default 2) Granularity 4uSec.

### 12.3.6.15 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.

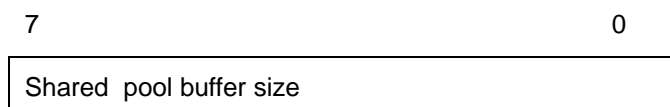
Bits [7:4]: Expressed in multiples of 4 packets. Threshold for dropping all best effort frames when destination port best efforts queues reaches UCC threshold, shared pool is all used and source port reservation is at or below the PR100[7:4] level. Also the threshold for initiating UC flow control.

- Default:
  - h36 for 24+2 configuration with memory 2 MB/bank;
  - h24 for 24+2 configuration with 1MB/bank;

### 12.3.6.16 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)



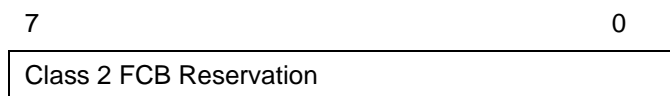
Bits [7:0]: Expressed in multiples of 4 packets. Buffer reservation for shared pool.

- Default:
  - h64 for 24+2 configuration with memory of 2 MB/bank;
  - h14 for 24+2 configuration with memory of 1 MB/bank;

### 12.3.6.17 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)

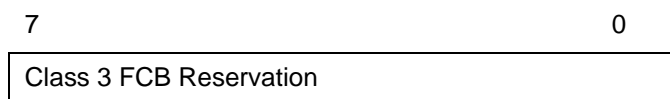


Buffer reservation for class 2 (third lowest priority). Granularity 1. **(Default 0)**

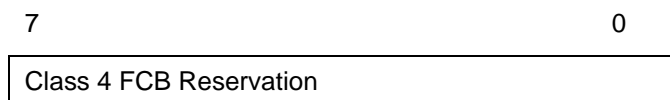
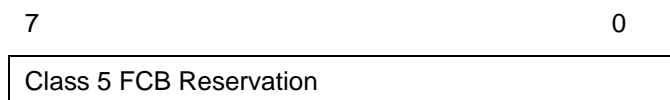
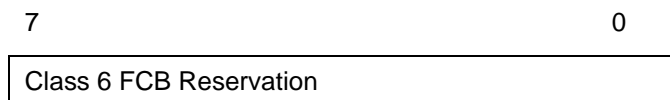
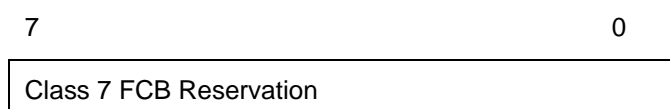
### 12.3.6.18 C3RS – Class 3 Reserve Size

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)**

**12.3.6.19 C4RS – Class 4 Reserve Size**I<sup>2</sup>C Address h0BD, CPU Address 513Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)Buffer reservation for class 4. Granularity 1. **(Default 0)****12.3.6.20 C5RS – Class 5 Reserve Size**I<sup>2</sup>C Address h0BE; CPU Address 514Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)Buffer reservation for class 5. Granularity 1. **(Default 0)****12.3.6.21 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)****12.3.6.22 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)****12.3.6.23 QOSC00~02 - 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.

#### **12.3.6.24 QOSC03~05 - 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.

#### **12.3.6.25 QOSC06~08 - 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

#### **12.3.6.26 QOSC09~11 - 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

### 12.3.6.27 QOSC24~27 - Classes WFQ Credit Set 0

Accessed by CPU and serial interface

W0 - QOSC24[5:0] – CREDIT\_C00 (CPU Address 52f)

W1 - QOSC25[5:0] – CREDIT\_C01 (CPU Address 530)

W2 - QOSC26[5:0] – CREDIT\_C02 (CPU Address 531)

W3 - 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 W3 and QOSC24 corresponds to W0.

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.

### 12.3.6.28 QOSC28~31 - Classes WFQ Credit Set 1

Accessed by CPU and serial interface

W0 - QOSC28[5:0] – CREDIT\_C10 (CPU Address 533)

W1 - QOSC29[5:0] – CREDIT\_C11 (CPU Address 534)

W2 - QOSC30[5:0] – CREDIT\_C12 (CPU Address 535)

W3 - 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 W3 and QOSC28 corresponds to W0.

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

### 12.3.6.29 QOSC32~35 - Classes WFQ Credit Set 2

Accessed by CPU and serial interface

W0 - QOSC32[5:0] – CREDIT\_C20 (CPU Address 537)

W1 - QOSC33[5:0] – CREDIT\_C21 (CPU Address 538)

W2 - QOSC34[5:0] – CREDIT\_C22 (CPU Address 539)

W3 - 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 W3 and QOSC32 corresponds to W0.

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

### 12.3.6.30 QOSC36~39 - Classes WFQ Credit Set 3

Accessed by CPU and serial interface

W0 - QOSC36[5:0] – CREDIT\_C30 (CPU Address 53b)

W1 - QOSC37[5:0] – CREDIT\_C31 (CPU Address 53c)

W2 - QOSC38[5:0] – CREDIT\_C32 (CPU Address 53d)

W3 - 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 W0.

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

### 12.3.6.31 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

### 12.3.6.32 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)

7	4	3	0
Z Rate		B Rate	

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

## 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:

- 23
- 512
- 6000
- 443
- 111
- 22555
- 22
- 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.

