ZL50411
Managed 9FE Layer-2 Ethernet Switch
Data Sheet

Zarlink Features

- Integrated Single-Chip 10/100 Mbps Ethernet Switch
  - Eight 10/100 Mbps auto-negotiating Fast Ethernet (FE) ports with RMII, MII, GPSI, Reverse MII & Reverse GPSI interface options
  - One 10/100 Mbps auto-negotiating port with MII interface option, that can be used as a WAN uplink or as a 9th port
  - a 10/100 Mbps Fast Ethernet (FE) CPU port with Reverse MII interface option
- Embedded 2.0 Mbits (256 KBytes) internal memory for control databases and frame data buffer
  - Supports jumbo frames up to 4 KBytes
- CPU access supports the following interface options:
  - 8/16-bit ISA interface
  - Serial interface with MII port; recommended for light management
  - Serial interface in lightly managed mode, or in unmanaged mode with optional I^2C EEPROM interface
- Ethernet IEEE 802.3x flow control for full duplex ports, back pressure flow control for half duplex ports


Ordering Information

ZL50411GDG   208-Ball LBGA
ZL50411GDG2   208-Ball LBGA**

**Pb Free Tin/Silver/Copper
-40°C to +85°C

- Built-in reset logic triggered by system malfunction
- Built-In Self Test for internal SRAM
- IEEE-1149.1 (JTAG) test port

L2 Switching

- L2 switching
  - MAC address self learning, up to 4 K MAC addresses
  - MAC address table supports unicast and multicast MAC address and IP multicast address learning
- Supports IP Multicast with IGMP snooping, up to 4 K IP Multicast groups
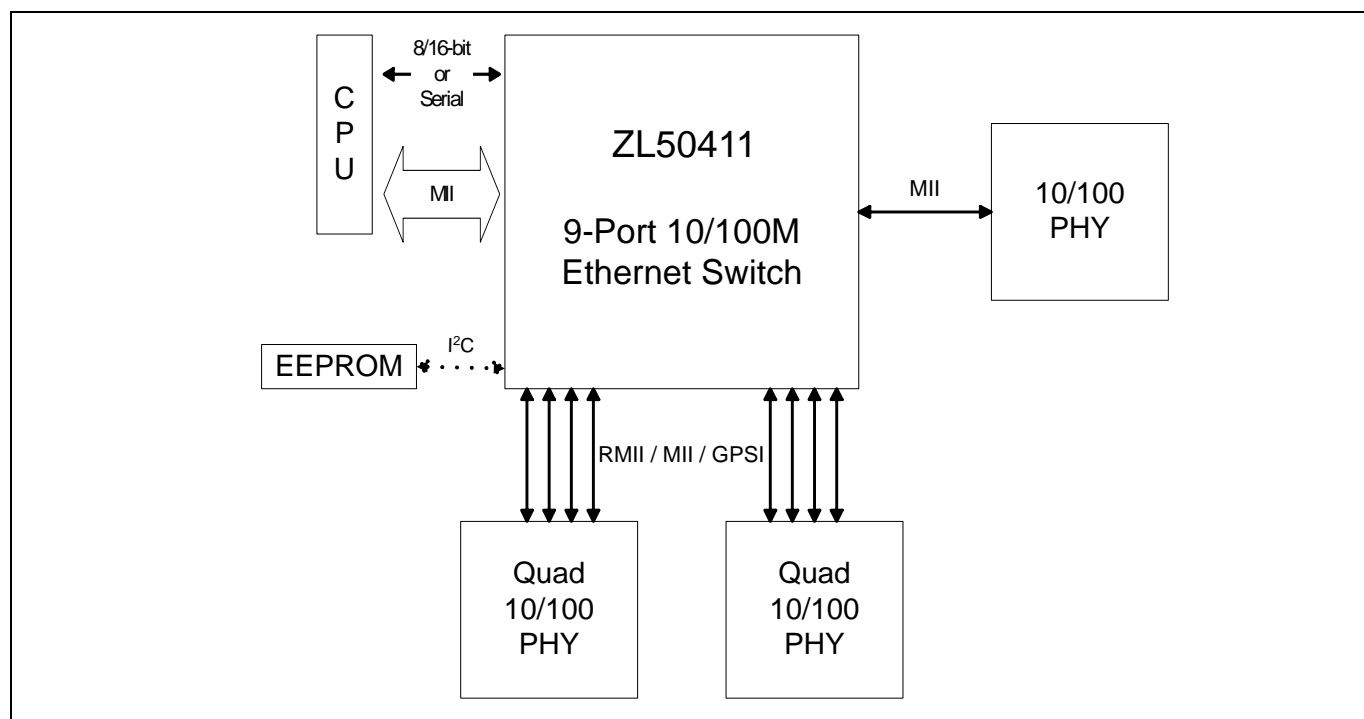
Figure 1 - System Block Diagram

1

- Supports the following spanning standards
  - IEEE 802.1D spanning tree
  - IEEE 802.1w rapid spanning tree
- Supports Ethernet multicasting and broadcasting and flooding control

VLAN Support Features

- Supports the following VLAN standards
  - port-based VLAN
  - IEEE 802.1Q tag-based VLAN, up to 4 K VLANs
- Supports both shared VLAN learning (SVL) and independent VLAN learning (IVL) of MAC addresses
- Limited support for VLAN stacking ("Q-in-Q")
- Supports Private VLAN Edge (Protected Ports)

Classification and Security

- Search engine classification
  - Classifies packets based on single field
    - Source and destination L4 logical ports, or
    - TOS/DS, or
    - VLAN (IEEE 802.1p), or
    - Physical port
  - Assigns a transmission priority and drop precedence
- Packet filtering and security
  - Static address filtering for source and/or destination MAC addresses
  - Static MAC address not subject to aging
  - Secure mode freezes MAC address learning (each port may independently use this mode)
  - IEEE 802.1x access control
  - Port-based priority: VLAN priority in a tagged frame can be overwritten by the priority of port VLAN ID

Traffic Management

- Two (2) transmission classes for FE ports and four (4) transmission classes for uplink port
- Scheduling using weighted fair queuing (WFQ) or strict priority (SP) discipline
- At egress, per-queue weighted random early discard (WRED) with 2 drop precedence levels
  - Configurable WRED thresholds
- For FE ports, supports ingress and egress rate control
  - Bandwidth rationing, Bandwidth on demand, SLA (Service Level Agreement)
  - Granularity of rate regulation to 16 Kbps
  - Ingress rate regulated using WRED, with 2 drop precedence levels, or flow control
- Output traffic regulation per class available on uplink port
- Fully supports Differentiated Services' Expedited and Assured Forwarding (EF and AF) per-hop behaviours
- Intelligent buffer management
  - Achieves high buffer utilization while ensuring fairness among traffic classes and ports
  - Buffer reservations per class and per source port
- Failover Features
  - Rapid link failure detection using hardware-generated heartbeat packets
  - link failover in less than 50 ms
- Supports concentration mode

Zarlink Semiconductor Inc.

- Supports IEEE 802.3ad link aggregation
  - Eight (8) groups of up to 8 ports per group
  - Supports load sharing among trunk ports based on:
    - Source port
    - Source and/or destination MAC address
- Traffic Mirroring
  - Physical port based (RMII enabled ports only)
  - Source or destination MAC address based
  - MAC address pair based
- Supports module hot swap on all ports

Network Management Support

- Built-in RMON MIB counters

Description

The ZL50411 is a low density, low cost, high performance, non-blocking Ethernet switch chip. A single chip provides 8 ports at 10/100 Mbps, 1 uplink port at 10/100 Mbps, and a CPU interface for managed, lightly managed and unmanaged switch applications. The chip supports up to 4 K MAC addresses and up to 4 K tagged-based Virtual LANs (VLANs). It also supports the Private VLAN Edge feature, allowing the ability to set up protected ports in a tagged-based VLAN system.

With strict priority and/or WFQ transmission scheduling and WRED dropping schemes, the ZL50411 provides powerful QoS functions for various multimedia and mission-critical applications. The chip provides 2 transmission priorities (4 priorities for uplink port) 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 ZL50411 recognizes a total of 16 UDP/TCP logical ports, 8 hard-wired and 8 programmable (including one programmable range).

The ZL50411 provides the ability to monitor a link, detect a simple link failure, and provide notification of the failure to the CPU. The CPU can then failover that link to an alternate link.

The ZL50411 supports up to 8 groups of port trunking/load sharing. Each group can contain up to 8 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 ZL50411 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 ZL50411 is fabricated using 0.18 micron technology. The ZL50411 is packaged in a 208-pin Ball Grid Array package.

Zarlink Semiconductor Inc.

Changes Summary

July 2003

- Initial Release

November 2003

- Clarified IP Multicast support is up to 4K groups, as it wasn't mentioned in the data sheets
- Updated Ball Signal Description Table (1.3, "Ball Signal Descriptions" on page 14):
  - clarified the ball signal I/O description for Mn_TXCLK & Mn_RXCLK showing these signals are either inputs *OR* outputs
  - clarified that M9_MTXCLK is an input only
- Updated 1.4, "Signal Mapping and Internal pull-up/Down Configuration" on page 19 to indicate operation of the internal pull-up/down resistors in different modes
- Clarified 10.1.3, "MMAC Reference Clock (REF_CLK) Speed Requirement" on page 53 on usage of REF_CLK
- Clarified PVMODE register bit description for bits [2] & [5]
- Updated ECR4Pn register description as port 9 (uplink) operates differently than the RMAC ports for MII bi-directional clocking (bits [1:0])
- I^{2}C address mapping was corrected for QOSCn registers
- Added Maximum Junction Temperature to 13.1, "Absolute Maximum Ratings" on page 128
- Updated I/O voltage levels to use TTL spec values rather than % of Vcc (13.2, "DC Electrical Characteristics" on page 128)

February 2004

- Added the following to the Feature List:
  - 4 K jumbo frames
  - IEEE 802.3ad support
  - Reverse MII/GPSI
- Added section on PHY addresses (2.2.4, "PHY Addresses" on page 27)
  - Clarified that they are hard-coded
- Fixed error in DS on sending Ethernet Frames via 8/16-bit or serial interface.
  - The Status Bytes is sent before the frame, for both Tx and Rx
- Added more cross-references to available AppNotes
- Added section on Stacked VLAN (Q-in-Q) (5.9.3, "VLAN Stacking (Q-in-Q)" on page 42) and IP Multicast Switching (5.10, "IP Multicast Switching" on page 43) since they weren't really discussed in the DS
- Added more clock descriptions to 10.0, "Clocks" on page 52
- INT_MASK and INTP_MASK registers should state that the default register value is 0x00

August 2004

- Added section Changes Summary to document
- Added section on SCL clock generation (10.2.2, "SCL" on page 53)
- Interrupt Register was incorrectly identified as read only, should be read/write
  - Clarified that only bit [7] is not self-clearing
- Updated CPU timing diagrams to clarify timing (13.4, "AC Characteristics and Timing" on page 130)

November 2004

- Added section 1.6, "Default Switch Configuration and Initialization Sequence" on page 23
- Added Private VLAN Edge (protected ports), force VLAN tag out, and unknown IP Multicast filtering support
- Updated CPU timing diagrams to clarify P_A timing (13.4, "AC Characteristics and Timing" on page 130)

Zarlink Semiconductor Inc.

January 2005

- Removed reference to direct register INDEX_REG1 (address 0x1) from SSI diagrams, as not applicable

June 2005

- Corrected ordering code to ZL50411GD"**G**"
- Clarified that port mirroring is only available if the source & destination ports are in RMII mode
- Updated PVMODE bit [5] to reflect the proper MAC address range: 01-80-C2-00-00-00~F
- Clarified DATAOUT output can be open-drain or totem-pole based on debounce selection via bootstrap TSTOUT[0]
- Added power sequencing recommendation (1.7, "Power Sequencing" on page 25)
- Added Reverse MII/GPSI timing characteristics (13.4.10, "Reverse General Purpose Serial Interface (RvGPSI)" on page 139 and 13.4.11, "MII Management Data Interface (MDIO/MDC)" on page 140)
- Clarified that counter "DelayExceededDiscards" is not applicable for the ZL50411 (11.0, "Hardware Statistics Counters" on page 53)

December 2005

- Clarified that TRST signal should be externally tied to GND via weak resistor, as per JTAG standard (1.3, "Ball Signal Descriptions" on page 14)
- Added more text to section 2.8, "JTAG" on page 28
- Clarified counter definitions (11.0, "Hardware Statistics Counters" on page 53)
- Added more explaination to VLAN ID Hashing feature: register FEN, bit [3]
- Removed definition for SE_OPMODE bit[5] (ARP report control), as this feature was not implemented and this bit was mistakenly left in the register definition.

April 2006

- Added Pb-free order code (ZL50411GDG2)
- Added section on multicast MAC address learning/switching (5.11, "L2 Multicast Switching" on page 44) since it wasn't really discussed in the DS
- Clarified registers UCC, MCC & MCCTH
- Renamed register UCC (now PCC), as name was misleading
- Updated timing to CPU RvMII, as min. output delay should have been 0ns

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

Table of Contents

Zarlink Semiconductor Inc.

1.0 BGA and Ball Signal Descriptions

1.1 BGA Views (Top-View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
A	SCLK	P_CS#	P_RD#	P_WE#	P_DATA1	P_DATA3	P_DATA5	P_DATA7	P_DATA9	P_DATA11	P_DATA13	P_DATA15	REF_CLK	RSVD	M9_MTXCLK	M9_TXEN	A
B	P_INT	P_A0	P_A1	P_A2	P_DATA0	P_DATA2	P_DATA4	P_DATA6	P_DATA8	P_DATA10	P_DATA12	P_DATA14	TCK	TMS	RSVD	M9_RXCK	B
C	RESETOUT#	TSTOUT1	TSTOUT3	TSTOUT5	TSTOUT6	TSTOUT7	TSTOUT9	TSTOUT11	TSTOUT12	TSTOUT14	TSTOUT15	TRST#	TDI	RSVD	M9_CRS	M9_COL	C
D	RESIN#	TSTOUT0	TSTOUT2	TSTOUT4	3.3V	IC_GND	TSTOUT8	TSTOUT10	1.8V	TSTOUT13	TDO	3.3V	RSVD	RSVD	M9_RXDV	RSVD	D
E	M2_COL	M0_COL	M1_COL	3.3V									3.3V	RSVD	RSVD	RSVD	E
F	M_MDC	M_MDIO	M0_RXD2	M0_RXD3									M9_RXD2	M9_RXD3	RSVD	RSVD	F
G	M0_RXD0	M0_RXD1	M0_RXCK	M0_TXD3			GND	GND	GND	GND			M9_RXD0	M9_RXD1	M9_TXD2	M9_TXD3	G
H	M0_CRS	M0_TXEN	M0_TXD2	1.8V			GND	GND	GND	GND			1.8V	M7_COL	M9_TXD0	M9_TXD1	H
J	M0_TXD0	M0_TXD1	M0_TXCK	M1_RXD3			GND	GND	GND	GND			M7_TXD3	M7_TXCK	M7_TXD1	M7_TXD0	J
K	M1_RXD0	M1_RXD1	M1_RXD2	M1_RXCK			GND	GND	GND	GND			M7_TXD2	M7_RXD2	M7_TXEN	M7_CRS	K
L	M1_CRS	M1_TXEN	M1_TXD2	M1_TXD3									M7_RXD3	M7_RXCK	M7_RXD1	M7_RXD0	L
M	M1_TXD0	M1_TXD1	M1_TXCK	3.3V									3.3V	M6_COL	M5_COL	M4_COL	M
N	M2_RXD3	M2_TXCK	M2_TXD3	M3_RXD3	3.3V	M3_TXD3	1.8V	M4_RXD3	M4_TXCK	M4_TXD3	M5_RXD3	M5_TXCK	M5_TXD3	M6_RXD3	M6_TXCK	M6_TXD3	N
P	M2_RXD2	M2_RXCK	M2_TXD2	M3_RXD2	M3_RXCK	M3_TXD2	M3_COL	M4_RXD2	M4_RXCK	M4_TXD2	M5_RXD2	M5_RXCK	M5_TXD2	M6_RXD2	M6_RXCK	M6_TXD2	P
R	M2_RXD1	M2_TXEN	M2_TXD1	M3_RXD1	M3_TXEN	M3_TXD1	M_CLK	M4_RXD1	M4_TXEN	M4_TXD1	M5_RXD1	M5_TXEN	M5_TXD1	M6_RXD1	M6_TXEN	M6_TXD1	R
T	M2_RXD0	M2_CRS	M2_TXD0	M3_RXD0	M3_CRS	M3_TXD0	M3_TXCK	M4_RXD0	M4_CRS	M4_TXD0	M5_RXD0	M5_CRS	M5_TXD0	M6_RXD0	M6_CRS	M6_TXD0	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

1.2 Power and Ground Distribution

G7-10, H7-10, J7-10, K7-10	GND	V_{SS}	Ground
D5, D12, E4, E13, M4, M13, N5	3.3V	V_{CC}	I/O Power
D9, H4, H13, N7	1.8V	V_{DD}	Core Power

13

1.3   Ball Signal Descriptions

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

Notes

# =	Active low signal
Input =	Input signal
Input-ST =	Input signal with Schmitt-Trigger
Output =	Output signal (Tri-State driver)
I/O-TS =	Input & Output signal with Tri-State driver
pull-up =	Weak internal pull-up (nominal 100KΩ)
	(refer to Section 1.4 on page 19 as some internal pull-ups are not enabled in certain configurations)
pull-down =	Weak internal pull-down (nominal 100KΩ)
	(refer to Section 1.4 on page 19 as some internal pull-downs are not enabled in certain configurations)

Ball Signal Description Table

Ball No(s)	Symbol	I/O	Description
16-Bit CPU Bus Interface			
A12, B12, A11, B11, A10, B10, A9, B9, A8, B8, A7, B7, A6, B6, A5, B5	P_DATA[15:0]	I/O-TS with pull-up	Processor Bus Data Bit [15:0]. P_DATA[7:0] is used in 8-bit mode.
B4, B3, B2	P_A[2:0]	Input with pull-up	Processor Bus Address Bit [2:0]
A4	P_WE#	Input with pull-up	CPU Bus-Write Enable
A3	P_RD#	Input	CPU Bus-Read Enable
A2	P_CS#	Input with pull-up	Chip Select
B1	P_INT	Output	CPU Interrupt Active high, but can be changed via register DEVICE, bit [1].
Fast Ethernet Access Ports [7:0] MII			
L13, K14, L15, L16, N14, P14, R14, T14, N11, P11, R11, T11, N8, P8, R8, T8, N4, P4, R4, T4, N1, P1, R1, T1, J4, K3, K2, K1, F4, F3, G2, G1	M[7:0]_RXD[3:0]	Input with pull-up	Ports [7:0] – Receive Data Bit [3:0]

Zarlink Semiconductor Inc.

Ball Signal Description Table (continued)

Ball No(s)	Symbol	I/O	Description
K16, T15, T12, T9, T5, T2, L1, H1	M[7:0]_CRS_DV	Input with pull-up	Ports [7:0] – Carrier Sense and Receive Data Valid
K15, R15, R12, R9, R5, R2, L2, H2	M[7:0]_TXEN	Output, slew	Ports [7:0] – Transmit Enable This pin also serves as a bootstrap pin.
J13, K13, J15, J16, N16, P16, R16, T16, N13, P13, R13, T13, N10, P10, R10, T10, N6, P6, R6, T6, N3, P3, R3, T3, L4, L3, M2, M1, G4, H3, J2, J1	M[7:0]_TXD[3:0]	Output, slew	Ports [7:0] – Transmit Data Bit [3:0]
H14, M14, M15, M16, P7, E1, E3, E2	M[7:0]_COL	Input with pull-down	Ports[7:0] – Collision
J14, N15, N12, N9, T7, N2, M3, J3	M[7:0]_TXCLK	Input or Output with pull-up	Ports[7:0] – Transmit Clock This pin in an output if ECR4Pn[1:0]='11'
L14, P15, P12, P9, P5, P2, K4, G3	M[7:0]_RXCLK	Input or Output with pull-up	Ports[7:0] – Receive Clock This pin in an output if ECR4Pn[1:0]='11'
Fast Ethernet Uplink Port MII			
G16, G15, H16, H15	M9_TXD[3:0]	Output	Transmit Data Bit [3:0]
D15	M9_RXDV	Input with pull-up	Receive Data Valid
C15	M9_CRS	Input with pull-down	Carrier Sense
C16	M9_COL	Input with pull-down	Collision Detected
B16	M9_RXCLK	Input or Output with pull-up	Receive Clock This pin in an output if ECR4P9[1:0]='11'
F14, F13, G14, G13	M9_RXD[3:0]	Input with pull-up	Receive Data Bit [3:0]
A16	M9_TXEN	Output with pull-up	Transmit Data Enable
A15	M9_MTXCLK	Input with pull-up	Transmit Clock

Zarlink Semiconductor Inc.

Ball Signal Description Table (continued)

Ball No(s)	Symbol	I/O	Description
Test Interface			
C11, C10, D10, C9, C8, D8, C7, D7, C6, C5, C4, D4, C3, D3, C2, D2	TSTOUT[15:0]	Output	[15:4] Reserved [3] EEPROM checksum is good [2] Initialization Completed [1] Memory Self Test in progress [0] Initialization started These pins also serve as bootstrap pins.
Test Facility			
C13	TDI	Input with pull-up	JTAG - Test Data In
C12	TRST#	Input with pull-up	JTAG - Test Reset In normal operation, this pin should be pulled low. Recommend weak external pull-down resistor (470 Ω to 1 kΩ).
B13	TCK	Input with pull-up	JTAG - Test Clock
B14	TMS	Input with pull-up	JTAG - Test Mode State
D11	TDO	Output	JTAG - Test Data Out
System Clock, Power, and Ground Pins			
A1	SCLK	Input	System Clock. Based on system requirement, SCLK needs to operate at difference frequency. SCLK requires 40/60% duty cycle clock.
D9, H4, H13, N7,	V_{DD}	Power	+1.8 Volt DC Supply
D5, D12, E4, E13, M4, M13, N5,	V_{CC}	Power	+3.3 Volt DC Supply
G7-10, H7-10, J7-10, K7-10	V_{SS}	Power Ground	Ground
Misc.			
D1	RESIN#	Input	Reset Input
C1	RESETOUT#	Output	Reset PHY
F1	M_MDC	Output	MII Management Data Clock
F2	M_MDIO	I/O-TS with pull-up	MII Management Data I/O
R7	M_CLK	Input	RMAC Reference Clock

Zarlink Semiconductor Inc.

Ball Signal Description Table (continued)

Ball No(s)	Symbol	I/O	Description
A13	REF_CLK	Input with pull-up	MMAC Reference Clock
, C14, D14, D13, E14, E16, E15, F16, F15, D16, B15, A14	RSVD	N/A	Reserved. Leave unconnected.
D6	IC_GND	IC_GND	Internal Connect. Tie to ground (V_{SS}) via 1 K resistor.

Bootstrap Pins[1]

External pull-up/down resistors are required on all bootstrap pins for proper operation. See "Bootstrap Options" on page 22 for more information.

Ball No(s)	Symbol	I/O	Description
D2	TSTOUT[0]	Input (Reset Only)	Enable Debounce on SSI interface Pullup – Enabled Pulldown - Disabled When enabled, DATAOUT is an open-drain output; when disabled, DATAOUT is a totem-pole output. See "Synchronous Serial Interface (SSI)" on page 133 for more details on debounce timing.
C3, D3, C2	TSTOUT[3:1]	Input (Reset Only)	Management interface operation mode: 000 – 16-bit parallel interface 001 – 8-bit parallel interface 010 – Serial with MII as Ethernet frame transfer interface. 011 – Serial only. CPU can transmit/receive frames with the serial interface. 111 – Unmanaged Serial. No CPU packet can be transmit or received with the serial interface. EEPROM can be used to configure the device at bootup. A one (1) indicates pullup. A zero (0) indicates pulldown. TSTOUT[1] is the Least Significant Bit (LSB).
C5, C4, D4	TSTOUT[6:4]	Input (Reset Only)	Device ID. Default address of the device for serial interface. Up to 8 device can be sharing the serial management bus with different device ID. A one (1) indicates pullup. A zero (0) indicates pulldown. TSTOUT[4] is the Least Significant Bit (LSB).

Zarlink Semiconductor Inc.

Ball Signal Description Table (continued)

Ball No(s)	Symbol	I/O	Description
C6	TSTOUT[7]	Input (Reset Only)	EEPROM not installed. Pullup: Not installed Pulldown: Installed
D7	TSTOUT[8]	Input (Reset Only) Must be externally pulled-up	Manufacturing Option. Must be pulled up.
C7	TSTOUT[9]	Input (Reset Only)	Module Detect Pullup: Enable. In this mode, the device will detect the existence of a PHY (for hot swap purpose). Pulldown: Disable
D8	TSTOUT[10]	Input (Reset Only) Must be externally pulled-down	Manufacturing Option. Must be pulled down.
C8	TSTOUT[11]	Input (Reset Only)	Power Saving Pullup: Enable MAC power saving mode Pulldown: Disable MAC power saving mode
C9	TSTOUT[12]	Input (Reset Only)	Timeout Reset Enable Pullup: Enable Pulldown: Disable
C11, C10, D10	TSTOUT[15:13]	Input (Reset Only) Must be externally pulled-up	Manufacturing Options. Must be pulled-up.
K15, R15, R12, R9, R5, R2, L2, H2	M[7:0]_TXEN	Input (Reset Only)	User Defined Bootstrap: Usually used in conjuction with Module Detect to determine what interface to use for the inserted module. Can be read from BOOTSTRAP2 register

1. Note: 1=pull-up; 0=pull-down

Zarlink Semiconductor Inc.

1.4   Signal Mapping and Internal pull-up/Down Configuration

The ZL50411 Fast Ethernet access ports (0-7) support 3 interface options: RMII, MII & GPSI. The table below summarizes the interface signals required for each interface and how they relate back to the Pin Symbol name shown in the "Ball Signal Description Table" on page 14. It also specifies whether the internal pull-up/down resistor is present for each pin in the specific operating mode.

Notes:
I – Input
O – Output
U – Pullup
D - Pulldown

Fast Ethernet Access Ports Pin Symbol	No Module (Bootstrap TSTOUT9='1')	RMII Mode (ECR4Pn[4:3]='11')	MII Mode (ECR4Pn[4:3]='01')	GPSI Mode (ECR4Pn[4:3]='00')
M[7:0]_RXD0	(U)	M[7:0]_RXD0 (I)	M[7:0]_RXD0 (I)	M[7:0]_RXD (I)
M[7:0]_RXD1	(U)	M[7:0]_RXD1 (I)	M[7:0]_RXD1 (I)	NC (U)
M[7:0]_RXD2	(U)	NC (U)	M[7:0]_RXD2 (I)	NC (U)
M[7:0]_RXD3	(U)	NC (U)	M[7:0]_RXD3 (I)	NC (U)
M[7:0]_TXEN	(O)	M[7:0]_TXEN (O)	M[7:0]_TXEN (O)	M[7:0]_TXEN (O)
M[7:0]_CRS_DV	(U)	M[7:0]_CRS_DV (I)	M[7:0]_DV (I)	M[7:0]_CRS (I)
M[7:0]_TXD0	(O)	M[7:0]_TXD0 (O)	M[7:0]_TXD0 (O)	M[7:0]_TXD (O)
M[7:0]_TXD1	(O)	M[7:0]_TXD1 (O)	M[7:0]_TXD1 (O)	NC (O)
M[7:0]_TXD2	(O)	NC (O)	M[7:0]_TXD2 (O)	NC (O)
M[7:0]_TXD3	(O)	NC (O)	M[7:0]_TXD3 (O)	NC (O)
M[7:0]_COL	(D)	NC (D)	M[7:0]_COL (I)	M[7:0]_COL (I)
M[7:0]_TXCLK	(U)	NC (U)	M[7:0]_TXCLK (IO)	M[7:0]_TXCLK (IO)
M[7:0]_RXCLK	(U)	NC (U)	M[7:0]_RXCLK (IO)	M[7:0]_RXCLK (IO)

Table 1 - Signal Mapping In Different Operation Mode

Zarlink Semiconductor Inc.

The ZL50411 Fast Ethernet uplink port (port 9) supports 1 interface option: MII. 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 Description Table" on page 14.

Fast Ethernet Uplink Port Pin Symbol	No Module (Bootstrap TSTOUT9='1')	MII Mode (ECR4P9[4:3]='11')
M9_RXD0	(U)	M9_RXD0 (I)
M9_RXD1	(U)	M9_RXD1 (I)
M9_RXD2	(U)	M9_RXD2 (I)
M9_RXD3	(U)	M9_RXD3 (I)
M9_RXDV	(U)	M9_RXDV (I)
M9_CRS	(D)	M9_CRS (I)
M9_COL	(D)	M9_COL (I)
M9_RXCLK	(U)	M9_RXCLK (IO)
M9_TXD0	(O)	M9_TXD0 (O)
M9_TXD1	(O)	M9_TXD1 (O)
M9_TXD2	(O)	M9_TXD2 (O)
M9_TXD3	(O)	M9_TXD3 (O)
M9_TXEN	(U)	M9_TXEN (O)
REF_CLK	(U)	REF_CLK (I)
M9_MTXCLK	(U)	M9_MTXCLK (I)

Table 2 - Signal Mapping In Different Operation Mode

Zarlink Semiconductor Inc.

The ZL50411 CPU access support 5 interface options: 8 or 16-bit parallel, serial+MII (port 8), serial only, 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 Description Table" on page 14.

Management Interface Pin Symbol	16-bit CPU (TSTOUT[3:1]='000')	8-bit CPU (TSTOUT[3:1]='001')	Serial with MII (TSTOUT[3:1]='010')	Serial Only (TSTOUT[3:1]='011' or '111')
P_A[0]	P_A[0] (I)	P_A[0] (I)	NC (U)	SDA (IOU) (111 only)
P_A[1]	P_A[1] (I)	P_A[1] (I)	NC (U)	SCL (OU) (111 only)
P_A[2]	P_A[2] (I)	P_A[2] (I)	NC (U)	NC (U)
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)	P_INT (O)	P_INT (O)
P_DATA0	P_DATA0 (IOU)	P_DATA0 (IOU)	CPU_MII_TXD0 (O)	NC (U)
P_DATA1	P_DATA1 (IOU)	P_DATA1 (IOU)	CPU_MII_TXD1 (O)	NC (U)
P_DATA2	P_DATA2 (IOU)	P_DATA2 (IOU)	CPU_MII_TXD2 (O)	NC (U)
P_DATA3	P_DATA3 (IOU)	P_DATA3 (IOU)	CPU_MII_TXD3 (O)	NC (U)
P_DATA4	P_DATA4 (IOU)	P_DATA4 (IOU)	CPU_MII_TXCLK (O)	NC (U)
P_DATA5	P_DATA5 (IOU)	P_DATA5 (IOU)	CPU_MII_TXEN (O)	NC (U)
P_DATA6	P_DATA6 (IOU)	P_DATA6 (IOU)	NC (U)	NC (U)
P_DATA7	P_DATA7 (IOU)	P_DATA7 (IOU)	NC (U)	NC (U)
P_DATA8	P_DATA8 (IOU)	NC (U)	CPU_MII_RXD0 (I)	NC (U)
P_DATA9	P_DATA9 (IOU)	NC (U)	CPU_MII_RXD1 (I)	NC (U)
P_DATA10	P_DATA10 (IOU)	NC (U)	CPU_MII_RXD2 (I)	NC (U)
P_DATA11	P_DATA11 (IOU)	NC (U)	CPU_MII_RXD3 (I)	NC (U)
P_DATA12	P_DATA12 (IOU)	NC (U)	CPU_MII_RXCLK (O)	NC (U)
P_DATA13	P_DATA13 (IOU)	NC (U)	CPU_MII_RXDV (I)	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 3 - Signal Mapping In Different Operation Mode

Zarlink Semiconductor Inc.

1.5   Bootstrap Options

TSTOUT[15:0] and M[7:0]_TXEN pins serve as bootstrap pins during device power-up or reset. Please refer to "Typical Reset & Bootstrap Timing Diagram" on page 130 for more information on when the bootstrap pins are sampled. The bootstrap pins require external pull-up/down resistors for proper operation. Recommend 10 K for pull-ups and 1 K for pull-downs.

The table below summarizes the bootstrap options.

Feature	Description
CPU Interface	The ZL50411 allows the selection of 5 different management interfaces: 8/16-bit parallel, serial with MII, serial only and unmanaged serial with I^2C EEPROM. TSTOUT[3:1] is used to select the interface options mentioned above. If the serial interface is selected, addition bootstrap options are required: • TSTOUT[0] enables or disables the DEBOUNCE feature (refer to "Synchronous Serial Interface" on page 35) • TSTOUT[6:4] selects the device ID Also, in unmanaged mode, an optional I^2C EEPROM can be used to configure the device at power-up or reset. TSTOUT[7] selects the EEPROM option.
Ethernet Interface	The ZL50411 supports module hotswap on all it's ports. This is enabled via TSTOUT[9]. When enabled, bootstrap pins M[7:0]_TXEN (ports 0-7) are used to specify the module type to support multiple ethernet interfaces during module hotswap. Another feature is the MAC power savings mode. When enabled via TSTOUT[11], each port's MAC will detect inactivity on the port and go into a power savings state. When activity is detected once again on the port, the MAC will come out of this state.
Misc. Features	One other feature selected via bootstrap is Timeout Reset Enable (TSTOUT[12]). This enables a monitoring block with the device which will detect if any hardware state machine is in a non-idle state for more than 5 seconds. Refer to section 2.7 for more details on this feature.

Table 4 - Bootstrap Features

1.5.1   Recommended Default Bootstrap Settings

The following are the recommended default settings for the bootstrap options:

- Unmanaged/Lightly Managed
  - Reserved/Manufacturing bootstraps
    - TSTOUT[15:13,8] must be pulled-up
    - TSTOUT[10] must be pulled-down
  - CPU Interface:
    - Strobe debounce bootstrap, TSTOUT[0], should be normally pulled-up, unless you wish to disable the debounce logic
    - CPU Interface bootstrap, TSTOUT[3:1], should be pulled-up to indicate unmanaged SSI CPU interface. To enable SSI-only lightly managed mode, pulled-down TSTOUT[3]. To enable SSI+MII lightly managed mode, pulled-down TSTOUT[3,1]

Zarlink Semiconductor Inc.

- SSI Device ID bootstrap, TSTOUT[6:4], should be pulled-down to indicate device ID 0x0 for the SSI interface. Can be changed to whatever device ID required if more than one device on the SSI bus.
    - EEPROM bootstrap, TSTOUT[7], should be pulled-up to disable until the system is debugged. You can pull-down this bootstrap if using the optional EEPROM in unmanaged mode (NOTE: this bootstrap is not valid in any other CPU mode)
  - Module Detect bootstrap, TSTOUT[9], should be pulled-down
    - In lightly managed mode, you can enable the optional Module Detect feature
    - If enabled, need to use Mn_TXEN to indicate module type
  - Power Saving bootstrap, TSTOUT[11], should be normally pulled-up
  - Timeout Reset bootstrap, TSTOUT[12], should be pull-down
   - Once system is debugged, you can enable the optional feature with pull-up (Refer to section 2.7 for more details on this feature)
- Managed
  - Reserved/Manufacturing bootstraps
    - TSTOUT[15:13,6:4,8,7,0] must be pulled-up
    - TSTOUT[10] must be pulled-down
  - CPU Interface
    - TSTOUT[3:1], should be pulled-down to indicate 16-bit CPU mode
    - For 8-bit CPU mode, pull-up TSTOUT[1]
  - Module Detect bootstrap, TSTOUT[9], should be pulled-down, unless using the Module Detect feature
    - If enabled, need to use Mn_TXEN to indicate module type
  - Power Saving bootstrap, TSTOUT[11], should be normally pulled-up
  - Timeout Reset bootstrap, TSTOUT[12], should be pull-down
    - Once system is debugged, you can enable the optional feature with pull-up (Refer to section 2.7 for more details on this feature)

1.6   Default Switch Configuration and Initialization Sequence

The ZL50411 will come out of reset in a default configuration, which will allow for basic L2 switching and automatic MAC address learning.

In unmanaged mode, the default configuration will take effect immediately after reset. The default settings can be changed using the optional EEPROM.

- System Defaults
  - Port-based VLAN
  - MAC address 00-00-00-00-00-00 not learned
  - Drop MAC addresses 01-80-C2-00-00-01~F
  - No IP multicast switching support
  - Trunking and mirroring disabled
  - MAC address agetime is 300 seconds
  - VLAN 802.1p prioritization
    - All priority bits mapped to priority 0 (lowest)
  - 96 queued unicast/multicast frames will trigger flow control
  - All WRED drop percentages equal to 0%
  - Unicast/multicast/broadcast flood control disabled
  - No shared or per-class buffer pools
- Per-port Defaults
  - Disable per-port fixed priority and drop precedence
  - Disable asynchronous flow control
  - Spanning Tree per-port state equal to forwarding
  - Don't filter tagged/untagged VLAN frames

Zarlink Semiconductor Inc.

- Automatic learning enabled
- Per-port security disabled
- Support frame size 64 <= n <= 1522
- Pad transmit frames < 64B
- Standard preamble
- Strict Priority scheduling
- FE Ports
    - RMII mode
    - Auto-negotiate 10/100 M, Full Duplex, Flow Control
    - Rate control disabled
    - per-source port buffer pool of 96 buffers, with flow control threshold of 48 buffers
- Uplink Port
    - MII mode
    - Auto-negotiate 10/100 M, Full Duplex, Flow Control
    - per-source port buffer pool of 384 buffers, with flow control threshold of 192 buffers

In lightly managed/managed mode, the default configuration can be used as well, however, the device needs to be told when to start switching. This is done via the "Init Complete" bit, set in GCR[4]. The default settings can be overridden using the CPU interface, but should be done before setting of GCR[4]. One thing to note is after reset, the device will start to initialize the control tables. Therefore, a short delay (100 us~1 ms) is necessary before changing the register settings and/or control tables, and before setting GCR[4].