### 12.3.6.33 USER\_PORT0~7)\_L/H – USER DEFINE LOGICAL PORT (0~7)

USER\_PORT0\_L/H - I<sup>2</sup>C Address h0D6 + 0DE; CPU Address 580(Low) + 581(high)

USER\_PORT1\_L/H - I<sup>2</sup>C Address h0D7 + 0DF; CPU Address 582 + 583

USER\_PORT2\_L/H - I<sup>2</sup>C Address h0D8 + 0E0; CPU Address 584 + 585

USER\_PORT3\_L/H - I<sup>2</sup>C Address h0D9 + 0E1; CPU Address 586 + 587

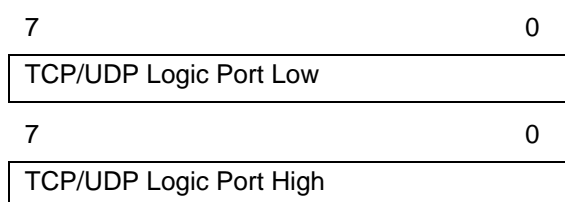
USER\_PORT4\_L/H - I<sup>2</sup>C Address h0DA + 0E2; CPU Address 588 + 589

USER\_PORT5\_L/H - I<sup>2</sup>C Address h0DB + 0E3; CPU Address 58A + 58B

USER\_PORT6\_L/H - I<sup>2</sup>C Address h0DC + 0E4; CPU Address 58C + 58D

USER\_PORT7\_L/H - 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.

**12.3.6.34 USER\_PORT\_[1:0]\_PRIORITY - User Define Logic Port 1 and 0 Priority**I<sup>2</sup>C Address h0E6, CPU Address 590Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

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

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)

**12.3.6.35 USER\_PORT\_[3:2]\_PRIORITY - User Define Logic Port 3 and 2 Priority**I<sup>2</sup>C Address h0E7, CPU Address 591Accessed by CPU, serial interface and I<sup>2</sup>C (R/W)

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

**12.3.6.36 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)

**12.3.6.37 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)

**12.3.6.38 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)

**12.3.6.39 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)

**12.3.6.40 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)

**12.3.6.41 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)

**12.3.6.42 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)

**12.3.6.43 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)

**12.3.6.44 RLOWL – USER DEFINE RANGE LOW BIT 7:0**I<sup>2</sup>C Address h0F4, CPU Address: 59aAccessed by CPU, serial interface and I<sup>2</sup>C (R/W)

[7:0] Lower 8 bit of the User Define Logical Port Low Range (Default 00)

**12.3.6.45 RLOWH – User Define Range Low Bit 15:8**I<sup>2</sup>C Address h0F5, CPU Address: 59bAccessed by CPU, serial interface and I<sup>2</sup>C (R/W)

[7:0] Upper 8 bit of the User Define Logical Port Low Range (Default 00)

**12.3.6.46 RHIGHL – User Define Range High Bit 7:0**I<sup>2</sup>C Address h0D3, CPU Address: 59cAccessed by CPU, serial interface and I<sup>2</sup>C (R/W)

[7:0] Lower 8 bit of the User Define Logical Port High Range (Default 00)

**12.3.6.47 RHIGHH – User Define Range High Bit 15:8**I<sup>2</sup>C Address h0D4, CPU Address: 59dAccessed by CPU, serial interface and I<sup>2</sup>C (R/W)

[7:0] Upper 8 bit of the User Define Logical Port High Range (Default 00)

**12.3.6.48 RRIORITY – User Define Range Priority**I<sup>2</sup>C Address h0D5, CPU Address: 59eAccessed by CPU, serial interface and I<sup>2</sup>C (R/W)

7	4	3	0
Range Transmit Priority			Drop

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

### 12.3.6.49 CPUQOSC1,2,3

CPU Address: 5a0, 5a1, 5a2

Accessed by CPU and serial interface (R/W)

C - CPUQOSC1 – CPU BYTE\_C1 (CPU Address 5A0)

B - CPUQOSC2 – CPU BYTE\_C2 (CPU Address 5A1)

A - CPUQOSC3 – CPU BYTE\_C3 (CPU Address 5A2)

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

### 12.3.7 (Group 6 Address) MISC Group

#### 12.3.7.1 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 10 Mbps.

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.

#### 12.3.7.2 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)

### 12.3.7.3 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; Must also set ECR2Pn[3] = 1

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

#### 12.3.7.4 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.

#### 12.3.7.5 MIIC1 – MII Command Register 1

CPU Address:h604

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

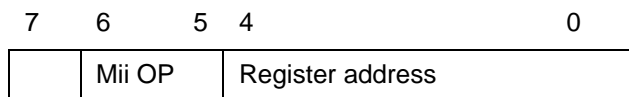
Bit [7:0] - MII Data [15:8]

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

#### 12.3.7.6 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.

### 12.3.7.7 MIIC3 – MII Command Register 3

CPU Address:h606

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

7	6	5	4	0
Rdy	Valid		PHY address	

Bits [4:0] - PHY\_AD – 5 Bit PHY Address

Bit [6] - VALID – Data Valid from PHY (Read Only)

Bit [7] - RDY – Data is returned from PHY (Ready Only)

**Note:** Before programming MII command: set FEN[6], check MIIC3, making sure no RDY and no VALID; then program MII command. Writing this register will initiate a serial management cycle to the MII management interface.