- System Defaults
    - CPU MAC address is 00-00-00-00-00-00
    - Forward MAC addresses 01-80-C2-00-00-00~F to CPU port
        - Except 01-80-C2-00-00-01~F, which are dropped
    - All interrupts enabled
    - MAC address learn report to CPU disabled
    - Statistics counters disabled
    - DiffServ EF code support disabled
    - No VLAN ID hashing
- Per-port Defaults
    - FE Ports
        - Link heart beat disabled
    - CPU Port
        - 100 M, Full Duplex, Flow Control
        - 8-byte header padding
        - per-source port buffer pool of 96 buffers, with flow control threshold of 48 buffers

Zarlink Semiconductor Inc.

1.7   Power Sequencing

The ZL50411 has two separate power supplies: V_{DD} (1.8 V) and V_{CC} (3.3 V). The recommended power-up sequence is for V_{CC} to be applied first, followed by the V_{DD} supply. V_{CC} should lead V_{DD} supply by at least 0.2 V, but by no more than 2 V.

Both supplies may be powered-down simultaneously.
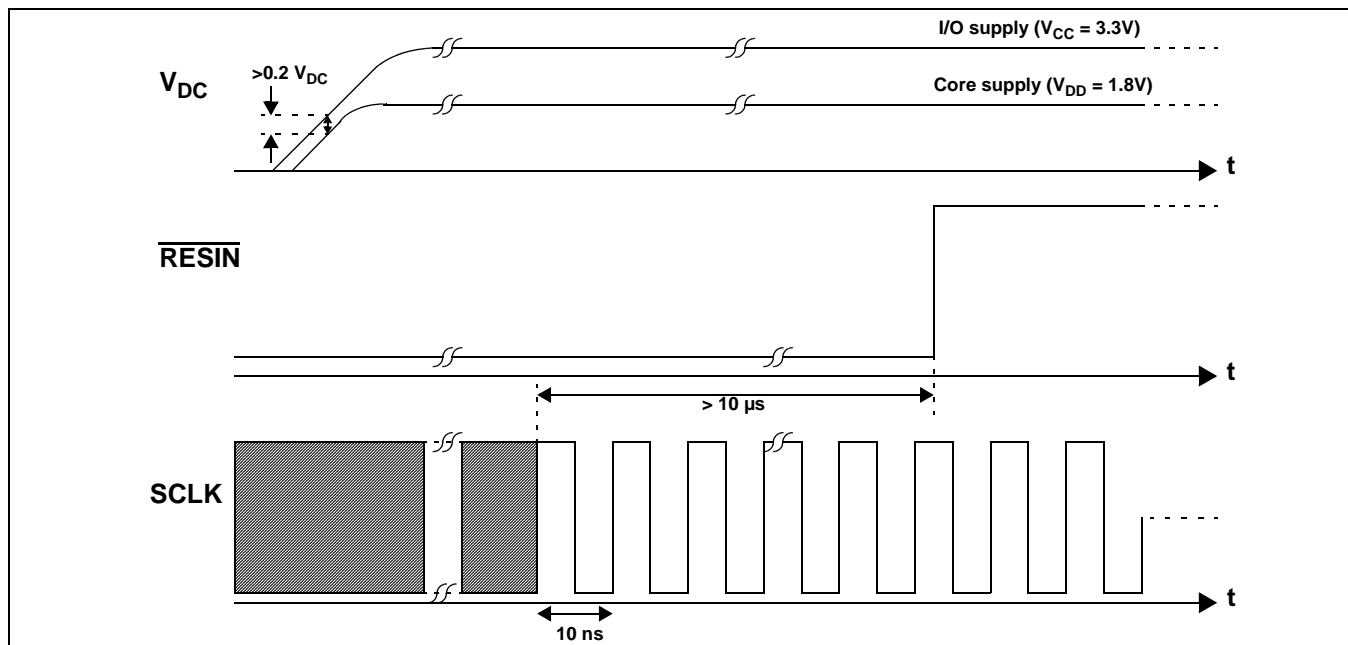
Figure 2 - Power-up Sequence

See "Typical Reset & Bootstrap Timing Diagram" on page 130 for more details on reset and bootstrap sampling.

2.0   Block Functionality
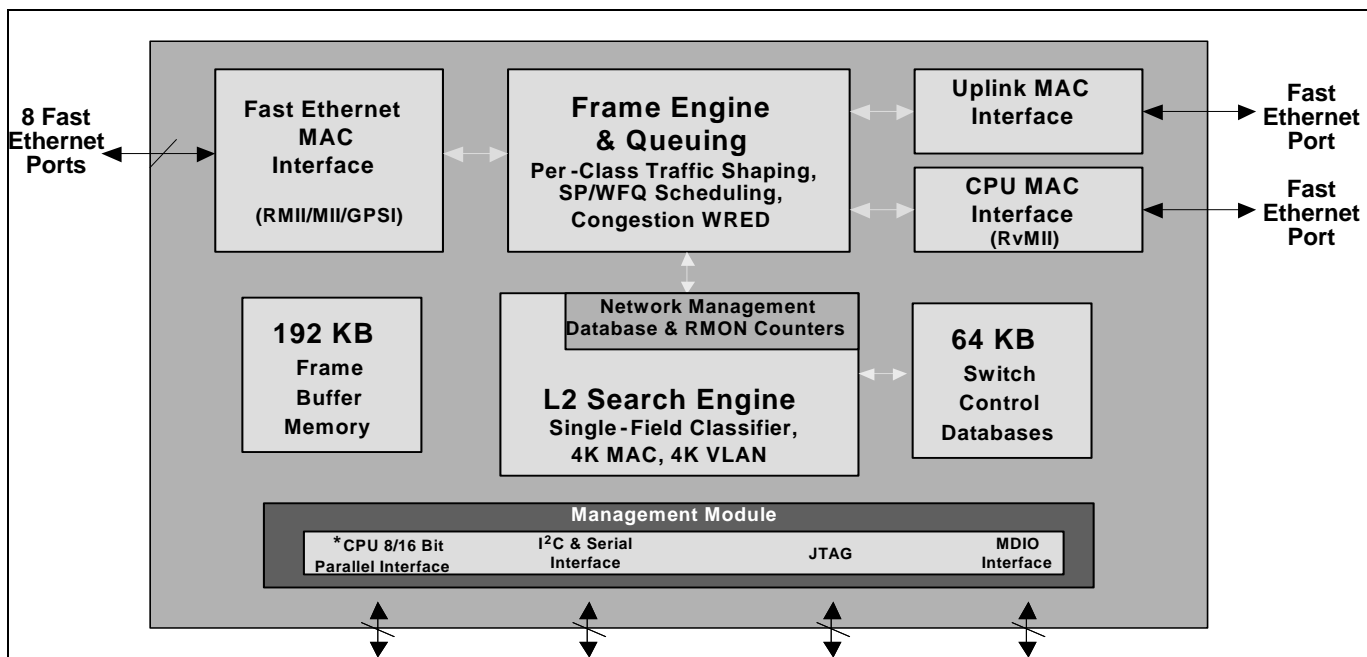
Figure 3 - Functional Block Diagram

25

Zarlink Semiconductor Inc.

2.1 Internal Memory

Two Megabit of internal memory is provided for ethernet Frame Data Buffering (FDB), storing of MAC Control Table database (MCT), and the Network Management (NM) Database statistics counters and MIB.

The MCT is used for storing MAC addresses and their physical port number. The FDB is used for storing the received frame data contents. The contents are stored in this memory until it is ready to be transmitted to the egress port.

A memory arbiter is used to arbitrary the memory access requests from various sources. A Built In Self Test (BIST) is used to detect any error in the memory array when the device is powered up. The BIST can also be requested by the writing to the GCR register.

2.2 MAC Modules

2.2.1 RMII MAC Module (RMAC)

The RMII Media Access Control (RMAC) module provides the necessary buffers and control interface between the Frame Engine (FE) and the external physical device (PHY). It has five interfaces: MII, RMII, GPSI (only for 10M), Reverse MII, or Reverse GPSI (only for 10M).

The RMAC of the ZL50411 device meets the IEEE 802.3 specification. It is able to operate in either Half or Full Duplex mode with a back pressure/flow control mechanism. In addition, it will automatically retransmit upon collision for up to 16 total transmissions.

These eight ports are denoted as ports 0 to 7. The PHY addresses for the PHY devices connected to the 8 RMAC ports has to be from 08h (port 0) to 0Fh (port 7).

2.2.1.1 GPSI (7WS) Interface

The RMAC ethernet port can function in GPSI (7WS) mode. In this mode, the TXD[0], RXD[0] serve as TX data, RX data and respectively. The link and duplex of the port can be controlled by programming the ECR1Pn register. Only port-based VLAN is supported with GPSI interface.

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 either a Reverse MII interface, providing the necessary interface TX and RX clocks to the CPU, or a register access mechanism via the 8/16-bit or serial interface.

Using the MII interface, the CMAC of the ZL50411 device meets the IEEE 802.3 specification. It is able to operate in either Half or Full Duplex mode with a back pressure/flow control mechanism. In addition, it will automatically retransmit upon collision for up to 16 total transmissions.

This port is denoted as port 8.

2.2.3 MII MAC Module (MMAC)

The MII Media Access Control (MMAC) module provides the necessary buffers and control interface between the Frame Engine (FE) and the external physical device (PHY). The MMAC implements an MII interface.

The MMAC of the ZL50411 device meets the IEEE 802.3 specification. It is able to operate in 10 M/100 M either Half or Full Duplex mode with a back pressure/flow control mechanism. Furthermore, it will automatically retransmit upon collision for up to 16 total transmissions.

This port is denoted as port 9. The PHY address for the PHY device connected to the MMAC port has to be 10h.

Zarlink Semiconductor Inc.

2.2.4 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 ZL50411 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 7	0x0F
CMAC Port 8	NA
MMAC Port 9	0x10

Table 5 - PHY Addresses

2.3 Management Module

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

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, sent to the search engine, to resolve the destination port. The arriving frame is moved to the internal memory. 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 according to the destination MAC address (L2), or IP multicast address (L3) by searching the database. It also performs unicast MAC learning, trunking functions, and priority assignment.

2.6 Heartbeat Packet Generation and Response

The ZL50411 provides the ability to monitor a link and detect a simple link failure. The Link Heart Beat (LHB) packet generation module allows simultaneous tracking of all the RMAC ports.

Periodically, a LHB message will be sent for each link when inactivity is detected with in a programmable time period, If a reply is not received in a specified amount of time, the failover detection module will identify a point-to-point failure for that link. The failover detection module will then interrupt the CPU.

The LHB packet response module can also reply to LHB messages initiated by other ZL50411 devices in the system, or by non-ZL50411 devices which use a conventional and recognizable LHB message format.

2.7 Timeout Reset Monitor

The ZL50411 supports a state machine monitoring block which can trigger a reset or interrupt 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 (TSTOUT12). It also requires some register configuration via the CPU interface.

Zarlink Semiconductor Inc.

See Programming Timeout Reset application note, ZLAN-41, for more information.

2.8 JTAG

An IEEE1149.1 compliant test interface is provided for boundary scan. The JTAG interface, collectively known as a Test Access Port, or TAP, uses the following signals to support the operation of boundary scan:

- TCK – the TCK or 'test clock' synchronizes the internal state machine operations
- TMS – the TMS or 'test mode state' is sampled at the rising edge of TCK to determine the next state
- TDI – the TDI or 'test data in' represents the data shifted into the device's test or programming logic. It is sampled at the rising edge of TCK when the internal state machine is in the correct state
- TDO – the TDO or 'test data out' represents the data shifted out of the device's test or programming logic and is valid on the falling edge of TCK when the internal state machine is in the correct state
- TRST – the TRST or 'test reset' is an optional pin which, when available, can reset the TAP controller's state machine

2.8.1 Registers

There are two types of registers associated with boundary scan. Each compliant device has one instruction register and two or more data registers.

- Instruction Register – the instruction register holds the current instruction. Its content is used by the TAP controller to decide what to do with signals that are received. Most commonly, the content of the instruction register will define to which of the data registers signals should be passed.
- Data Registers – there are three primary data registers, the Boundary Scan Register (BSR), the BYPASS register and the IDCODES register. Other data registers may be present, but they are not required as part of the JTAG standard.
    - BSR – this is the main testing data register. It is used to move data to and from the 'pins' on a device.
    - BYPASS – this is a single-bit register that passes information from TDI to TDO. It allows other devices in a circuit to be tested with minimal overhead.
    - IDCODES – this register contains the ID code and revision number for the device. This information allows the device to be linked to its Boundary Scan Description Language (BSDL) file. The file contains details of the Boundary Scan configuration for the device.

2.8.2 Test Access Port (TAP) Controller

The TAP controller, a state machine whose transitions are controlled by the TMS signal, controls the behaviour of the JTAG system. For more detail on each state, refer to the IEEE 1149.1 Standard JTAG document.

2.8.3 Boundary Scan Instructions

The IEEE 1149.1 standard defines a set of instructions that must be available for a device to be considered compliant. These instructions are:

- BYPASS – the BYPASS instruction causes the TDI and TDO lines to be connected via a single-bit pass-through register (the BYPASS register). This instruction allows the testing of other devices in the JTAG chain without any unnecessary overhead.
- EXTEST – the EXTEST instruction causes the TDI and TDO to be connected to the Boundary Scan Register (BSR). The device's pin states are sampled with the 'capture dr' JTAG state and new values are shifted into the BSR with the 'shift dr' state; these values are then applied to the pins of the device using the 'update dr' state.
- SAMPLE/PRELOAD – the SAMPLE/PRELOAD instruction causes the TDI and TDO to be connected to the BSR. However, the device is left in its normal functional mode. During this instruction, the BSR can be accessed by a data scan operation to take a sample of the functional data entering and leaving the device. The instruction is also used to preload test data into the BSR prior to loading an EXTEST instruction.

Other available instructions for the ZL50411 include:

- IDCODE – the IDCODE instruction causes the TDI and TDO to be connected to the IDCODE register.
- HIGHZ – the HIGHZ instruction causes all of the logic outputs to be placed in an inactive drive state (e.g., high impedance).

3.0   Management and Configuration

One extra port is dedicated to the CPU via the CPU interface module. Three modes this port can operate: managed, lightly managed or unmanaged mode. The different between these modes is tx/rx Ethernet frame, tx/rx control frame and receiving interrupt due to the lack of constant attention or processing power from the CPU.

The CPU interface utilizes a 8/16-bit bus in managed mode. It also supports a serial+MII, serial only, and an I^2C interface, which provides an easy and lower cost way to configure the system for reduced management.

Supported CPU interface modes are

Operation Mode	ISA Interface	Serial	MII	I²C
16-bit CPU	16-bit	NA	NA	NA
8-bit CPU	8-bit	NA	NA	NA
Serial with MII interface	NA	Yes	Yes	No
Lightly Managed Serial	NA	Yes	No	No
Unmanaged Serial	NA	Yes	No	Yes

Table 6 - Supported CPU interface modes

1. 16-bit CPU interface similar to the Industry Standard Architecture (ISA) specification.

2. 8-bit CPU interface similar to ISA.

3. Serial with MII. A synchronous serial interface (SSI) bus is used for accessing the configuration register and control frame. MII is used for sending and receiving CPU packets.

4. Lightly Managed Serial. Configuration registers access, Control frame and CPU transmit/receive packets are sent through a synchronous serial interface (SSI) bus.

5. Unmanaged Serial. The device can be configured by EEPROM using an I²C interface at bootup, or via a synchronous serial interface (SSI) otherwise. All configuration registers and internal control blocks are accessible by the interface. However, the CPU cannot receive or transmit frames nor will it receive any interrupt information.

The CPU interface provides for easy and effective management of the switching system.

Zarlink Semiconductor Inc.

Figure 4 on page 30 provides an overview of the 8/16-bit interface. Figure 5 on page 31 provides an overview of the SSI interface. Figure 6 on page 32 provides an overview of the SSI+MII interface.
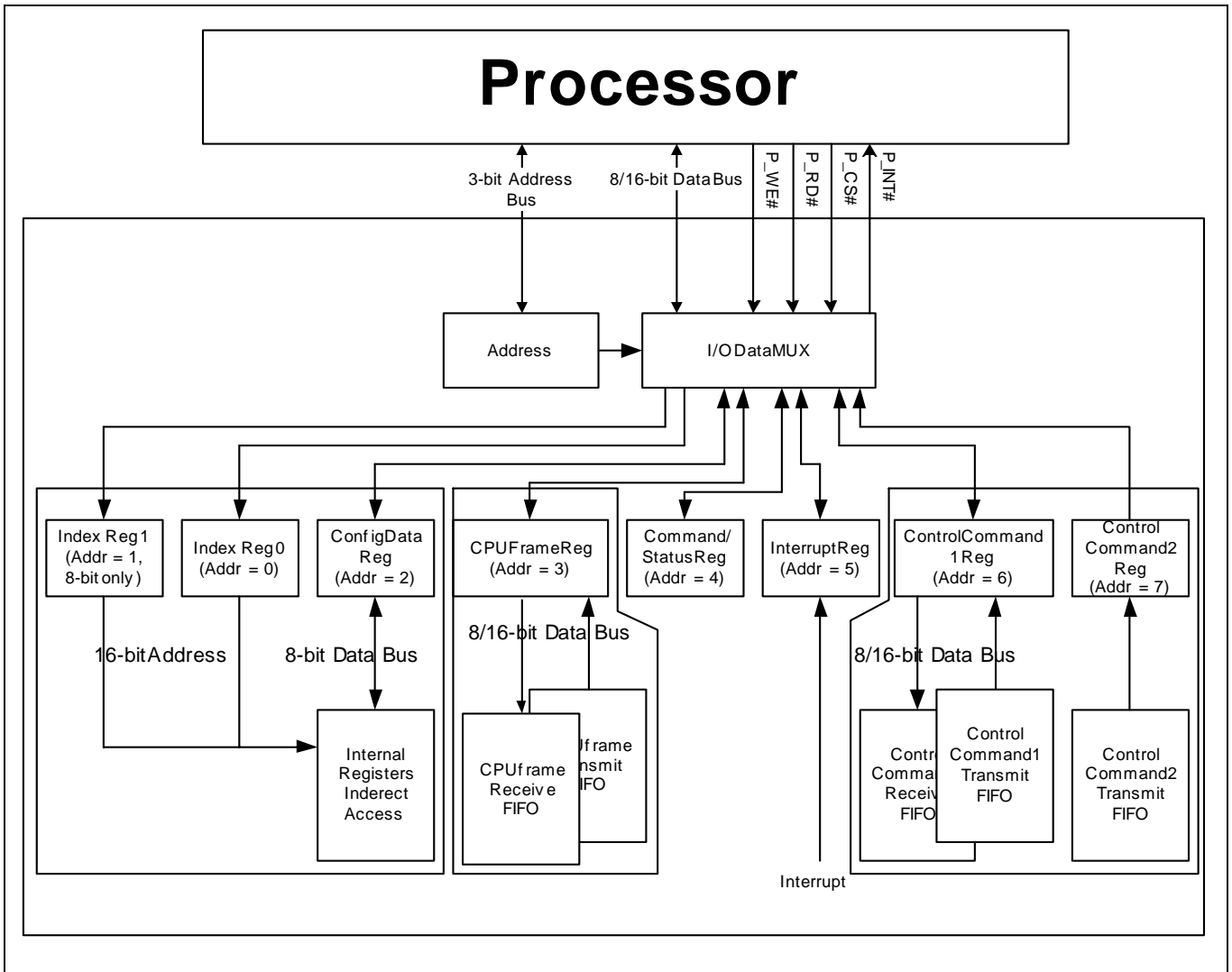
Figure 4 - Overview of the 8/16-bit Interface