### 12.3.7.8 MIID0 – MII Data Register 0

CPU Address:h607

Accessed by CPU and serial interface only (RO)

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

### 12.3.7.9 MIID1 – MII Data Register 1

CPU Address:h608

Accessed by CPU and serial interface only (RO)

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

### 12.3.7.10 LED 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=8 msec	01=16 msec
10=32 msec	11=64 msec

Bit [4:3]: LED clock frequency (**Default 0**)

For 100MHz SCLK

00 = 100MHz/8 = 12.5 MHz	01 = 100MHz/16 = 6.25 MHz
10 = 100MHz/32 = 3.125 MHz	11 = 100MHz/64 = 1.5625 MHz

For 125 MHz SCLK

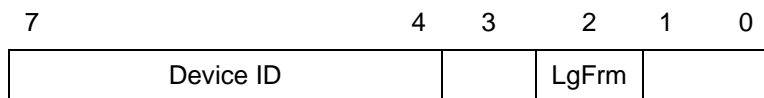
00 = 125MHz/64 = 1953 KHz	01 = 125MHz/128 = 977 KHz
10 = 125MHz/512 = 244 KHz	11 = 125MHz/1024 = 122 KHz

Bit [7:5]: Reserved. Must be set to '0' (**Default 0**)

### 12.3.7.11 DEVICE Mode

CPU Address:h60a

Accessed by CPU and serial interface (R/W)



- Bit [1:0]: Reserved. Must be set to '0' (Default 0)
- Bit [2]: Support < = 1536 frames  
 0: < = 1518 bytes (< = 1522 bytes with VLAN tag) (Default)  
 1: < = 1536 bytes
- Bit [3]: Reserved. Must be set to '0' (**Default 0**)
- Bit [7:4]: DEVICE ID (**Default 0**). This is for stacking operation. This is the stack ID for loop topology.

### 12.3.7.12 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)

Bit [7:0]: (**Default 0**)

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:

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

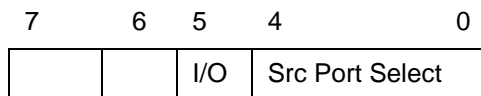
When the ZL50416 boots from the EEPROM the checksum is calculated and the value must be zero. If the checksum is not zeroed the ZL50416 does not start and pin CHECKSUM\_OK is set to zero.

## 12.3.8 (Group 7 Address) Port Mirroring Group

### 12.3.8.1 MIRROR1\_SRC - Port Mirror source port

CPU Address 700

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



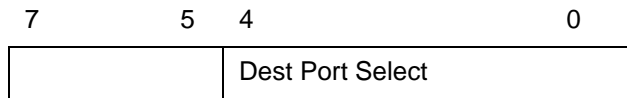
- Bit [4:0]: Source port to be mirrored. Use illegal port number to disable mirroring
- Bit [5]: 0 – select ingress data  
 1 – select egress data

- Bit [6] Reserved  
 Bit [7] Reserved must be set to '1'

### 12.3.8.2 MIRROR1\_DEST – Port Mirror destination

CPU Address 701

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



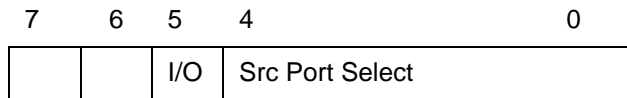
Bit [4:0]: Port Mirror Destination

When port mirroring is enable, destination port can not serve as a data port.

### 12.3.8.3 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]: 0 – select ingress data  
 1 – select egress data

Bit [6] Reserved

Bit [7] Reserved must be set to '1'

### 12.3.8.4 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.

### 12.3.9 (Group F Address) CPU Access Group

#### 12.3.9.1 GCR-Global Control Register

CPU Address: hF00

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

7	5	4	3	2	1	0
		Init	Reset	Bist	SR	SC

- Bit [0]: Store configuration (**Default = 0**)  
Write '1' followed by '0' to store configuration into external EEPROM
- Bit [1]: Store configuration and reset (**Default = 0**)  
Write '1' to store configuration into external EEPROM and reset chip
- Bit [2]: Start BIST (Default = 0)  
Write '1' followed by '0' to start the device's built-in self-test. The result is found in the DCR register.
- Bit [3]: Soft Reset (Default = 0)  
Write '1' to reset 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.

### 12.3.9.2 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  
10: ZL50416 device
- Bit [7:6]: Revision  
00: Initial Silicon  
01: XA1 Silicon  
10: Production Silicon

### 12.3.9.3 DCR1 - Chip Status

CPU Address: hF02

Accessed by CPU and serial interface. (RO)

7	6	4	3	2	1	0
CIC						

- Bit [7] Chip initialization completed

---

### 12.3.9.4 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)

### 12.3.9.5 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.

7	6	5	4	3	2	1	0
				Inkdn	FE	Fdpx	FcEn

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 [4] – Reserved

Bit [5] – Reserved

Bit [6] - Reserved

Bit [7] – Reserved

### 12.3.9.6 DA – Dead or Alive 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.

---

## 12.4 Characteristics and Timing

### 12.4.1 Absolute Maximum Ratings

Storage Temperature	-65°C to +150°C
Operating Temperature	-40°C to +85°C
Maximum Junction Temperature	+125°C
Supply Voltage $V_{CC}$ with Respect to $V_{SS}$	+3.0V to +3.6V
Supply Voltage $V_{DD}$ with Respect to $V_{SS}$	+2.38V to +2.75V
Voltage on Input Pins	+0.5V to ( $V_{CC} + 3.3V$ )