Zarlink Semiconductor Inc.
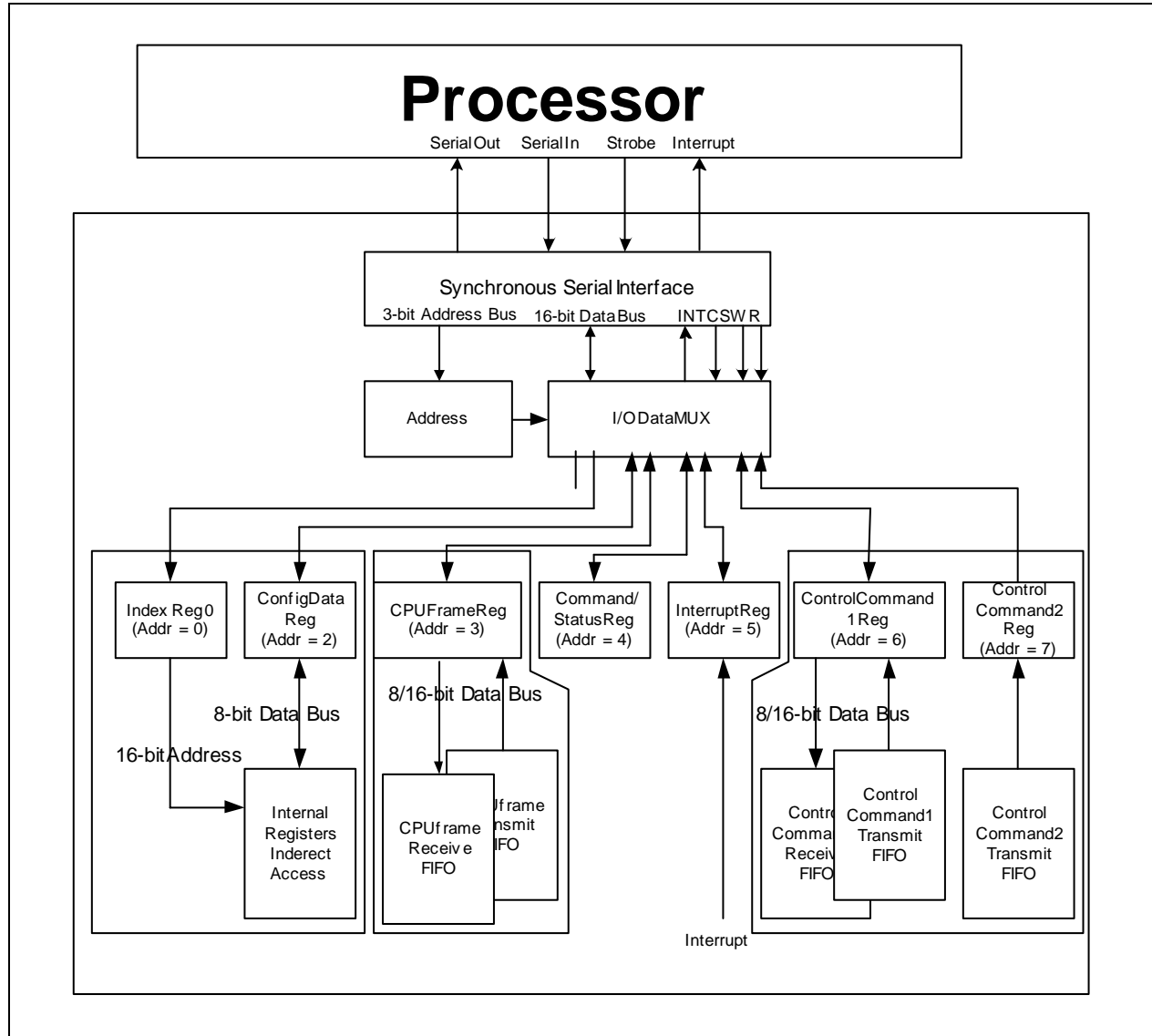
Figure 5 - Overview of the SSI Interface

Zarlink Semiconductor Inc.
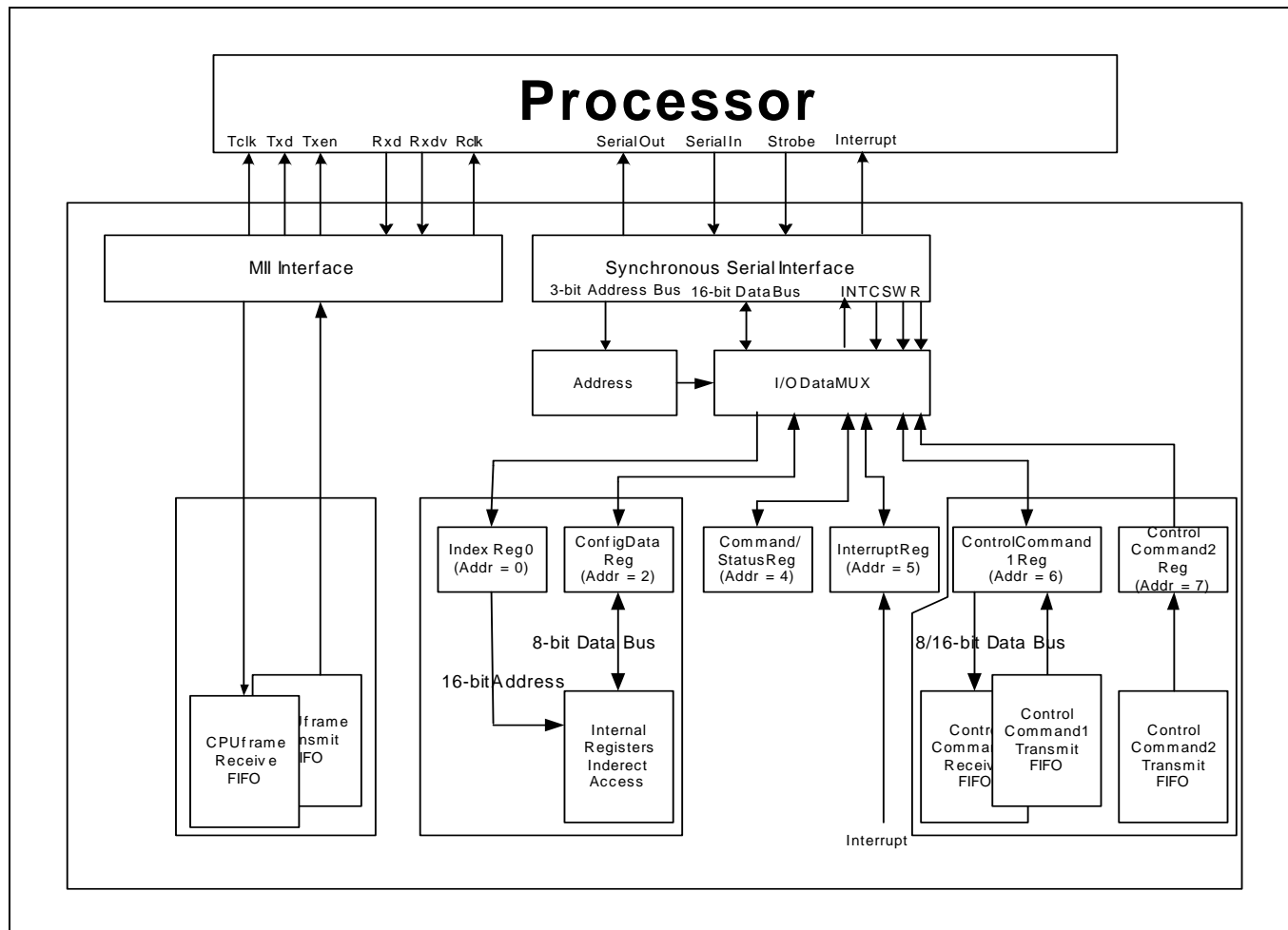
Figure 6 - Overview of the SSI+MII Interface

3.1   Register Configuration, Frame Transmission and Frame Reception

3.1.1   Register Configuration

The ZL50411 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 device allows indirect access to these registers, as follows:

- If operating in 8-bit interface mode, two "index" registers (addresses 000b and 001b) need to be written, to indicate the desired 16-bit register address. In 16-bit mode, only one register (address 000b) needs to be written for the desired 16-bit register address.
- In serial mode, the address, command and data are shifted in serially. To access the configuration registers, only one "index" register (addresses 000b) needs to be written with the configuration register address. The desired data can be written into or read from the "data" register (address 010b).
  - For example, if "XX" is required to be written to register "YY", a write of "YY" is required to write to address "000b" (Index register). Then, a write of "XX" is required to write to address "010b" (Data Register). This completes the register write and register "YY" will contain the value of "XX".

Zarlink Semiconductor Inc.

- To indirectly configure the register addressed by the index register(s), a "data" register (address 010b) must be written with the desired 8-bit data.
  - The ZL50411 supports special register-write in serial and 16-bit mode. This allows CPU to write to two consecutive configuration registers in a single write operation. By writing to bit[14] of configuration register address, CPU can write 16-bit data to address 010b. Lower 8 bit of data is for the address specified in index register and upper 8 bit of data is for the address + 1. In 8-bit mode, this special feature will be ignored.

15	14	13	12	11	0
INC R/W	SP W	Reserved		12 Bit Register Address	

- Similarly, to read the value in the register addressed by the index register(s), the "data" register can now simply be read.
- The ZL50411 supports an incremental read/write. If CPU requires to read or write to the configuration registers incrementally, CPU only has to write to index register once with the MSB of configuration register address set and then CPU can continuously reading or writing to "data" register (010b).

In summary, access to the many internal registers is carried out simply by directly accessing only two registers – one register to indicate the index 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.

3.1.2   Rx/Tx of Standard Ethernet Frames

In serial mode with MII, the MII interface is used for CPU to transmit and receive Ethernet frames. In 8/16-bit or serial only mode, the Ethernet frame is transmitted and received through the CPU interface. There is no ability to send/receive Ethernet frames in unmanaged mode.

To transmit a frame from the CPU in 8/16-bit or serial only mode:

- The CPU writes to the "data frame" register (address 011) with the frame size, destination port number, and frame status. After writing all the transmitting status bytes, it then writes the data it wants to transmit (minimum 64 bytes).
- The ZL50411 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 in 8/16-bit or serial only mode:

- 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.

To transmit a frame from the CPU with MII interface:

- ZL50411 acts as a PHY to provide receive clock (RXCLK) to CPU so the CPU will depend on this receive clock to send packets to ZL50411
- ZL50411 has the ability to halt the receive clock if the receive FIFO of ZL50411 is overflow. Transmitting from CPU to ZL50411 will resume once the receive FIFO of ZL50411 is no longer overflow
- Follow the standard Ethernet transmission format. CPU assert receive data valid (RXDV) before transmitting data to ZL50411 and de-assert RXDV after transmitting the last data

To receive a frame into the CPU with MII interface:

- ZL50411 acts as a PHY to provide transmit clock (TXCLK) to CPU so the CPU will depend on the transmit clock to receive packets from ZL50411
- ZL50411 has the ability to halt the transmit clock if the transmit FIFO of ZL50411 is under-run. CPU will resume receiving packets from ZL50411 once the transmit FIFO of ZL50411 is no longer under-run

Zarlink Semiconductor Inc.

- Follow the standard Ethernet transmission format. CPU will see transmit enable (TXEN) be asserted by ZL50411 and CPU can start receiving data. CPU will stop receiving data once TXEN is de-asserted by ZL50411.

In summary, in 8/16-bit or serial only mode, receiving and transmitting frames to and from the CPU is a simple process that uses one direct access register only. In serial mode with MII interface, the CPU will be allowed to transmit and receive frames using standard IEEE 802.3 Ethernet transmission format.

The details of sending an Ethernet Frame via the CPU interface is described in the Processor Interface Application Note, ZLAN-26.

3.1.3   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 ZL50411 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 the following types of control frames generated by the CPU and sent to the ZL50411:

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

Note: Memory read and write requests by the CPU may include all internal memories which include statistic counters, MAC address control link table and the 2 Mbit (256 KB) memory block.

In addition, the following types of Control frames are generated by the ZL50411 and sent to the CPU:

- Interrupt CPU when statistics counter rolls over
- Response to memory read request from CPU
- Learn Unicast MAC address
- Delete Unicast MAC address
- Delete Multicast MAC address
- Delete IP Multicast address
- Response to search Unicast MAC address request from CPU
- Response to search IP Multicast address request from CPU
- Response to search Multicast MAC address request from CPU

The format of the Control Frame is described in the Processor Interface application note, ZLAN-26.

3.2   I²C Interface

The I²C interface serves the function of configuring the ZL50411 at boot time. The master is the ZL50411, and the slave is the EEPROM memory.

The I²C 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 /

Zarlink Semiconductor Inc.

acknowledgment style of protocol. The master IC generates the timing signals and terminates data transfer. Figure 7 depicts the data transfer format. The slave address is the memory address of the EEPROM. Refer to "ZL50411 Register Description" on page 62 for I²C address for each register.

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

Figure 7 - Data Transfer Format for I²C Interface

3.2.1 Start Condition

Generated by the master (in our case, the ZL50411). 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 I²C bus is free, both lines are High.

3.2.2 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.

3.2.3 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.

3.2.4 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.

3.2.5 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.

3.2.6 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 Synchronous Serial Interface

The synchronous serial interface (SSI) serves the function of configuring the ZL50411 not at boot time but via a PC. The PC serves as master and the ZL50411 serves as slave. The protocol for the synchronous serial interface is nearly identical to the I²C protocol. The main difference is that there is no acknowledgment bit after each byte of data transferred. Debounce logic on the clock signal (STROBE) can be turned off to speedup command time.

3 ID bits are used to allow up to eight ZL50411 devices to share the same synchronous serial interface. The ID of each device can be setup by bootstrap.

To reduce the number of signals required, the register address, command and data are shifted in serially through the DATAIN pin. STROBE- pin is used as the shift clock. DATAOUT 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

Write operation can be aborted in the middle by sending an ABORT pulse to the ZL50411. Read operation can only be aborted before issuing the read command to the ZL50411.

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

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

3.3.1 Write Command

All registers in ZL50411 can be modified through this synchronous serial interface. Once the data has been sent, two extra STOBE clocks must be generated to indicate the end of the write command. The DATAIN line is ignored for these two pulses.
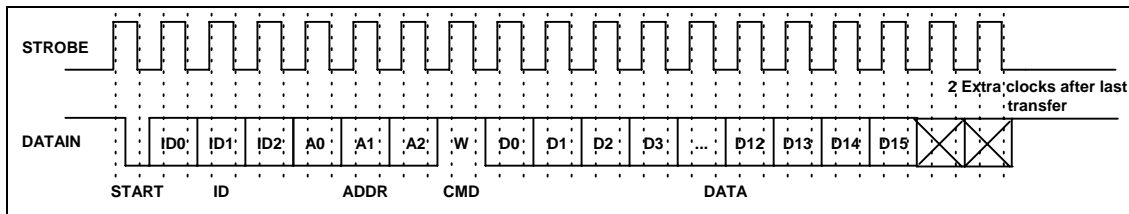
Figure 8 - Serial Interface Write Command Functional Timing

3.3.2 Read Command

All registers in ZL50411 can be read through this synchronous serial interface.
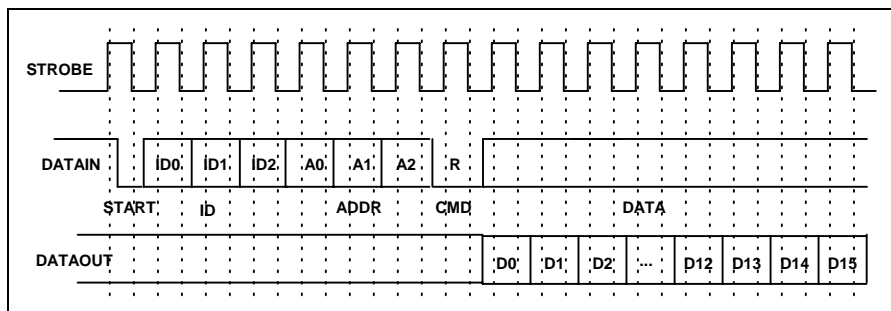
Figure 9 - Serial Interface Read Command Functional Timing

36

Zarlink Semiconductor Inc.

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

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

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

After processing the switch response, the Transmission Queue Manager (TxQ manager) of the frame engine is responsible for notifying the destination port that it has a frame to forward. 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. The switch response will come with 8 classified results. The TxQ manager will map this result into the per-port-per-class queue. 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 2 transmission classes for each of the 8 RMAC ports, and 4 classes for the MMAC and CPU ports – a total of 24 unicast queues.

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

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

The search engine will resolve the destination ports of the multicast frame. This will be all ports within the VLAN (L2 multicast), or the ports contained in the portmap of the learned multicast MAC address or IP multicast group. 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 8 RMAC ports. There are 4 multicast queues for the MMAC and CPU ports. The mapping from the classified result to the priority queue is the same as the unicast traffic. By default, for the RMAC ports to map the 8 transmit priorities into 2 multicast queues, the 2 LSB are discarded. For the MMAC

Zarlink Semiconductor Inc.

and CPU ports, to map the 8 transmit priorities into 4 multicast queues, the LSB is discarded. The priority mapping can be modified through memory configuration command. The multicast queue that is in FIFO format shares the space in the internal memory block. The size and starting address can also be programmed through memory configuration command.

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   Search Engine

5.1   Search Engine Overview

The ZL50411 search engine is optimized for high throughput searching, with enhanced features to support:

- Up to 4 K of Unicast/Multicast MAC addresses and IP Multicast addresses
- Up to 4 K VLANs
- Private VLAN Edge (protected ports) support
- Up to 8 groups of port trunking
- Traffic classification into 2 (or 4 for MMAC) transmission priorities, and 2 drop precedence levels
- Packet filtering based on MAC address, Protocol or Logical Port number
- Security
- Up to 4 K IP Multicast groups
- Individual Flooding, Broadcast, Multicast Storm Control
- MAC address learning and aging

5.2   Basic Flow

Shortly after a frame enters the ZL50411 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 packet's VLAN ID, and whether the frame is unicast or multicast or broadcast. Requests are sent to the SRAM to locate the associated entries in the MCT table.

When all the information has been collected from the 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. The selection of the port within the trunk that will transmit the frame is computed using a hash of the source and/or destination MAC addresses.

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

5.3 Search, Learning and Aging

5.3.1 MAC Search

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

In tag-based VLAN mode, if the frame is unicast, and the frame's destination port is recognized as a member of the VLAN, then the frame is forwarded to that port; otherwise, the frame is forwarded to all the members in the VLAN domain. If the frame is multicast or broadcast, the frame is forwarded to all the members in the VLAN, or those ports in the learned mulicast MAC address's or IP multicast group's portmap. In private VLAN Edge (protected ports) mode, the private VLAN domain is used for final qualification at the egress port.

In port based VLAN mode, a bit map is used to determine whether the frame should be forwarded to the outgoing port. The main difference in this mode is that the bit map 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 used for aging. Moreover, if port trunking is enabled, this block selects the destination port (among those in the trunk group).

5.3.2 Learning

The learning module learns new MAC addresses and performs port change operations on the MCT database. The goal of learning is to update this database as the networking environment changes over time.

When CPU reporting is enabled, learning and port change will be performed when the CPU request queue has room, and a "Learn MAC Address" message is sent to the CPU.

5.3.3 Aging

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

The aging module scans and ages MCT entries based on a programmable "age out" time interval. As we indicated earlier, the search module updates the source MAC address 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, multicast, source filter, destination filter, source and destination filter, and secure MAC addresses. The only supported IP entry type is IP multicast. Only dynamic entries can be aged; all others are static. The entry type is stored in the "status" field of the MCT data structure.

5.4 MAC Address Filtering

The ZL50411'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.

Broadcast, unknown unicast or unknown multicast MAC address and unknown IP multicast address can also be filter on per VLAN basis.

MAC address filtering allows the ZL50411 to block an incoming packet to an interface when it sees a specified MAC address in either the source address or destination address of the incoming packet. For example, if your network is congested because of high utilization from a MAC address, you can filter all traffic transmitted from that address and restore network flow, while you troubleshoot the problem.

5.5   Protocol Filtering

Packet filtering can be performed based on protocol type field in the packets. Up to eight protocols can be programmed to filter or allow packet to pass through the switch.

5.6   Logical Port Filtering

Similar to protocol filtering, if the packet's logical ports match the programmable registers, the packet can be filtered or passed through the switch. Up to eight programmable ports and one ranges can be assigned.

5.7   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 ZL50411 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 ZL50411, 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 ZL50411 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.

Zarlink Semiconductor Inc.

5.8 Priority Classification Rule

Figure 10 shows the ZL50411 priority classification rule.
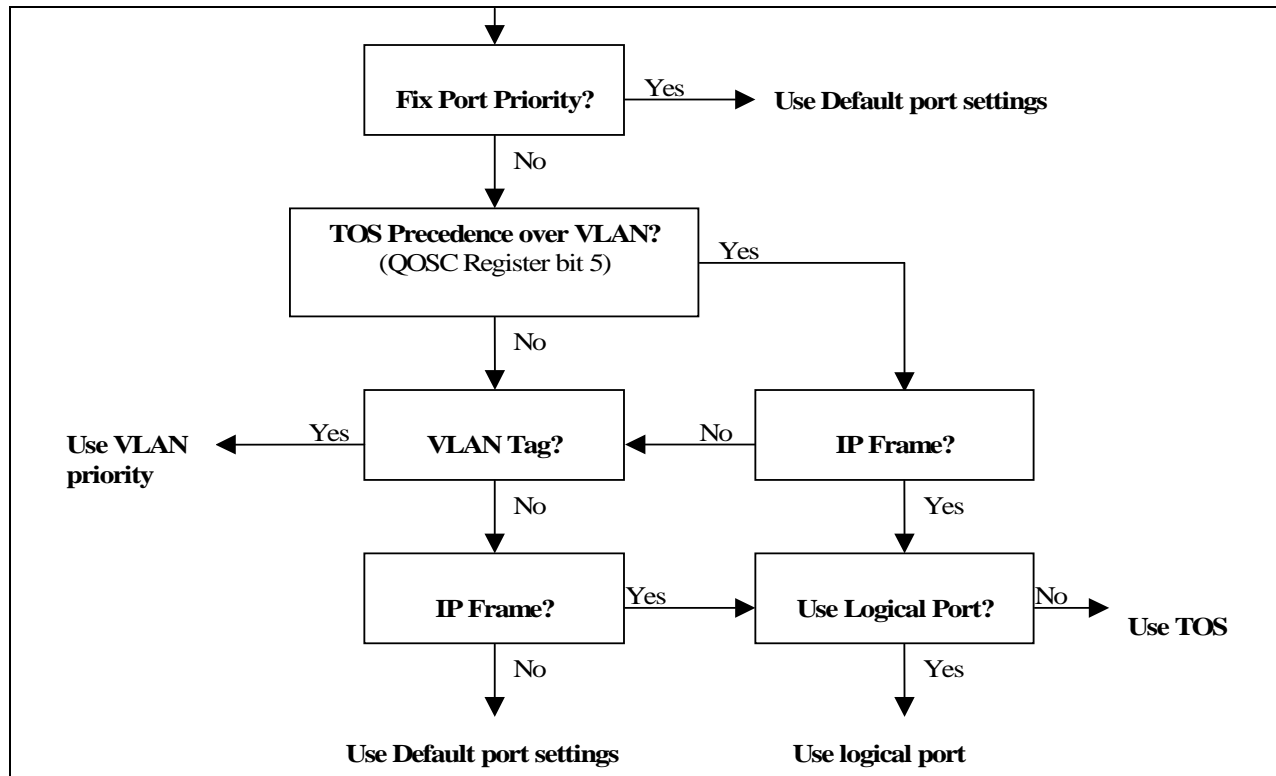
Figure 10 - Priority Classification Rule

5.9 Port and Tag Based VLAN

The ZL50411 supports two models for determining and controlling how a packet gets assigned to a VLAN: port priority and tag -based VLAN.

5.9.1 Port-Based VLAN

An administrator can use the PVMAP Registers to configure the ZL50411 for port-based VLAN (see "Register Definition" on page 62). 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 ZL50411 determines the VLAN membership of each packet by noting the port on which it arrives. From there, the ZL50411 determines which outgoing port(s) is/are eligible to transmit each packet, or whether the packet should be discarded.

	Destination Port Numbers Bit Map				
Port Registers	9	…	2	1	0
Register for Port #0 PVMAP00_0[7:0] to PVMAP00_1[1:0]	0		1	1	0
Register for Port #1 PVMAP01_0[7:0] to PVMAP01_1[1:0]	0		1	0	1
Register for Port #2 PVMAP02_0[7:0] to PVMAP02_1[1:0]	0		0	0	0
…					
Register for Port #9 PVMAP09_0[7:0] to PVMAP09_1[1:0]	0		0	0	0

Table 7 - Port-Based VLAN Mapping

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

In this example:

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

5.9.2   Tag-Based VLAN

The ZL50411 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 ZL50411 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 4 K VLANs are supported in the ZL50411. When tag-based VLAN is enabled, each MAC address is learned with it associated VLAN.

See IEEE 802.1Q VLAN Setup application note, ZLAN-51, for more information.

5.9.3   VLAN Stacking (Q-in-Q)

The ZL50411 partially supports VLAN stacking, also called IEEE 802.1Q-in-Q. This technology allows an additional VLAN tag, called a provider VLAN tag, to be inserted into an existing IEEE 802.1Q tagged Ethernet frame. This technology has been widely adapted in Metro Ethernet applications since it provides a very cost-effective solution to transport multiple customers' VLAN across the service provider's MAN/WAN without interfering each other. The below figure illustrates the IEEE 802.1Q frame and the Q-in-Q frame, where the provider VLAN tag is inserted in front of the IEEE 802.1Q tag.

Zarlink Semiconductor Inc.
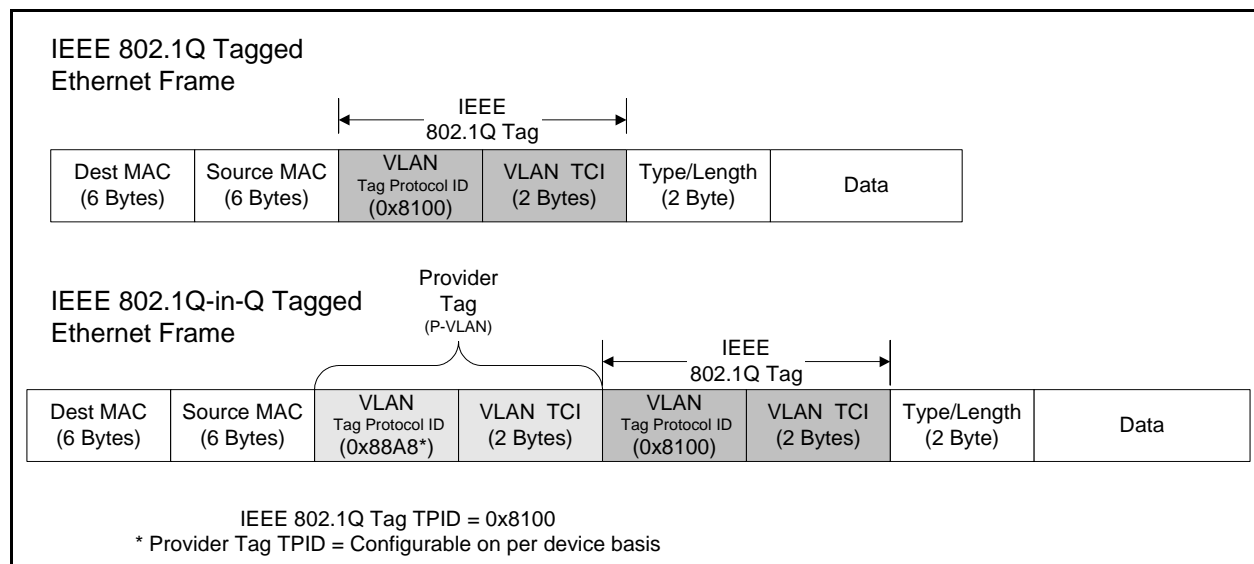
Figure 11 - Q-in-Q Tagged Ethernet Frame

The value of the TPID of the Provider VLAN tag is not assigned in the IEEE 802.1ad standard. The ZL50411 provides a global configurable TPID but only supports the Extreme EtherType TPID (i.e. the stacked VLAN tag cannot equal 0x81-00).

See Stacked VLAN application note, ZLAN-82, for more information.

5.9.4 Private VLAN Edge

The PVLAN Edge (protected port) is a feature that has only local significance to the switch (unlike Private VLANs), and there is no isolation provided between two protected ports located on different L2 switches. A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port in the same switch. Traffic cannot be forwarded between protected ports at L2, thus, all traffic passing between protected ports must be forwarded through a Layer 3 (L3) device.

The ZL50410 supports private VLAN Edge by allowing each of the RMAC ports to setup a private VLAN domain, defined by registers PVLAN_Pn. The private VLAN domain is used for final qualification at the egress port.

See Private VLAN Edge application note, ZLAN-130, for more information.

5.10 IP Multicast Switching

The ZL50411 supports IP Multicast Filtering by:

- Passively snooping on the IGMP Query and IGMP Report packets transferred between IP Multicast Routers and IP Multicast host groups to learn IP Multicast group members, and
- Actively sending IGMP Query messages to solicit IP Multicast group members.

The purpose of IP multicast filtering is to optimize a switched network performance, so multicast packets will only be forwarded to those ports containing multicast group hosts members and routers instead of flooding to all ports in the subnet (VLAN).

The ZL50411 with IP multicast filtering/switching capability not only passively monitor IGMP Query and Report messages, DVMRP Probe messages, PIM, and MOSPF Hello messages; they also actively send IGMP Query messages to learn locations of multicast routers and member hosts in multicast groups within each VLAN.

See IP Multicast Switching application note, ZLAN-52, for more information.

5.11   L2 Multicast Switching

The ZL50411 supports multicast MAC address learning when in tagged-based VLAN mode. The purpose of multicast MAC learning is to optimize a L2 switched network performance, so multicast packets will only be forwarded to those ports configured to receive that multicast traffic instead of flooding to all ports within the VLAN.

6.0   Frame Engine

6.1   Data Forwarding Summary

When a frame enters the device at the RxMAC, the RxDMA will move the data from the MAC RxFIFO to the FDB. Data is moved in 8-byte granules in conjunction with the scheme for the SRAM interface.

A switch request is sent to the Search Engine. The Search Engine processes the switch request.

A switch response is sent back to the Frame Engine and indicates whether the frame is unicast or multicast, and its destination port or ports. On receiving the response, the Frame Engine will check all the QoS related information and decide if this frame can be forwarded.

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 2 TxSch Q for each RMAC port (and 4 per MMAC and CPU ports), one for each priority.   Creation of a queue entry either involves linking a new job to the appropriate linked list if unicast, or adding an entry to a physical queue if multicast.

When the port is ready to accept the next frame, the TxQ manager will get the head-of-line (HOL) entry of one of the TxSch Qs, according to the transmission scheduling algorithm (so as to ensure per-class quality of service). (The unicast linked list and the multicast queue for the same port-class pair are treated as one logical queue. The older HOL between the two queues goes first.

The TxDMA will pull frame data from the memory and forward it granule-by-granule to the MAC TxFIFO of the destination port.

6.2   Frame Engine Details

This section briefly describes the functions of each of the modules of the ZL50411 frame engine.

6.2.1   FCB Manager

The FCB manager allocates FCB handles to incoming frames, and releases FCB handles upon frame departure. The FCB manager is also responsible for enforcing buffer reservations and limits that will be used for QoS control and source port flow control. The default values can be determined by referring to Section 7.6 on page 47. The frame buffer is managed in a 128bytes block unit. During initialization, this block will link all the available blocks in a free buffer list. When each port is ready to receive, this module hands the buffer handle to each requesting port. The FCB manager will also link the released buffer back into the free buffer list.

The maximum buffer size can be increased from the standard 1518 bytes (1522 with VLAN tag) to up to 4 K bytes. This is done using BUF_LIMIT, and is enabled on a per port basis via bit [1] in ECR3Pn. See Buffer Allocation application note, ZLAN-47, for more information.

6.2.2   Rx Interface

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

Zarlink Semiconductor Inc.

6.2.3   RxDMA

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

6.2.4   TxQ Manager

First, the TxQ manager checks the per-class queue status and global reserved resource situation, and using this information, makes the frame dropping decision after receiving a switch response. The dropping decision includes the head-of-link blocking avoidance if the source port is not flow control enabled. If the decision is not to drop, the TxQ manager links the unicast frame's FCB to the correct per-port-per-class TxQ and updates the FCB information. If multicast, the TxQ manager writes to the multicast queue for that port and class and also update the FCB information including the duplicate count for this multicast frame. 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. The detail of the QoS decision guideline is described in chapter 5.

6.2.5   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.

6.2.6   TxDMA

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

7.0   Quality of Service and Flow Control

7.1   Model

Quality of service is an all-encompassing term for which different people have different interpretations. In general, the approach to quality of service described here assumes that we do not know the offered traffic pattern. We also assume that the incoming traffic is not policed or shaped. Furthermore, we assume that the network manager knows his applications, such as voice, file transfer, or web browsing, and their relative importance. The manager can then subdivide the applications into classes and set up a service contract with each. The contract may consist of bandwidth or latency assurances per class. Sometimes it may even reflect an estimate of the traffic mix offered to the switch. As an added bonus, although we do not assume anything about the arrival pattern, if the incoming traffic is policed or shaped, we may be able to provide additional assurances about our switch's performance.

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.

Each traffic type may each have their own distinct properties and applications. Classes may receive bandwidth assurances or latency bounds. For example, the highest transmission class may require that all frames receive 50% of the 100 Mbps of bandwidth at that port.

Best-effort (P0) traffic forms a class that only receives bandwidth when none of the other classes have any traffic to offer. It is also possible to add a class that has strict priority over all others; if this class has even one frame to transmit, then it goes first. In the ZL50411, each RMAC port will support two total classes, and the MMAC port will support four 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. For example, casual web browsing fits into the category of high-loss, high-latency-tolerant traffic, whereas VoIP fits into the category of low-loss, low-latency traffic.

7.2   Two QoS Configurations

There are two basic pieces to QoS scheduling in the MMAC port of ZL50411: strict priority (SP) or weighted fair queuing (WFQ). The only configuration for a RMAC and CPU port is strict priority between the queues.

7.2.1   Strict Priority

When strict priority is part of the scheduling algorithm, if a queue has any frame to transmit, it goes first. For RMAC ports, this is an easy way to provide the different service. For all recognizable traffic, the bandwidth is guaranteed to 100% of the line rate. This scheme works as long as the overall high priority bandwidth is not over the line rate and the latency on all the low priority traffic is don't care. The lowest priority queue is treated as "best effort" queue.

The strict priority queue in the MMAC and CPU ports is similar to RMAC ports other than having 4 queues instead of 2 queues. The priority queue P0 can be scheduled only if the priority queue P1 is empty, so as to priority queues P2 and P3.

Because we do not provide any 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 ZL50411, 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 queue size is too long or global / class buffer resources become scarce.

7.2.2   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 strict assurance scheduling discipline. The ZL50411 provides this kind of scheduling algorithm on MMAC port only. The user sets four WFQ "weights" such that all weights are whole numbers and sum to 64. This provides per-class bandwidth partitioning with granular within 2%.

In WFQ mode, though we do not assure frame latency, the ZL50411 still retains a set of dropping rules that helps to prevent congestion and trigger higher level protocol end-to-end flow control.

7.3   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.

	Px > WRED_L1	Px > WRED_L2	BM Reject
High Drop	X%	100%	100%
Low Drop	Y%	Z%	100%

Table 8 - WRED Logic Behaviour

Px is the total byte count, in the priority queue x, can be the strict priority queue of RMAC ports and higher 3 priority queues for MMAC port. The WRED logic has two drop levels, depending on the value of Px. Each drop level 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. All packets will be dropped only if the system runs out of the specific buffer resource, per class buffer or per source port buffer. The

WRED thresholds of each queue can be programmed by the QOS control registers (refer to the register group 8). See Programming QoS Registers application note, ZLAN-42, for more information.

7.4   Shaper

Although traffic shaping is not a primary function of the ZL50411, the chip does implement a shaper for every queue in the MMAC port. Our goal in shaping is to control the average rate of traffic exiting the ZL50411. If shaper is enabled, strict priority will be applied to that queue. The priority between two shaped queue is the same as in strict priority scheduling.

Traffic rate is set using a programmable whole number, no greater than 64. For example, if the setting is 32, then the traffic rate transmit out of the shaped queue is 32/64 * 1000 Mbps = 500 Mbps. See Programming QoS Register application note, ZLAN-42, for more information.