**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.

### 12.4.2 DC Electrical Characteristics

$$V_{CC} = 3.3V \pm 10\%$$

$$T_{AMBIENT} = -40^{\circ}C \text{ to } +85^{\circ}C$$

$$V_{DD} = 2.5V +10\% / -5\%$$

### 12.4.3 Recommended Operating Conditions

Symbol	Parameter Description	Min.	Typ.	Max.	Unit
$f_{osc}$	Frequency of Operation		100		MHz
$I_{CC}$	Supply Current – @ 100 MHz ( $V_{CC}=3.3$ V)			250	mA
$I_{DD}$	Supply Current – @ 100 MHz ( $V_{DD}=2.5$ 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 5 V tolerant)	2.0		$V_{CC} + 2.0$	V
$V_{IL-TTL}$	Input Low Voltage (TTL 5 V tolerant)			0.8	V
$I_{IL}$	Input Leakage Current ( $0.1$ V < $V_{IN}$ < $V_{CC}$ ) (all pins except those with internal pull-up/pull-down resistors)			10	$\mu$ A
$I_{OL}$	Output Leakage Current ( $0.1$ V < $V_{OUT}$ < $V_{CC}$ )			10	$\mu$ 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 2 m/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

## 12.5 AC Characteristics and Timing

### 12.5.1 Typical Reset & Bootstrap Timing Diagram

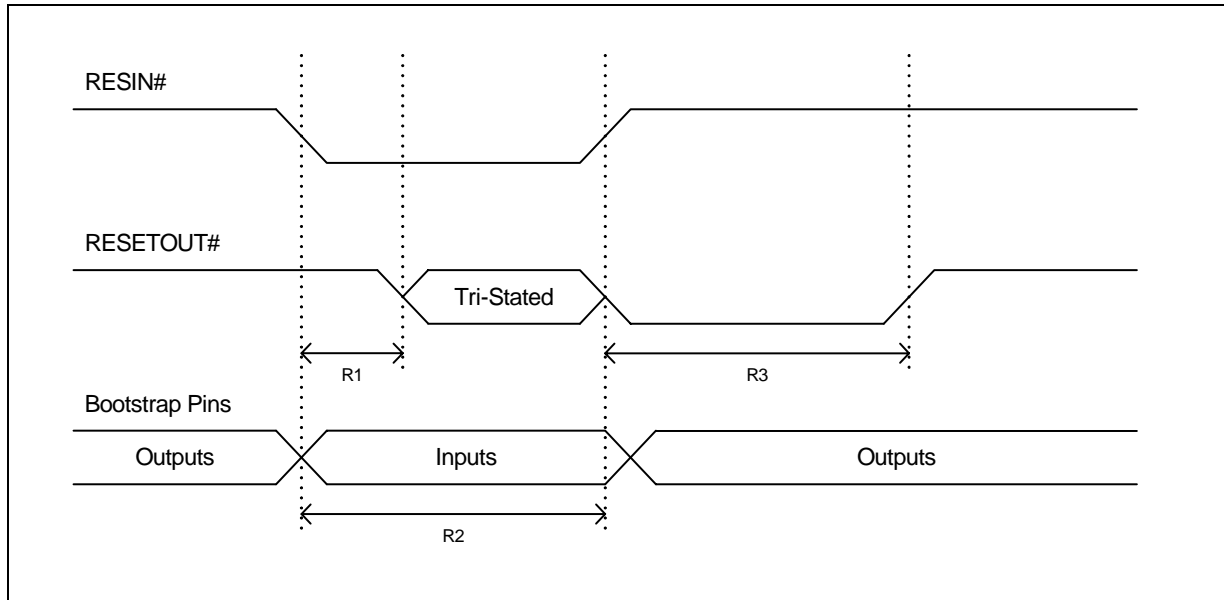


Figure 14 - Typical Reset & Bootstrap Timing Diagram

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

Table 14 - Reset & Bootstrap Timing

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

### 12.5.2 Typical CPU Timing Diagram for a CPU Write Cycle

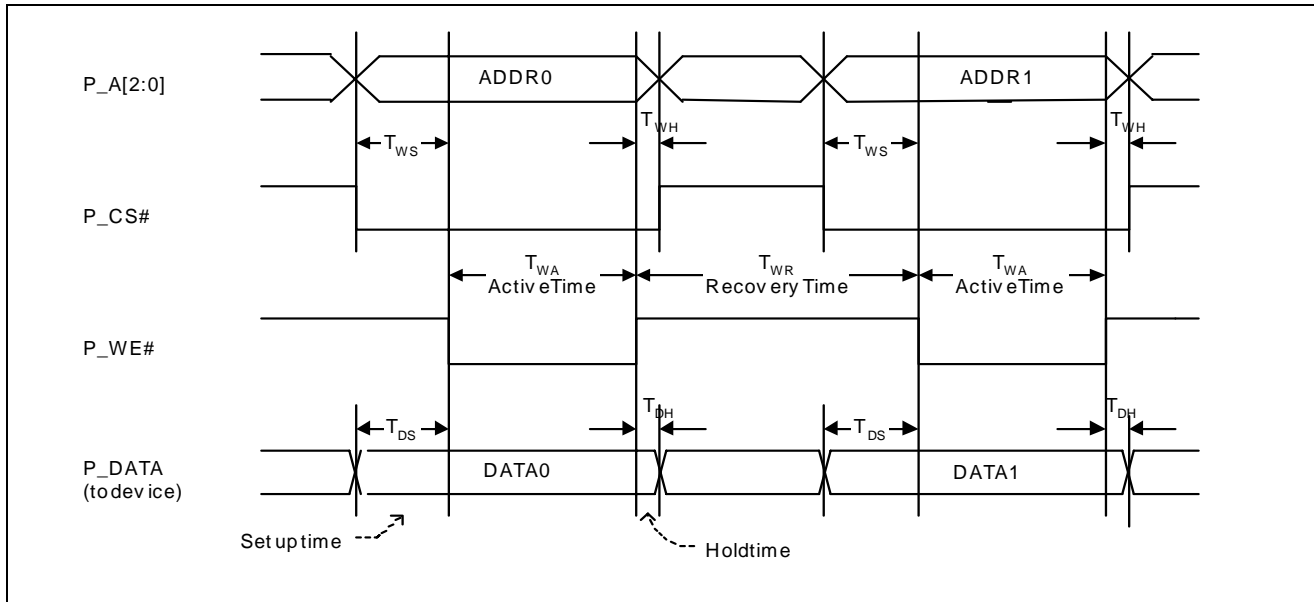


Figure 15 - Typical CPU Timing Diagram for a CPU Write Cycle

Description		(SCLK=100 Mhz)		(SCLK=125 Mhz)		Refer to Figure 17
Write Cycle	Symbol	Min.	Max.	Min.	Max.	
Write Set up Time	$T_{WS}$	10		10		P_A and P_CS# to falling edge of P_WE#
Write Active Time	$T_{WA}$	20		16		At least 2 SCLK cycles
Write Hold Time	$T_{WH}$	2		2		P_A and P_CS# to rising edge of P_WE#
Write Recovery time	$T_{WR}$	30		24		At least 3 SCLK cycles
Data Set Up time	$T_{DS}$	10		10		P_DATA to falling edge of P_WE#
Data Hold time	$T_{DH}$	2		2		P_DATA to rising edge of P_WE#

12.5.3 Typical CPU Timing Diagram for a CPU Read Cycle

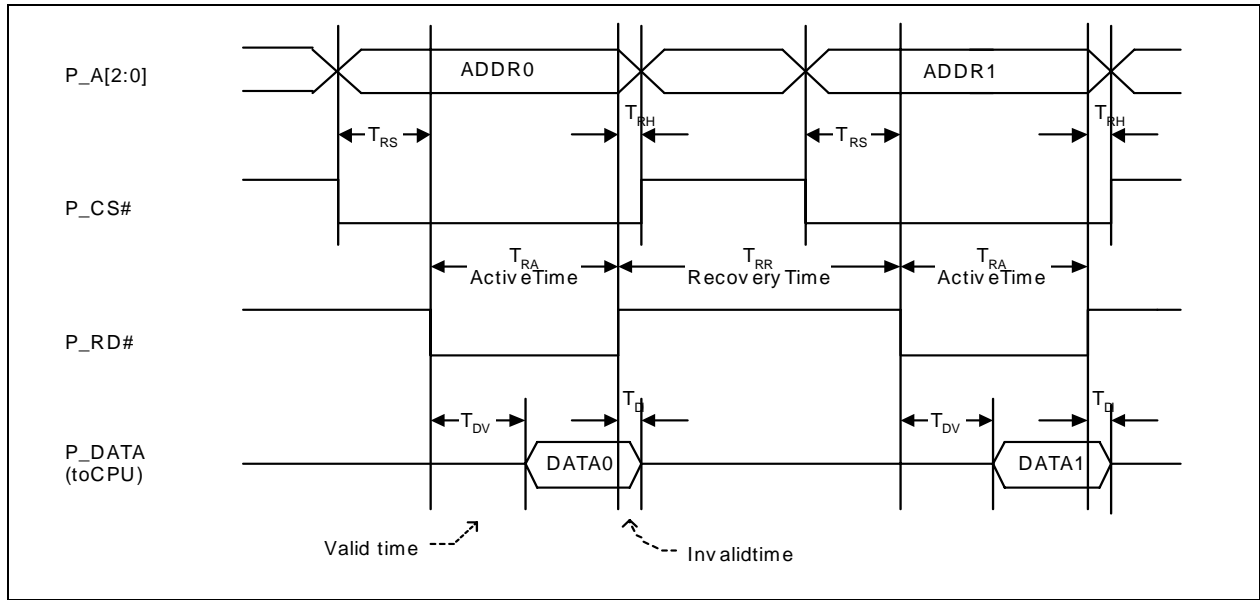


Figure 16 - Typical CPU Timing Diagram for a CPU Read Cycle

Description		(SCLK=100 Mhz)		(SCLK=125 Mhz)		Refer to Figure 18
Read Cycle	Symbol	Min.	Max.	Min.	Max.	
Read Set up Time	$T_{RS}$	10		10		P_A and P_CS# to falling edge of P_RD#
Read Active Time	$T_{RA}$	20		16		At least 2 SCLK cycles
Read Hold Time	$T_{RH}$	2		2		P_A and P_CS# to rising edge of P_RD#
Read Recovery time	$T_{RR}$	30		24		At least 3 SCLK cycles
Data Valid time	$T_{DV}$		10		10	P_DATA to falling edge of P_RD#
Data Invalid time	$T_{DI}$		6		6	P_DATA to rising edge of P_RD#