Also, when shaping is enabled, it is possible for a queue to explode in length if fed by a greedy source. The reason is that a shaper is by definition not work-conserving; that is, it may hold back from sending a packet even if the line is idle. Though we do have global resource management, we do nothing other than per port WRED to prevent this situation locally. We assume the traffic is policed at a prior stage to the ZL50411 or WRED dropping is fine and shall restrain this situation.

7.5   Rate Control

The ZL50411 provides a rate control function on its RMAC ports. The concept is much the same as shaping, except that it applies to both ingress and egress directions and the control is per port rather than per queue. It provides a way of reducing the total bandwidth of all frames received from or transmitted to a port, to a rate below wire speed. As with shaping, the maximum burst size can also be configured.

Rate control may be a valuable feature on RMAC ports in access applications where the service provider would like to limit the traffic received and transmitted by each port independently of each other, and independently of the physical line rate. The service provider can then provide differential pricing, based on the negotiated bandwidth requirements for each user. In such applications of the ZL50411, the MMAC port is viewed as an uplink port, where rate control is not desired.

See Rate Control application note, ZLAN-33, for more information.

7.6   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 ZL50411. Our buffer management scheme is designed to divide the total buffer space into numerous reserved regions and one shared pool, as shown in Figure 12 on page 48.

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 ZL50411, 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.

Three priority sections, one for each pair 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, a frame is stored in the region of the FDB corresponding to its class. As we have indicated, the eight classes use only two transmission scheduling queues for RMAC ports (four queues for the MMAC & CPU 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 10 ports — 9 ports for Ethernet and one CPU port (port number 8). Each port has it's own programmable source port reservation. These 10 reserved regions make sure that no well-behaved source port can be blocked by another misbehaving source port.

Zarlink Semiconductor Inc.

In addition, there is a shared pool, which can store any type of frame. The frame engine allocates the frames first in the three 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_N - Port Reservation for RMAC Ports
- PR100_CPU - Port Reservation for CPU Port
- PRM - Port Reservation for MMAC Port
- SFCB - Share FCB Size
- C1RS - Class 1 Reserve Size (priority 2 & 3)
- C2RS - Class 2 Reserve Size (priority 4 & 5)
- C3RS - Class 3 Reserve Size (priority 6 & 7)
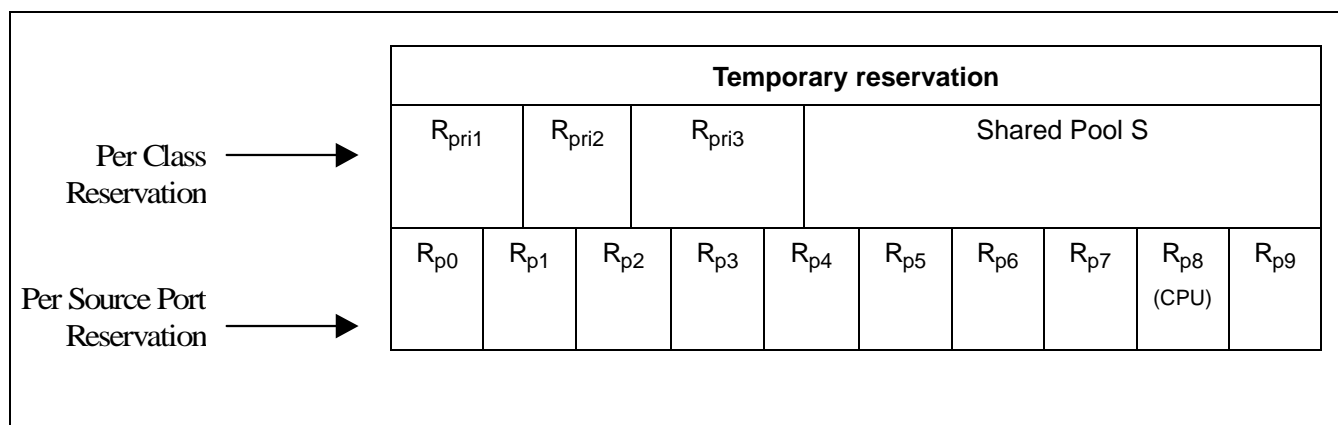
Figure 12 - Buffer Partition Scheme

See Buffer Allocation application note, ZLAN-47, for more information.

7.6.1　Dropping When Buffers Are Scarce

As already discussed, the WRED mechanism may drop frames on output queue status. 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.

If a received frame is dispatched to the P0 ("best effort") queue, the buffer management will check on the overall buffer situation plus the output queue status to decide the frame drop condition. If the source port has not enough buffer for it, the frame will be dropped. If the output queue reaches PCC (packet congest control) and the shared buffer has run out, the frame may be dropped by RED (at B%). If the output queue reaches PCC and the source port reservation is lower than the buffer low threshold, the frame will be dropped.

For multicast frames, global thresholds are used to manage scarce resources. When a multicast frame is received, the buffer management will check on the overall buffer situation. If total multicast resources reaches MCCTH (multicast congest control threshold), the frame may be dropped by RED (at B%). If total multicast resources reaches MCC, the frame will be dropped.

All the 100% dropping functions are disabled if the source port is flow control capable, however, RED (at B%) will still be applied.

Zarlink Semiconductor Inc.

7.7 Flow Control Basics

Because frame loss is unacceptable for some applications, the ZL50411 provides a flow control option. When flow control is enabled, scarcity of source port buffer space may trigger a flow control signal; this signal tells a source port sending a packet to this switch, to temporarily hold off.

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

On the other hand, the ZL50411 will still prioritize the received frame disregarding the outgoing port flow control capability. If a frame is classified as high priority, it is still subjected to the WRED, which means the no-loss on the high priority queue is not guaranteed. To resolve this situation, the user may set the output port WRED threshold so high that may never be reached, or program the priority mapping table in the queue manager to map all the traffic to "best effort" (P0) queue on the flow control capable port. The first method has side impact on the global resource management since the port may hold too much per class resource that is scarce in the system. The second method, by nature, lost the benefit of prioritization.

See Programming Flow Control Registers application note, ZLAN-44, for more information.

7.7.1 Unicast Flow Control

For unicast frames, flow control is triggered by source port resource availability. Recall that the ZL50411'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 ZL50411's per-source-port FDB reservations assure that a source port that sends a single frame to a congested destination will not be flow controlled.

7.7.2 Multicast Flow Control

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.

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

7.8 Mapping to IETF Diffserv Classes

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

ZL50411	P3	P2	P1	P0
IETF	NM+EF	AF0	AF1	BE

Table 9 - Mapping to IETF Diffserv Classes for M**MAC & CPU Ports**

As the table illustrates, the classes of Table 9 are merged in pairs— P3 is used for network management (NM) and expedited forwarding service (EF) frames. 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 ZL50411 that correspond to the requirements of their associated IETF classes are summarized in the table below.

Network management (NM) and Expedited forwarding (EF)	Global buffer reservation for NM and EF Shaper for traffic on uplink port No dropping if admission controlled
Assured forwarding (AF)	Global buffer reservation for two AF classes Shaper for traffic on uplink port Random early discard, with programmable levels
Best effort (BE)	Service only when other queues are idle means that QoS not adversely affected Shaper for traffic on uplink port Random early discard, with programmable levels Traffic from flow control enabled ports automatically classified as BE

Table 10 - ZL50411 Features Enabling IETF Diffserv Standards

7.9 Failover Backplane Feature

The ZL50411 implements a hardware assisted link failure detection mechanism utilizing a Link Heart Beat (LHB) packet. The LHB packet format is defined as a 64-byte MAC control frame with a user defined opcode. The packet format is illustrated below:

01-80-C2-00-00-01	xx-xx-xx-xx-xx-xx	88-08	yy-yy	00-00-...	CRC

Where "xx-xx-xx-xx-xx-xx" is the source port MAC address and "yy-yy" is the special opcode defined by register setup (LHBReg0,1). The opcode "00-01" is reserved for the flow control packet. We recommend opcode "00-12" for the LHB packet.

The LHB is done between two compatible MACs providing this function. A timer parameter will be set for both the receiver and transmitter (LHBTimer).

On the transmission side, the MAC will monitor the transmission activities. If there is no activity for more than the set period, a LHB packet will be sent to its link partner. Therefore, there should always be at least one packet transmitted from the MAC for every period specified.

On the receiving side, the MAC will also monitor the activity. If there is no good packet received for more than 2X the set period, an alarm will be raised to the CPU. The LHB packet is only used by the ZL50411 to reset the timeout counter, it is ignored otherwise (i.e. not passed on within the system).

See the Failover Protection Application Note, ZLAN-43, for more information.

8.0 Port Trunking

See Port Trunking application note, ZLAN-48, for more information.

8.1 Features and Restrictions

A port group (i.e., trunk) can include up to 8 physical ports to form a fault tolerant link. There are eight trunk groups total.

Load distribution among the ports in a trunk for unicast is performed using hashing based on source MAC address and destination MAC address. Three other options include source MAC address only, destination MAC address only, and source port (in bidirectional ring mode only). Load distribution for multicast is performed similarly.

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

The ZL50411 also provides a safe fail-over mode for port trunking automatically. If one of the ports in the trunking group goes down, the ZL50411 can redistribute the traffic over to the remaining ports in the trunk with software assistance.

8.2 Unicast Packet Forwarding

The search engine finds the destination MCT entry, and if the status field says that the destination port found belongs to a trunk, then the trunk group number is retrieved.

The source port of the packet is checked against the destination trunk group. If the source port belongs to the destination trunk group, the packet is discarded.

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. Each trunk has eight trunk_hash registers which selects one of the potential eight outgoing ports. The hash key provides a pseudo flow identifier which force the same flow to the same destination flow. As a result, the packet will always arrive in order.

8.3 Multicast Packet Forwarding

Multicast frame forwarding is only slightly more complex than unicast frame forwarding in a port trunking environment:

1. Determine the VLAN to which the multicast frame belongs.
2. Is the frame's source port a member of the VLAN? If so, exclude that port. Is the frame's source port a member of a trunk group? If so, exclude all other ports in that group as well.
3. Finally, among the still-eligible members of the VLAN that belong to trunk groups, exclude all but one port from each trunk group. This selection is based on hashing or round robin, as in unicast frame forwarding.

Example: Ports 0, 1, and 2 belong to trunk group A. Ports 3 and 4 belong to trunk group B. A multicast frame is received from source port 3. Suppose that the VLAN of the frame has member ports 0, 1, 2, 3, 4, 5, and 6.

- Ports 0, 1, 2, 3, 4, 5, and 6 are potential destination ports for the frame.
- Port 3 is excluded as a destination because it is the frame's source port.
- Port 4 is excluded as a destination because it belongs to the same trunk group as port 3.
- Two of the three ports in trunk group A must now be excluded. The ZL330ENG hashing algorithm is applied, and in this example, assume that ports 0 and 1 are excluded.
- Therefore, ports 2, 5 and 6 comprise the final set of destination ports for the multicast frame.

9.0 Traffic Mirroring

See Traffic Mirroring application note, ZLAN-50, for more information.

9.1 Mirroring Features

Packets can be mirrored (duplicated) for network monitor purpose and/or network debug purpose. Three types of mirroring is available in ZL50411.

1. Source or Destination MAC address based
2. Flow based
3. Port based (RMII mode only)

In source or destination mac address based mirroring, the "M" bit of the mirroring MAC address in the MCT is set. Also, the user need to specify the mirroring MAC address is source or destination of the packet. If source is selected, any packet received with the mirroring MAC address as source MAC address will be copied to the

mirrored port. In the same way, if destination is selected, any packet received with mirroring MAC address as destination MAC address will be copied to the mirrored port.

In flow based mirroring, a flow is established based on the source and destination mac address pair. When enabled, a packet with source and destination address match the pre-programmed source and destination mac address pair will be copied to the mirrored port. In reverse direction (source and destination match pre programmed destination and source), the flow can also be enabled and the frame will be copied to the mirrored port.

In port based mirroring, traffic from any RMII port can be mirrored to any other RMII port. The traffic from the source port can be either ingress or egress traffic. Up to two ports can be setup as mirrored ports. As a result, the traffic (both ingress and egress) of a specific port can be monitored by setting up both mirrored ports. Once a port is setup as mirrored port, it cannot be used for regular traffic.

The mirrored port can be any port in the ZL50411.

9.2 Using port mirroring for loop back

To perform remote loop back test, port mirroring can be used to bounce back the packet to the source port to check the data path.

The CPU needs to setup the remote device through the command channel to enable port mirroring in the remote device. A CPU packet is send to the port in test in Device A. The packet will be forwarded to the test port, external cable, the destination port in Device B, and loop back to itself, back to the cable and go back to Device A and the CPU. This way, the whole channel can be tested.
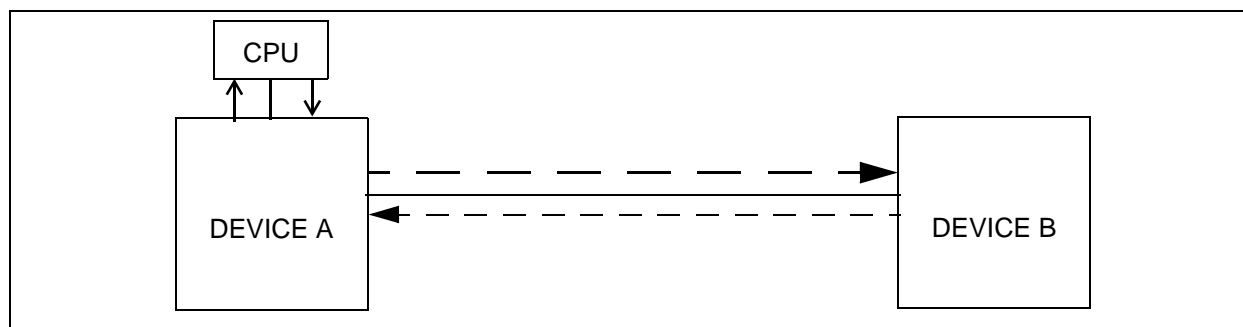
Figure 13 - Remote Loopback Test

10.0 Clocks

10.1 Clock Requirements

10.1.1 System Clock (SCLK) Speed Requirement

SCLK is the primary clock for the ZL50411 device. The speed requirement is based on the system configuration. Below is a table for a few configuration.

Configuration	Minimum SCLK speed required
6-9 ports 10/100 M	50 MHz
1-5 ports 10/100 M	25 Mhz

Table 11 - SCLK Speed Requirements

Zarlink Semiconductor Inc.

10.1.2   RMAC Reference Clock (M_CLK) Speed Requirement

M_CLK is a 50 MHz clock used for the RMAC ports (ports 0-7) and CPU port (port 8).

If none of the RMAC ports are configured in RMII mode or Reverse MII mode, a different clock frequency can be applied to M_CLK, as long as it's less than 50 MHz. In this case, register USD must be set to provide an internal 1usec timing.

10.1.3   MMAC Reference Clock (REF_CLK) Speed Requirement

REF_CLK is a reference clock required for the MMAC port (port 9).

If the device is in a 9 port 10/100 configuration only, REF_CLK can be a lower frequency clock and can be connected to M_CLK to reduce the number of clock sources.

If port 9 is not being used, REF_CLK can be left unconnected.

10.1.4   JTAG Test Clock (TCK) Speed Requirements

TCK is a clock used for the JTAG port. The frequency on this clock can vary. Refer to "JTAG (IEEE 1149.1-2001)" on page 141 for the frequency range.

10.2   Clock Generation

10.2.1   MDC

MDC is used for the MII Management Interface and clocks data on MDIO. It is generated by the device from M_CLK and is equal to 500 kHz (M_CLK/100). If a different speed clock other than 50 MHz is used on M_CLK, the USD register must be programmed to reset MDC.

10.2.2   SCL

SCL is used for the I2C interface and clocks data on SDA. It is generated by the device from M_CLK and is equal to 50kHz (M_CLK/1000). If a different speed clock other than 50 MHz is used on M_CLK, the USD register must be programmed to reset SCL.

10.2.3   Ethernet Interface Clocks

If the RMAC ports are configured in Reverse MII mode, TXCLK and RXCLK are generated from M_CLK and are equal to M_CLK/2 for 100 M mode or M_CLK/20 for 10M mode. M_CLK needs to be a 50 MHz clock in this mode.

If the RMAC ports are configured in Reverse GPSI mode, TXCLK and RXCLK are generated from M_CLK and are equal to M_CLK/2 for 10 M mode. M_CLK needs to be a 20 MHz clock in this mode and USD must be programmed accordingly.

For the CPU port in serial+MII mode, TXCLK and RXCLK are generated from M_CLK and are equal to M_CLK/2 for 100 M mode or M_CLK/20 for 10 M mode. M_CLK needs to be a 50 MHz clock in this mode.

If the MMAC port is configured in Reverse MII mode, RXCLK is generated from REF_CLK and is equal to REF_CLK/2 for 100 M mode (no support for 10M Reverse MII mode). REF_CLK needs to be a 50 MHz clock in this mode.

11.0   Hardware Statistics Counters

11.1   Hardware Statistics Counters List

ZL50411 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 filtering

Zarlink Semiconductor Inc.

counter (detected by queue manager). The following is the wrapped signal sent to the CPU through the command block.

See Processor Interface application note, ZLAN-26, for more information.

11.2   IEEE 802.3 HUB Management (RFC 1516)

11.2.1   Event Counters

11.2.1.1   PortReadableFrames

Counts the number of frames of valid frame length that have been received on this port.

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

No FCS (i.e. checksum) error

No collisions

Use ZL50411 counter "Total Frames Received", C[8].

Note: max. frame size may be BUF_LIMIT, if enabled on this port.

11.2.1.2   PortReadableOctets

Counts the number of bytes (i.e. octets) contained in valid frames that have been received on this port.

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

No FCS error

No collisions

Use ZL50411 counter "Total Bytes Received", C[7].

Note: max. frame size may be BUF_LIMIT, if enabled on this port.

Zarlink Semiconductor Inc.

11.2.1.3   PortFCSErrors

Counts number of valid frames received with bad FCS.

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

No framing error

No collisions

Use ZL50411 counter "CRC", C[23].

Note: if BUF_LIMIT is enabled on this port, this counter still uses the above max. frame size definition. If a frame >1518B but $\leq$BUF_LIMIT has a CRC error, it will be countered as an Oversized frame.

11.2.1.4   PortAlignmentErrors

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

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

Framing error

FCS error

No collisions

Use ZL50411 counter "Alignment Error", C[21].

Note: when this counter is incremented, the ZL50411 also increments the "CRC" counter, C[23].

11.2.1.5   PortFrameTooLongs

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

	Min. Frame Size	Max. Frame Size
Frame size:	n/a	> 1518 bytes
		(> 1522 bytes, if VLAN tagged)

Framing error: don't care

FCS error: don't care

No collisions

Use ZL50411 counter "Oversize Frames", C[15].

Zarlink Semiconductor Inc.

Note: max. frame size may be BUF_LIMIT, if enabled on this port. Also, if a frame >1518B but ≤BUF_LIMIT has a CRC error, it will be countered as an Oversized frame.