12.5.4 Local Frame Buffer SBRAM Memory Interface

12.5.4.1 Local SBRAM Memory Interface A

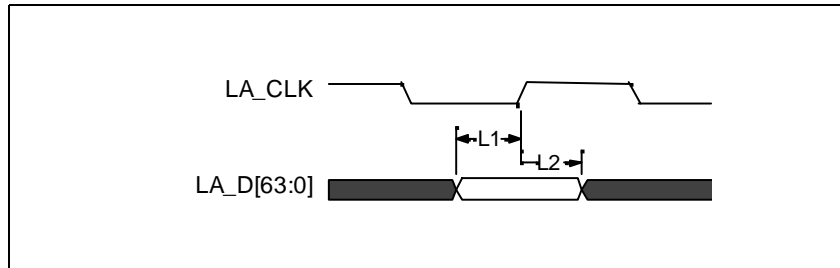


Figure 17 - Local Memory Interface – Input Setup and Hold Timing

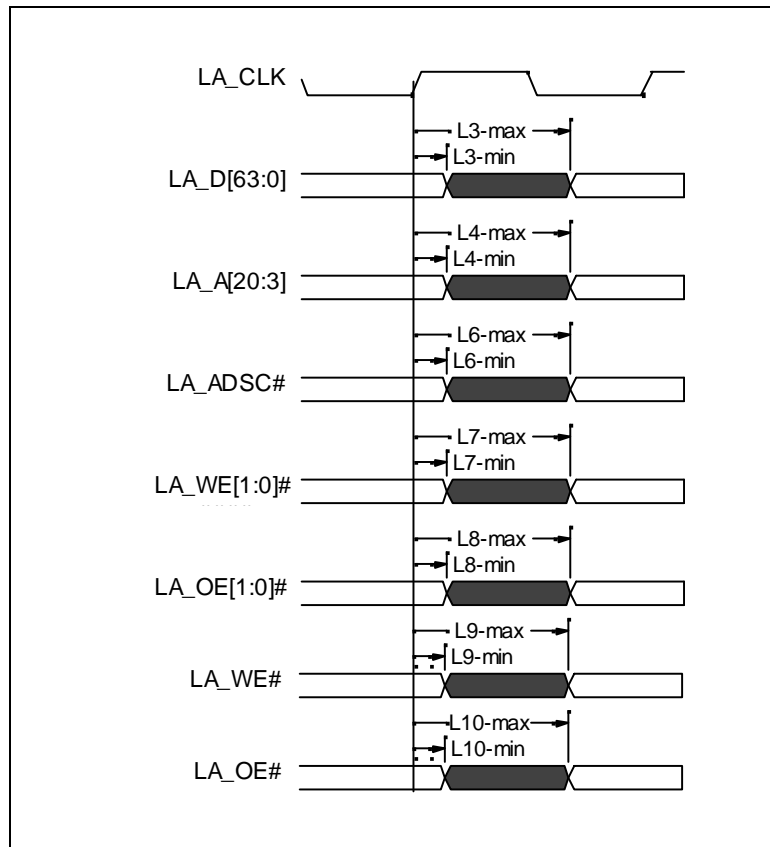


Figure 18 - Local Memory Interface - Output Valid Delay Timing

Symbol	Parameter	-100 MHz		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	$C_L = 25$ pf
L4	LA_A[20:3] output valid delay	2	7	$C_L = 30$ pf
L6	LA_ADSC# output valid delay	1	7	$C_L = 30$ pf
L7	LA_WE[1:0]#output valid delay	1	7	$C_L = 25$ pf
L8	LA_OE[1:0]# output valid delay	-1	1	$C_L = 25$ pf
L9	LA_WE# output valid delay	1	7	$C_L = 25$ pf
L10	LA_OE# output valid delay	1	5	$C_L = 25$ pf

**Table 15 - AC Characteristics – Local Frame Buffer SBRAM Memory Interface**

12.5.5 Reduced Media Independent Interface

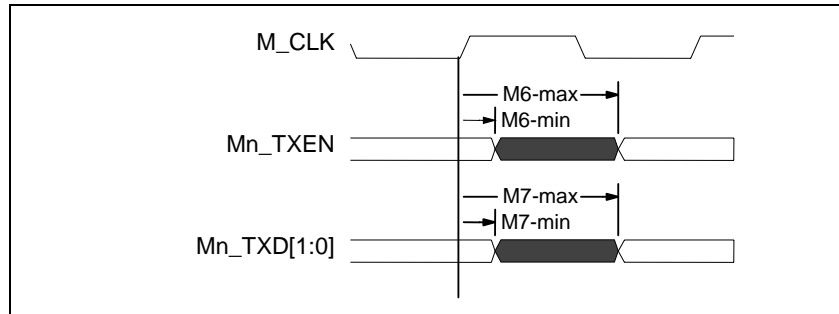


Figure 19 - AC Characteristics – Reduced Media Independent Interface

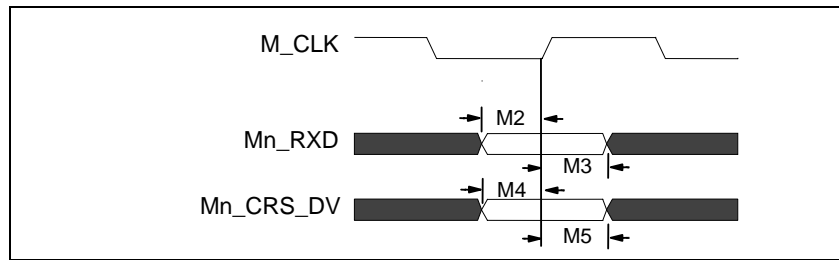


Figure 20 - AC Characteristics – Reduced Media Independent Interface

Symbol	Parameter	M_CLK=50 MHz		Note
		Min. (ns)	Max. (ns)	
M2	Mn_RXD[1:0] Input Setup Time	4		
M3	Mn_RXD[1:0] Input Hold Time	1		
M4	Mn_CRS_DV Input Setup Time	4		
M5	Mn_CRS_DV Input Hold Time	1		
M6	Mn_TXEN Output Delay Time	2	11	$C_L = 20 \text{ pF}$
M7	Mn_TXD[1:0] Output Delay Time	2	11	$C_L = 20 \text{ pF}$

Table 16 - AC Characteristics – Reduced Media Independent Interface

12.5.6 LED Interface

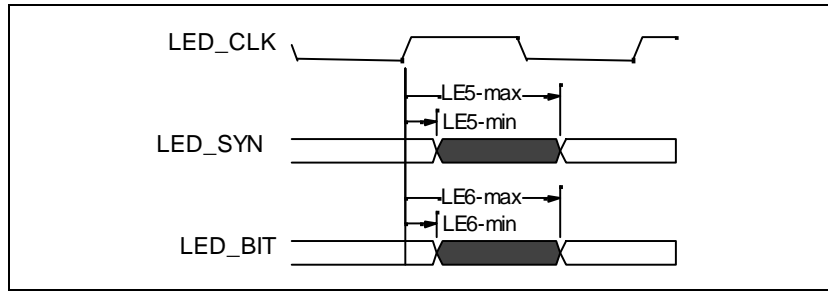


Figure 21 - AC Characteristics – LED Interface

Symbol	Parameter	Variable FREQ.		Note
		Min. (ns)	Max. (ns)	
LE5	LED_SYN Output Valid Delay	-1	7	$C_L = 30 \text{ pf}$
LE6	LED_BIT Output Valid Delay	-1	7	$C_L = 30 \text{ pf}$

Table 17 - AC Characteristics – LED Interface

12.5.7 SCANLINK, SCANCOL Interface

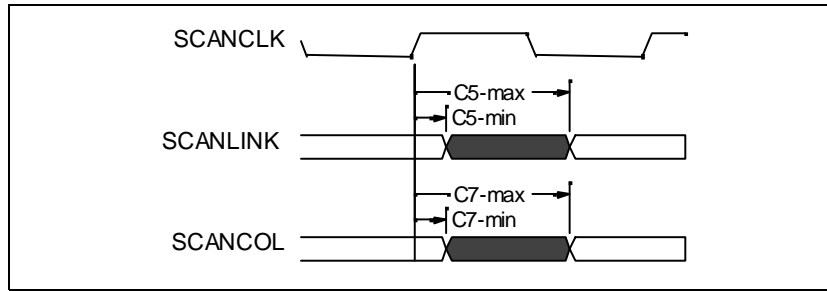


Figure 22 - SCANLINK, SCANCOL Output Delay Timing

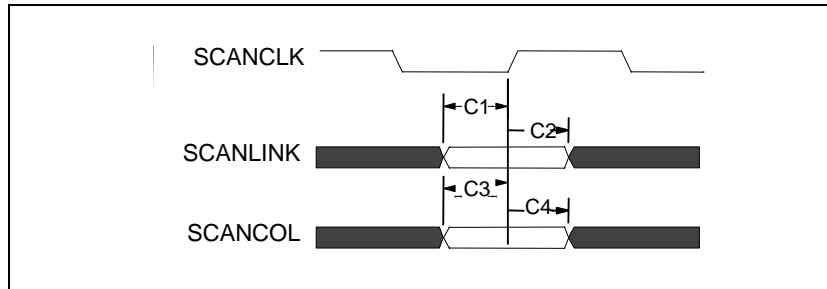


Figure 23 - SCANLINK, SCANCOL Setup Timing

Symbol	Parameter	-25 MHz		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 18 - SCANLINK, SCANCOL Timing

12.6 MDIO Interface

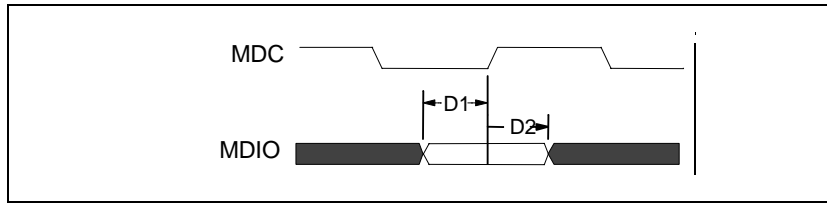


Figure 24 - MDIO Input Setup and Hold Timing

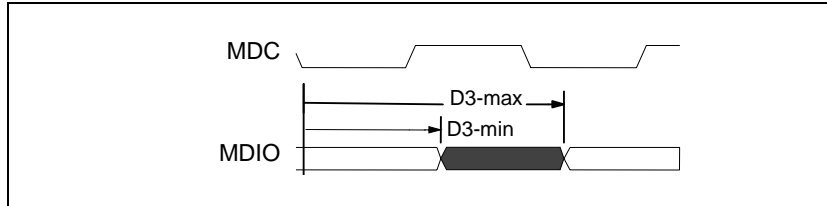


Figure 25 - MDIO Output Delay Timing

Symbol	Parameter	1 MHz		Note:
		Min. (ns)	Max. (ns)	
D1	MDIO input setup time	10		
D2	MDIO input hold time	2		
D3	MDIO output delay time	1	20	$C_L = 50$ pf