11.2.1.6   PortShortEvents

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

	Min. Frame Size	Max. Frame Size
Frame size:	< 10 bytes	

Framing error: don't care

FCS error: don't care

No collisions

Use ZL50411 counter "Short Event", C[24].

11.2.1.7   PortRunts

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

	Min. Frame Size	Max. Frame Size
Frame size:	≥ 10 bytes,	< 64 bytes

Framing error: don't care

FCS error: don't care

No collisions

Use ZL50411 counter "Undersize Frames", C[22].

11.2.1.8   PortCollisions

Counts number of collision events.

	Min. Frame Size	Max. Frame Size
Frame size:		any size

Use ZL50411 counter "Collision", C[25].

11.2.1.9   PortLateEvents

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

	Min. Frame Size	Max. Frame Size
Frame size:		any size

Use ZL50411 counter "Port Late Collision", C[29].

Zarlink Semiconductor Inc.

Note: when this counter is incremented, the ZL50411 also increments the "Collision" counter, C[25].

11.2.1.10   PortVeryLongEvents

Counts number of frames received with size larger than Jabber Lockup Protection Timer (TW3 = 5 ms or 50 000 bit times – 20% + 50%).

	Min. Frame Size	Max. Frame Size
Frame size:	> TW3	

No ZL50411 counter.

Note: the ZL50411 defines a jabber frame as any frame that exceeds 6400 bytes, however, the maximum buffer size supported by the device is 4 KBytes. Thus, any frame > 4KB would be counted as an Oversized Frame, C[15].

11.2.1.11   PortDataRateMisatches

For repeaters or HUB application only. No ZL50411 counter.

11.2.1.12   PortAutoPartitions

For repeaters or HUB application only. No ZL50411 counter.

11.2.1.13   PortTotalErrors

Sum of the following errors:

- PortFCSErrors
- PortAlignmentErrors
- PortFrameTooLong
- PortShortEvents
- PortLateEvents
- PortVeryLongEvents
- PortDataRateMisatches

11.3   IEEE 802.1 Bridge Management (RFC 1286)

11.3.1   Event Counters

11.3.1.1   PortDelayExceededDiscards

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

Note: Not applicable for the ZL50411.

11.3.1.2   PortMtuExceededDiscards

Counts number of frames discarded due to excessive size.

11.3.1.3   PortInFrames

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.

Zarlink Semiconductor Inc.

11.3.1.4 PortOutFrames

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.

11.3.1.5 PortInDiscards

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

11.4 RMON – Ethernet Statistic Group (RFC 1757)

11.4.1 Event Counters

11.4.1.1 DropEvents

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.

Use ZL50411 counter "Drop", C[26].

11.4.1.2 Octets

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

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

No framing error

FCS error: don't care

No collisions

Use ZL50411 counter "Bytes Received", C[5].

Note: max. frame size may be BUF_LIMIT, if enabled on this port.

11.4.1.3 Pkts

Counts the number of frames received, good or bad.

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)

No framing error

FCS error: don't care

No collisions

58

Zarlink Semiconductor Inc.

Use ZL50411 counter "Frames Received", C[6].

Note: max. frame size may be BUF_LIMIT, if enabled on this port.

11.4.1.4 BroadcastPkts

Counts the number of good frames received and forwarded with broadcast address. Does not include non-broadcast multicast frames.

Use ZL50411 counter "Broadcast Frames Received", C[11].

11.4.1.5 MulticastPkts

Counts the number of good frames received and forwarded with multicast address. Does not include broadcast frames.

Use ZL50411 counter "Multicast Frames Received", C[10].

Note: this counter includes broadcast frames received.

11.4.1.6 CRCAlignErrors

Counts number of frames received with FCS or alignment errors.

	Min. Frame Size	Max. Frame Size
Frame size:	$\geq$ 64 bytes,	$\leq$ 1518 bytes
		($\leq$ 1522 bytes, if VLAN tagged)
Framing error and/or FCS error		
No collisions		

Use ZL50411 counters "CRC", C[23], and "Alignment Error", C[21].

Note: when the "Alignment Error" counter, C[21], is incremented, the ZL50411 also increments the "CRC" counter, C[23], thus, the CRC counter can be used for this statistic.

11.4.1.7 UndersizePkts

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

	Min. Frame Size	Max. Frame Size
Frame size:	n/a	< 64 bytes
Framing error: don't care		
FCS error: don't care		
No collisions		

Use ZL50411 counter "Undersize Frames", C[22].

Zarlink Semiconductor Inc.

11.4.1.8 OversizePkts

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

	Min. Frame Size	Max. Frame Size
Frame size:	n/a	> 1518 bytes
		(> 1522 bytes, if VLAN tagged)

Framing error: don't care

FCS error: don't care

No collisions

Use ZL50411 counter "Oversize Frames", C[15].

Note: max. frame size may be BUF_LIMIT, if enabled on this port.

11.4.1.9 Fragments

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

	Min. Frame Size	Max. Frame Size
Frame size:	n/a	< 64 bytes

Framing error and/or FCS error

No collisions

Use ZL50411 counter "Fragments", C[20].

11.4.1.10 Jabbers

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

Note that this definition of jabber is different than the definition in IEEE 802.3 (this standard defines jabber as the condition where any packet exceeds Jabber Lockup Protection Timer (TW3)).

	Min. Frame Size	Max. Frame Size
Frame size:	n/a	> 1518 bytes
		(> 1522 bytes, if VLAN tagged)

Framing error: don't care

FCS error

No collisions

No ZL50411 counter.

Note: the ZL50411 defines a jabber frame as any frame that exceeds 6400 bytes, however, the maximum buffer size supported by the device is 4 KBytes. Thus, any frame > 4 KB would be counted as an Oversized Frame, C[15].

Zarlink Semiconductor Inc.

11.4.1.11   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.

	Min. Frame Size	Max. Frame Size
Frame size:		any size

Use ZL50411 counter "Collision", C[25].

11.4.1.12   Packet Count for Different Size Groups

Six different size groups – one counter for each. Counts both good and bad packets.

Pkts64Octets   for any packet with size = 64 bytes.

Use ZL50411 counter "Frames with Length of 64 Bytes", C[12].

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

Use ZL50411 counter "Frames with Length Between 65-127 Bytes", C[14].

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

Use ZL50411 counter "Frames with Length Between 128-255 Bytes", C[16].

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

Use ZL50411 counter "Frames with Length Between 256-511 Bytes", C[17].

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

Use ZL50411 counter "Frames with Length Between 512-1023 Bytes", C[18].

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

Use ZL50411 counter "Frames with Length Between 1024-1518 Bytes", C[19].

Note: the ZL50411 will include frames up to max. frame size.

Note: the ZL50411 doesn't include bad frames in the above counters.

11.5   Miscellaneous Counters

In addition to the statistics groups defined in previous sections, the ZL50411 has other statistics counters for its own purposes.

1. Flow control – one counting the number of flow control frames received, C[9], and another counting the number of flow control frames sent, C[3].
2. Frames sent – one for unicast frames sent, C[1], and one for non-unicast frames sent, C[4]. A broadcast or multicast frame qualifies as non-unicast.
3. A counter called "frame send fail", C[2]. This keeps track of FIFO under-runs, late collisions, and collisions that have occurred 16 times.
4. A counter called "filtering", C[27]. This keeps track of all frames dropped due to queue congestion (i.e. egress WRED).

Zarlink Semiconductor Inc.

12.0   Register Definition

12.1   ZL50411 Register Description

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
0. ETHERNET Port Control Registers (Substitute [n] with Port number (0..9))						
ECR1Pn	Port Control Register 1 for Port n	0000+2n	R/W	000+n	C0	
ECR2Pn	Port Control Register 2 for Port n	0001+2n	R/W	00A+n	00	
ECR3Pn	Port Control Register 3 for Port n	0080+2n	R/W	014+n	00	
ECR4Pn	Port Control Register 4 for Port n	0081+2n	R/W	01E+n	18	
BUF_LIMIT	Frame Buffer Limit	0036	R/W	NA	40	
FCC	Flow Control Grant Period	0037	R/W	NA	03	
1. VLAN Control Registers (Substitute [n] with Port number (0..9))						
AVTCL	VLAN Type Code Register Low	0100	R/W	028	00	
AVTCH	VLAN Type Code Register High	0101	R/W	029	81	
PVMAPn_0	Port n Configuration Register 0	0102+4n	R/W	02A+n	FF	
PVMAPn_1	Port n Configuration Register 1	0103+4n	R/W	034+n	FF	
PVMAPn_3	Port n Configuration Register 3	0105+4n	R/W	03E+n	00	
PVMODE	VLAN Operating Mode	0170	R/W	048	00	
PVLAN_Pn	Private VLAN Edge (protected ports) Port n Egress Portmap	0220+n	R/W	NA	00	
2. TRUNK Control Registers (Substitute [n] with Trunk Group number (0..7))						
TRUNKn	Trunk Group n	0200+n	R/W	NA	00	
TRUNKn_HASH10	Trunk Group n Hash 10 Destination Port	0208+4n	R/W	NA	00	
TRUNKn_HASH32	Trunk Group n Hash 32 Destination Port	0209+4n	R/W	NA	00	
TRUNKn_HASH54	Trunk Group n Hash 54 Destination Port	020A+4n	R/W	NA	00	
TRUNKn_HASH76	Trunk Group n Hash 76 Destination Port	020B+4n	R/W	NA	00	

Table 12 - Register Description

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
MULTICAST_HASHn-0	Multicast hash result n mask byte 0	0228+2n	R/W	NA	FF	
MULTICAST_HASHn-1	Multicast hash result n mask byte 1	0229+2n	R/W	NA	FF	
3. CPU Port Configuration						
MAC0	CPU MAC Address byte 0	0300	R/W	NA	00	
MAC1	CPU MAC Address byte 1	0301	R/W	NA	00	
MAC2	CPU MAC Address byte 2	0302	R/W	NA	00	
MAC3	CPU MAC Address byte 3	0303	R/W	NA	00	
MAC4	CPU MAC Address byte 4	0304	R/W	NA	00	
MAC5	CPU MAC Address byte 5	0305	R/W	NA	00	
INT_MASK0	Interrupt Mask 0	0306	R/W	NA	00	
INTP_MASKn	Interrupt Mask for MAC Port 2n, 2n+1	0310+n	R/W	NA	00	(n=0..4)
RQS	Receive Queue Select	0323	R/W	NA	00	
RQSS	Receive Queue Status	0324	RO	NA	NA	
MAC01	Increment MAC port 0,1 address	0325	R/W	NA	00	
MAC23	Increment MAC port 2,3 address	0326	R/W	NA	00	
MAC45	Increment MAC port 4,5 address	0327	R/W	NA	00	
MAC67	Increment MAC port 6,7 address	0328	R/W	NA	00	
MAC9	Port 9 MAC address byte 5	0329	R/W	NA	00	
CPUQINS[6:0]		0330-0336	R/W	NA	00	
CPUQINSRPT		0337	RO	NA	NA	
CPUGRNHDL[1:0]		0338-0339	RO	NA	NA	
CPURLSINFO[4:0]		033A-033E	R/W	NA	00	
CPUGRNCTR		033F	R/W	NA	00	
4. Search Engine Configurations						
AGETIME_LOW	MAC Address Aging Time Low	0400	R/W	049	5C	
AGETIME_HIGH	MAC Address Aging Time High	0401	R/W	04A	00	
SE_OPMODE	Search Engine Operating Mode	0403	R/W	NA	00	

Table 12 - Register Description (continued)

63

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
5. Global QOS Control						
QOSC	QOS Control	0500	R/W	04B	00	
PCC	Packet Congestion Control	0510	R/W	068	006	
MCC	Multicast Congestion Control	0511	R/W	069	06	
MCCTH	Multicast Congestion Threshold	0512	R/W	NA	03	
RDRC0	WRED Drop Rate Control 0	0513	R/W	090	00	
RDRC1	WRED Drop Rate Control 1	0514	R/W	091	00	
RDRC2	WRED Drop Rate Control 2	0515	R/W	NA	00	
SFCB	Share FCB Size	0518	R/W	074	00	
C1RS	Class 1 Reserve Size	0519	R/W	075	00	
C2RS	Class 2 Reserve Size	051A	R/W	076	00	
C3RS	Class 3 Reserve Size	051B	R/W	077	00	
AVPML	VLAN Priority Map Low	0530	R/W	056	00	
AVPMM	VLAN Priority Map Middle	0531	R/W	057	00	
AVPMH	VLAN Priority Map High	0532	R/W	058	00	
AVDM	VLAN Discard Map	0533	R/W	05C	00	
TOSPML	TOS Priority Map Low	0540	R/W	059	00	
TOSPMM	TOS Priority Map Middle	0541	R/W	05A	00	
TOSPMH	TOS Priority Map High	0542	R/W	05B	00	
TOSDML	TOS Discard Map	0543	R/W	05D	00	
USER_PROTOCOL_n	User Define Protocol n	0550+n	R/W	0B3+n	00	(n=0..7)
USER_PROTOCOL_FORCE_DISCARD	User Define Protocol 0 To 7 Force Discard Enable	0558	R/W	0BB	00	
WLPP10	Well Known Logic Port 0 and 1 Priority	0560	R/W	0A8	00	
WLPP32	Well Known Logic Port 2 and 3 Priority	0561	R/W	0A9	00	
WLPP54	Well Known Logic Port 4 and 5 Priority	0562	R/W	0AA	00	
WLPP76	Well Known Logic Port 6 and 7 Priority	0563	R/W	0AB	00	
WLPE	Well Known Logic Port 0 To 7 Enable	0564	R/W	0AC	00	
WLPFD	Well Known Logic Port 0 To 7 Force Discard Enable	0565	R/W	0AD	00	

Table 12 - Register Description (continued)

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
USER_PORTn_LOW	User Define Logical Port n Low	0570+2n	R/W	092+n	00	(n=0..7)
USER_PORTn_HIGH	User Define Logical Port n High	0571+2n	R/W	09A+n	00	
USER_PORT1:0_ PRIORITY	User Define Logic Port 0 and 1 Priority	0590	R/W	0A2	00	
USER_PORT3:2_ PRIORITY	User Define Logic Port 2 and 3 Priority	0591	R/W	0A3	00	
USER_PORT5:4_ PRIORITY	User Define Logic Port 4 and 5 Priority	0592	R/W	0A4	00	
USER_PORT7:6_ PRI ORITY	User Define Logic Port 6 and 7 Priority	0593	R/W	0A5	00	
USER_PORT_ ENABLE[7:0]	User Define Logic Port 0 To 7 Enable	0594	R/W	0A6	00	
USER_PORT_ FORCE_DISCARD[7:0]	User Define Logic Port 0 To 7 Force Discard Enable	0595	R/W	0A7	00	
RLOWL	User Define Range Low Bit [7:0]	05A0	R/W	0AE	00	
RLOWH	User Define Range Low Bit [15:8]	05A1	R/W	0AF	00	
RHIGHL	User Define Range High Bit [7:0]	05A2	R/W	0B0	00	
RHIGHH	User Define Range High Bit [15:8]	05A3	R/W	0B1	00	
RPRIORITY	User Define Range Priority	05A4	R/W	0B2	00	
6. MISC Configuration Register						
MII_OP0	MII Register Option 0	0600	R/W	0BC	00	
MII_OP1	MII Register Option 1	0601	R/W	0BD	00	
FEN	Feature Registers	0602	R/W	0BE	10	
MIIC0	MII Command Register 0	0603	R/W	NA	00	
MIIC1	MII Command Register 1	0604	R/W	NA	00	
MIIC2	MII Command Register 2	0605	R/W	NA	00	
MIIC3	MII Command Register 3	0606	R/W	NA	00	
MIID0	MII Data Register 0	0607	RO	NA	NA	
MIID1	MII Data Register 1	0608	RO	NA	NA	
USD	One micro second divider	0609	R/W	NA	00	
DEVICE	Device id and test	060A	R/W	NA	02	

Table 12 - Register Description (continued)

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
SUM	EEPROM Checksum Register	060B	R/W	0FF	00	
LHBTimer	Link heart beat time out timer	0610	R/W	NA	00	
LHBReg0	LHB control field value[7:0]	0611	R/W	NA	00	
LHBReg1	LHB control field value [15:8]	0612	R/W	NA	00	
fMACCReg0	Forced MAC control field value [7:0]	0613	R/W	NA	00	
fMACCReg1	Forced MAC control field value [15:8]	0614	R/W	NA	00	
FCB_BASE_ADDR0	FCB Base Address Register 0	0620	R/W	0BF	00	
FCB_BASE_ADDR1	FCB Base Address Register 1	0621	R/W	0C0	60	
FCB_BASE_ADDR2	FCB Base Address Register 2	0622	R/W	0C1	00	
7. Port Mirroring Controls						
MIRROR_DEST_MAC0	Mirror Destination MAC Address 0	0700	R/W	NA	00	
MIRROR_DEST_MAC1	Mirror Destination MAC Address 1	0701	R/W	NA	00	
MIRROR_DEST_MAC2	Mirror Destination MAC Address 2	0702	R/W	NA	00	
MIRROR_DEST_MAC3	Mirror Destination MAC Address 3	0703	R/W	NA	00	
MIRROR_DEST_MAC4	Mirror Destination MAC Address 4	0704	R/W	NA	00	
MIRROR_DEST_MAC5	Mirror Destination MAC Address 5	0705	R/W	NA	00	
MIRROR_SRC_MAC0	Mirror Source MAC Address 0	0706	R/W	NA	00	
MIRROR_SRC_MAC1	Mirror Source MAC Address 1	0707	R/W	NA	00	
MIRROR_SRC_MAC2	Mirror Source MAC Address 2	0708	R/W	NA	00	
MIRROR_SRC_MAC3	Mirror Source MAC Address 3	0709	R/W	NA	00	
MIRROR_SRC_MAC4	Mirror Source MAC Address 4	070A	R/W	NA	00	

Table 12 - Register Description (continued)

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
MIRROR_SRC_MAC5	Mirror Source MAC Address 5	070B	R/W	NA	00	
MIRROR_CONTROL	Port Mirror Control Register	070C	R/W	NA	00	
RMII_MIRROR0	RMII Mirror 0	0710	R/W	NA	00	
RMII_MIRROR1	RMII Mirror 1	0711	R/W	NA	00	
8. Per Port QOS Control						
FCRn	Flooding Control Register n	0800+n	R/W	04C+n	00	(n=0..9)
BMRCn	Broadcast/Multicast Rate Control n	0820+n	R/W	05E+n	00	
PR100_n	Port Reservation for RMAC Ports (n=0..7)	0840+n	R/W	06A+n	06	'd1536/16='d96, 'd96>>4='h6
PR100_CPU	Port Reservation for CPU Port	0848	R/W	073	06	'd96
PRM	Port Reservation for MMAC Port	0849	R/W	072	24	'd96x6='d576, 'd576>>4='h24
PTH100_n	Port Threshold for RMAC Ports (n=0..7)	0860+n	R/W	0C2+n	03	½
PTH100_CPU	Port Threshold for CPU Port	0868	R/W	0CB	03	½
PTHM	Port Threshold for MMAC Port	0869	R/W	0CA	12	½
QOSCn	QOS Control n	0880+n	R/W	078-08F NA	00	(n=0..39)
E. System Diagnostic						
DTSRL	Test Register Low	0E00	R/W	NA	00	
DTSRM	Test Register Medium	0E01	R/W	NA	01	
TESTOUT0	Testmux Output [7:0]	0E02	R/O	NA	NA	
TESTOUT1	Testmux Output [15:8]	0E03	R/O	NA	NA	
MASK0	MASK Timeout 0	0E10	R/W	0F6	00	
MASK1	MASK Timeout 1	0E11	R/W	0F7	00	
MASK2	MASK Timeout 2	0E12	R/W	0F8	00	
MASK3	MASK Timeout 3	0E13	R/W	0F9	00	
MASK4	MASK Timeout 4	0E14	R/W	0FA	00	