Table 19 - MDIO Timing

12.6.1 I<sup>2</sup>C Interface

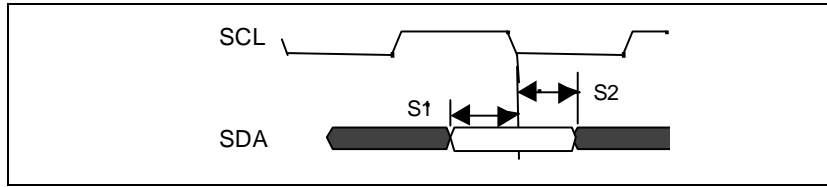


Figure 26 - I<sup>2</sup>C Input Setup Timing

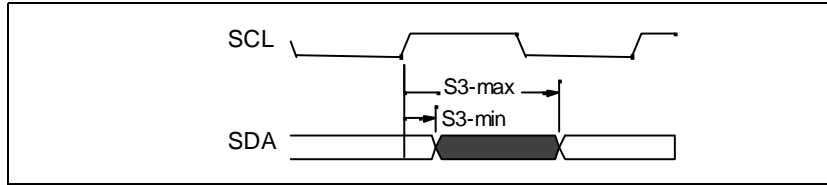


Figure 27 - I<sup>2</sup>C Output Delay Timing

Symbol	Parameter	50 KHz		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> = 30 pf

\* Open Drain Output. Low to High transistor is controlled by external pullup resistor.

Table 20 - I<sup>2</sup>C Timing

12.6.2 Synchronous Serial Interface

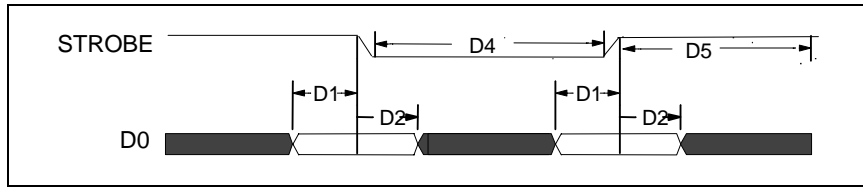


Figure 28 - Serial Interface Setup Timing

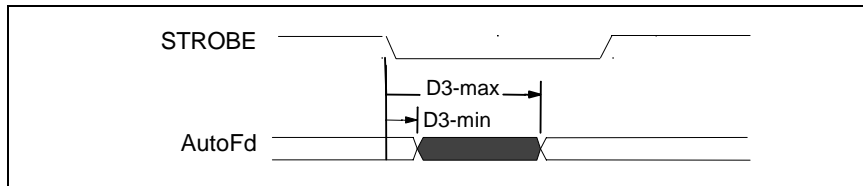
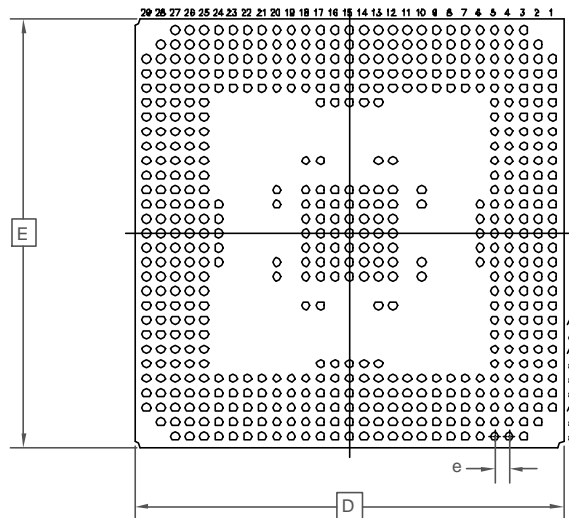
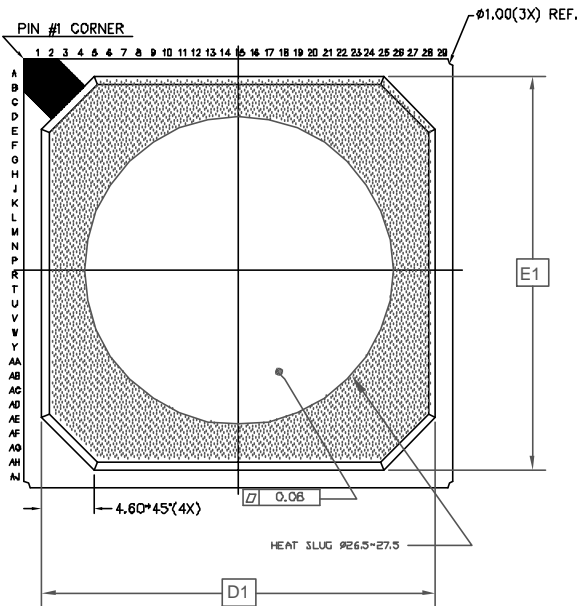


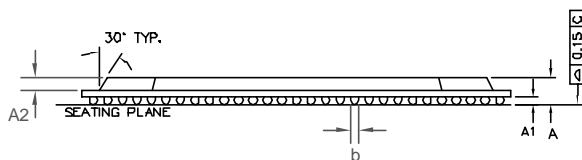
Figure 29 - Serial Interface Output Delay Timing

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	$C_L = 100$ pf
D4	Strobe low time	5 $\mu$ s		
D5	Strobe high time	5 $\mu$ s		

Table 21 - 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.

ISSUE	1			
ACN	213932			
DATE	20Jan03			
APPRD.				



Previous package codes:

BH / G

Package Code GK

Package Outline for 553 Ball HSBGA (37.5x37.5x2.33mm)

GPD00818



**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. 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 and 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 Zarlink Semiconductor Inc. All Rights Reserved.

**TECHNICAL DOCUMENTATION - NOT FOR RESALE**

---