Table 12 - Register Description (continued)

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
BOOTSTRAP[2:0]	BOOTSTRAP Read Back	0E80-0E82	RO	NA	NA	
PRTFSMSTn	Ethernet Port n Status Read Back	0E90+n	RO	NA	NA	(n=0..9)
PRTQOSSTn	RMAC Port n QOS and Queue Status	0EA0+n	RO	NA	NA	(n=0..7)
PRTQOSST8A	CPU Port QOS and Queue Status A	0EA8	RO	NA	NA	
PRTQOSST8B	CPU Port QOS and Queue Status B	0EA9	RO	NA	NA	
PRTQOSST9A	MMAC Port QOS and Queue Status A	0EAA	RO	NA	NA	
PRTQUSST9B	MMAC Port QOS and Queue Status B	0EAB	RO	NA	NA	
CLASSQOSST	Class Buffer Status	0EAC	RO	NA	NA	
PRTINTCTR	Buffer Interrupt Status	0EAD	R/W	NA	00	
QMCTRLn	Ports Queue Control Status	0EB0+n	R/W	NA	00	(n=0..9)
QCTRL	Ports Queue Control	0EBA	R/W	NA	00	
BMBISTR0	Memory bist result	0EBB	R/O	NA	NA	
BMBISTR1	Memory bist result	0EBC	R/O	NA	NA	
BMControl	Memory control	0EBD	R/W	NA	0F	
BUFF_RST	Buffer Reset Pool	0EC0	R/W	NA	00	
FCBHEADPTR0	FCB Head Pointer [7:0]	0EC1	R/W	NA	00	
FCB_HEAD_PTR1	FCB Head Pointer [15:8]	0EC2	R/W	NA	00	
FCB_TAIL_PTR0	FCB Tail Pointer [7:0]	0EC3	R/W	NA	00	
FCB_TAIL_PTR1	FCB Tail Pointer [15:8]	0EC4	R/W	NA	00	
FCB_NUM0	FCB Number [7:0]	0EC5	R/W	NA	00	
FCB_NUM1	FCB Init Start and FCB Number [14:8]	0EC6	R/W	NA	06	
BM_RLSFF_CTRL	Read control register	0EC7	R/W	NA	00	
BM_RLSFF_INFO0	Bm_rlsfifo_info[7:0]	0EC8	RO	NA	NA	
BM_RLSFF_INFO1	Bm_rlsfifo_info[15:8]	0EC9	RO	NA	NA	
BM_RLSFF_INFO2	Bm_rlsfifo_info[23:16]	0ECA	RO	NA	NA	
BM_RLSFF_INFO3	Bm_rlsfifo_info[31:24]	0ECB	RO	NA	NA	
BM_RLSFF_INFO4	Bm_rlsfifo_info[39:32]	0ECC	RO	NA	NA	
BM_RLSFF_INFO5	Fifo_cnt[2:0],Bm_rlsfifo_info[44:40]	0ECD	RO	NA	NA	

Table 12 - Register Description (continued)

Zarlink Semiconductor Inc.

Register	Description	CPU Addr (Hex)	R/W	I²C Addr (Hex)	Default (Hex)	Notes
F. System Control						
GCR	Global Control Register	0F00	R/W	NA	00	
DCR	Device Control Register	0F01	RO	NA	NA	
DCR1	Device Control Register 1	0F02	RO	NA	NA	
DPST	Device Port Status Register	0F03	R/W	NA	00	
DTST	Data read back register	0F04	RO	NA	NA	
DA	DA Register	0FFF	RO	NA	DA	

Table 12 - Register Description (continued)

12.2   Directly Accessed Registers

INDEX_REG0		Width	Access	Address
Used to write the address of the indirect register to be accessed.		8/16-bit	W	0
		Default: 00		

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

Register Table 1 - 0, INDEX_REG0

INDEX_REG1		Width	Access	Address
Used to write the address of the indirect register to be accessed.		8-bit	W	1
8-bit CPU Interface Only		**Default:** 00		

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

Register Table 2 - 1, INDEX_REG1

Zarlink Semiconductor Inc.

DATA_REG		Width	Access	Address
Reflects the value of the indirect access register set by INDEX_REG		8-bit	R/W	2
		Default: 00		

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

Register Table 3 - 2, DATA_REG

CPU_FRAME_REG		Width	Access	Address
Used to transmit/receive Ethernet frames via an 8-byte FIFO to/from the CPU MAC.		8/16-bit	R/W	3
This register (FIFO) is disabled in SSI+MII CPU interface mode.		**Default:** 00		

Bit #	Name	Type	Description
16-bit or serial CPU Interface			
[15:0]	CPU_FRAME	W	Send Ethernet frame to CPU MAC. Bits [7:0] is even byte, [15:8] is odd byte. Data sequence specified in Processor Interface application note, ZLAN-26.
		R	Receive Ethernet frame from CPU MAC. Bits [7:0] is even byte, [15:8] is odd byte. Data sequence specified in Processor Interface application note, ZLAN-26.
8-bit CPU Interface			
[7:0]	CPU_FRAME	W	Send Ethernet frame to CPU MAC Data sequence specified in Processor Interface application note, ZLAN-26.
		R	Receive Ethernet frame from CPU MAC Data sequence specified in Processor Interface application note, ZLAN-26.

Register Table 4 - 3, CPU_FRAME_REG

Zarlink Semiconductor Inc.

CMD_STATUS_REG	Width	Access	Address
CPU interface command status	8-bit	R/W	4
This register is not applicable in SSI+MII, MII-only and Remote/No CPU interface modes.	**Default:** 00		

Bit #	Name	Type	Description
[0]	RXBUF_DONE	W	Set Control Frame Receive buffer ready, after CPU writes a complete frame into the buffer. This bit is self-cleared.
	RXBUF_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]	TXBUF1_DONE	W	Set Control Frame Transmit buffer1 ready, after CPU reads out a complete frame from the buffer. This bit is self-cleared.
	TXBUF1_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]	TXBUF2_DONE	W	Set Control Frame Transmit buffer2 ready, after CPU reads out a complete frame from the buffer. This bit is self-cleared.
	TXBUF2_RDY	R	Control Frame transmit buffer2 ready for CPU to read 1 – CPU can read a new control command 1 0 – CPU has to wait until this bit is 1 to read a new control command
[3]	TXFIFO_RXDONE	W	Set this bit to indicate CPU received a whole frame (transmit FIFO frame receive done), and flushed the rest of frame fragment, If occurs. This bit will be self-cleared.
	TXFIFO_RDY	R	Transmit FIFO has data for CPU to read
[4]	RXFIFO_EOF	W	Set this bit to indicate that the following Write to the Receive FIFO is the last one. This bit will be self-cleared.
	RXFIFO_SPOK	R	Receive FIFO has space for incoming CPU frame
[5]	RXFIFO_REALIGN	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'.
	TXFIFO_EOF	R	Transmit FIFO End Of Frame
[7:6]	RSVD	R/W	Reserved

Register Table 5 - 4, CMD_STATUS_REG

Zarlink Semiconductor Inc.

INT_REG			Width	Access	Address
Interrupt sources			8-bit	R	5
			Default: N/A		

Bit #	Name	Type	Description
[0]	INT_CPU_FRAME	R	Ethernet frame interrupt. Ethernet Frame receive buffer has data for processor to read
[1]	INT_CTL_BUF1	R	Control frame 1 interrupt. Control Frame receive buffer 1 has data for processor to read
[2]	INT_CTL_BUF2	R	Control frame 2 interrupt. Control Frame receive buffer 2 has data for processor to read
[6:3]	RSVD	R	Reserved
[7]	INT_CHIP_RESET	R	Device Timeout Detected interrupt Note: This bit is not self-cleared. After reading, the CPU has to clear the bit writing 0 to it.

Register Table 6 - 5, INT_REG

CTL_FRAME_BUF1			Width	Access	Address
Used to transmit/receive control frames			8/16-bit	R/W	6
			Default: 00		

Bit #	Name	Type	Description
16-bit or serial CPU Interface			
[15:0]	CTL_BUF1	W	Send control frame to command engine. Bits [7:0] is even byte, [15:8] is odd byte. Control frame format specified in Processor Interface application note, ZLAN-26.
		R	Receive control frame from command engine. Bits [7:0] is even byte, [15:8] is odd byte. Control frame format specified in Processor Interface application note, ZLAN-26.

Register Table 7 - 6, CTL_FRAME_BUF1

Zarlink Semiconductor Inc.

8-bit CPU Interface			
[7:0]	CTL_BUF2	W	Send control frame to command engine Control frame format specified in Processor Interface application note, ZLAN-26.
		R	Receive control frame from command engine Control frame format specified in Processor Interface application note, ZLAN-26.

Register Table 7 - 6, CTL_FRAME_BUF1 (continued)

CTL_FRAME_BUF2	Width	Access	Address
Used to receive control frames	8/16-bit	R	7
	Default: 00		

Bit #	Name	Type	Description
16-bit or serial CPU Interface			
[15:0]	CTL_BUF2	R	Receive control frame from command engine. Bits [7:0] is even byte, [15:8] is odd byte. Control frame format specified in Processor Interface application note, ZLAN-26.
8-bit CPU Interface			
[7:0]	CTL_BUF2	R	Receive control frame from command engine Control frame format specified in Processor Interface application note, ZLAN-26.

Register Table 8 - 7, CTL_FRAME_BUF2

12.3   Indirectly Accessed Registers

12.3.1   (Group 0 Address) MAC Ports Group

12.3.1.1   ECR1Pn: Port n Control Register

I²C Address: h000+n; CPU Address: h0000+2n (n = port number)

Accessed by CPU and I²C (R/W)

Port 0 – 7 & 9: (RMAC & MMAC Ports)

Bit [0]	Flow Control 0 - Enable (Default) 1 - Disable
Bit [1]	Duplex Mode 0 - Full Duplex (Default) 1 - Half Duplex - Only in 10/100 mode

Zarlink Semiconductor Inc.

Bit [2]	Speed 0 - 100 Mbps (Default) 1 - 10 Mbps

Bits [4:3]	00 - Enable Auto-Negotiation (Default)
	      This enables hardware state machine for auto-negotiation.
	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.
Bits [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)

Port 8: (CPU Port)

8/16-bit or Serial Only Modes	
Bit [5:0]	Reserved
Bits [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)
Serial + MII Mode	
Bit [0]	Flow Control 0 - Enable (Default) 1 - Disable
Bit [1]	Duplex Mode Must be 0 - Full Duplex (Default)
Bit [2]	Speed 0 - 100 Mbps (Default) 1 - 10 Mbps

Bit [3]	1 - MII Port Up The configuration in ECR1Pn[2:0] is used for (speed/duplex/flow control) setup. 0 - MII Port Down Note: Bit [4] must be '1'.
Bit [4]	Must be '1'.
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.
Bits [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

I²C Address: h00A+n; CPU Address: h0001+2n (n = port number)

Accessed by CPU and I²C (R/W)

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]:	Rate control timer select (RMAC ports only) 0: 10 microsecond refreshing time (Default) 1: 1 millisecond refreshing time
Bit [4]	Force VLAN Tag Out 0: Disable (Default) 1: Enable If this feature is enabled, a packet leaving the device will ALWAYS have a VLAN tag. Only applicable in tagged-based VLAN mode (PVMODE[0]='1'). In port-based VLAN mode (PVMODE[0]='0'), this bit must be 0.

Zarlink Semiconductor Inc.

Bit [5]	Do not change VLAN tag. This overrides PVMAPnn_3 bit [2]. If this bit is set, no tag will be replaced nor removed. 0: Disable (Default) 1: Enable

Bits [7:6]	Security Enable. The ZL50411 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). ZL50411 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 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
	
	It also checks for one of the following additional conditions:
	• If the port is configured to filter untagged frames and an untagged frame arrives, or
	• If the port is configured to filter tagged frames and a tagged frame arrives, or
	• If the packet has the source mac address on the source mac address filter list, or
	• If the packet has the destination mac address on the destination mac address filter list
	If any one of the conditions is met, the packet will be handled according to:
	   0X – Discard violating packets
	   1X – Forward violating packets to CPU for inspection

12.3.1.3   ECR3Pn: Port n Control Register

I²C Address: h014+n; CPU Address: h0080+2n (n = port number)

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable receiving short frame < 64B 0: Disable (Default) 1: Allow receiving short frame with correct CRC.
Bit [1]:	Enable receiving long frame > 1522 0: Disable (Default) 1: Allow receiving long frame that are <= BUF_LIMIT value
Bit [2]:	Enable pad frame to 64B when transmitted 0: Allow padding to 64B (Default) 1: Disable

Zarlink Semiconductor Inc.

Bit [3]:	Enable compress preamble 0: Send standard preamble (Default) 1: Only one byte preamble+SFD
Bits [6:4]	Number of bytes removed from the Inter-Frame Gap (IFG). (Default 0x0)
Bit [7]	Link Heart Beat Transmit (RMAC ports only) 0: Disable (Default) 1: Enable

12.3.1.4 ECR4Pn: Port n Control Register

I²C Address: h01E+n; CPU Address: h0081+2n (n = port number)

Accessed by CPU and I²C (R/W)

Port 0 – 7: (RMAC Ports)

Bit [1:0]:	Enable PHY Mode 00: MAC Mode (Default) 11: PHY Mode (Reverse MII/GPSI) Only valid in MII and GPSI interface modes. In PHY mode, Mn_TXCLK & Mn_RXCLK pins becomes outputs.
Bit [2]:	Internal loopback. 0: Disable (Default) 1: Enable In this mode, the packet is looped back in the MAC layer before going out of the chip. You must force linkup at full duplex as well. External loopback is another level of system diagnostic which involves the PHY device to loopback the packet.

Bits [4:3]:	Interface mode:
	  00 - GPSI mode
	  01 - MII mode
	  10 - Reserved
	  11 - RMII mode (Default)

Bit [5]:	Frame loopback. 0: Disable frame from sending back to its source port. (Default) 1: Allow frame to send back to its source port In a regular ethernet switch, a packet should never be receive and forwarded to the same port. Setting the bit allows it to happen. This is not the same as an ingress MAC loopback. The destination MAC address has to be stored (learned) in the MCT and associated with the originating source port. The frame loopback will only work for unicast packets.
Bit [6]:	Link Heart Beat Receive 0: Disable (Default). Also clears all MAC LHB status. 1: Enable
Bit [7]:	Soft reset. 0: Normal operation (Default) 1: Reset. Not self clearing.

Zarlink Semiconductor Inc.

Port 8: (CPU Port)

Bits [1:0]:	Reserved
Bit [2]:	Enable special write to 2 registers in a single write operation. 0: Disable (Default) 1: Enable Should be enabled only in serial mode and disabled in 8/16-bit mode.
Bits [4:3]:	Enable insertion of 2-byte CPU information in CPU frame packet in Serial + MII mode 00: No information is inserted 01: Insert 2-byte of CPU information 10: Reserved 11: Insert 6-byte of padding + 2-byte of CPU information (Default) In port-based VLAN mode, the CPU MII interface must be in "No information is inserted" mode (ECR4P8[4:3]='00'). In tagged-based VLAN mode, the CPU MII interface supports all three modes (0,2,8 bytes insertion).
Bit [5]:	Frame loopback. 0: Disable frame from sending back to its source port. (Default) 1: Allow frame to send back to its source port In a regular ethernet switch, a packet should never be receive and forwarded to the same port. Setting the bit allows it to happen. This is not the same as an ingress MAC loopback. The destination MAC address has to be stored (learned) in the MCT and associated with the originating source port. The frame loopback will only work for unicast packets.
Bit [6]:	Reserved
Bit [7]:	Soft reset. 0: Normal operation (Default) 1: Reset. Not self clearing.

Port 9: (MMAC Port)

Bits [1:0]:	Enable PHY Mode 00: MAC Mode (Default) 11: PHY Mode (Reverse MII) In PHY mode, M9_RXCLK pin becomes an output and M9_MTXCLK must be tied to M9_RXCLK externally.
Bit [2]:	Internal loopback. 0: Disable (Default) 1: Enable In this mode, the packet is looped back in the MAC layer before going out of the chip. You must force linkup at full duplex as well. External loopback is another level of system diagnostic which involves the PHY device to loopback the packet.
Bits [4:3]:	Interface mode: 11 - MII mode (Default)

78

Zarlink Semiconductor Inc.

Bit [5]:	Frame loopback. 0: Disable frame from sending back to its source port. (Default) 1: Allow frame to send back to its source port
	In a regular ethernet switch, a packet should never be receive and forwarded to the same port. Setting the bit allows it to happen.
	This is not the same as an ingress MAC loopback. The destination MAC address has to be stored (learned) in the MCT and associated with the originating source port. The frame loopback will only work for unicast packets.
Bit [6]:	Reserved
Bit [7]:	Soft reset. 0: Normal operation (Default) 1: Reset. Not self clearing.

12.3.1.5   BUF_LIMIT – Frame Buffer Limit

CPU Address: h0036

Accessed by CPU (R/W)

Bits [6:0]:	Frame Buffer Limit (max 4 KB). Multiple of 64 bytes (Default 0x40)
Bit [7]:	Reserved

12.3.1.6   FCC – Flow Control Grant Period

CPU Address: h0037

Accessed by CPU (R/W)

Bits [2:0]:	Flow Control Grant Period. (Default 0x3) Units are (FCC[2:0]+1)*4us
Bits [7:3]:	Reserved

12.3.2   (Group 1 Address) VLAN Group

12.3.2.1   AVTCL – VLAN Type Code Register Low

I²C Address: h028; CPU Address: h0100

Accessed by CPU and I²C (R/W)

Bits [7:0]:	VLANType_LOW: Lower 8 bits of the VLAN type code (Default 0x00)

Zarlink Semiconductor Inc.

12.3.2.2 AVTCH – VLAN Type Code Register High

I²C Address: h029; CPU Address: h0101

Accessed by CPU and I²C (R/W)

Bits [7:0]:	VLANType_HIGH: Upper 8 bits of the VLAN type code (Default is 0x81)

12.3.2.3 PVMAP00_0 – Port 0 Configuration Register 0

I²C Address: h02A, CPU Address: h0102

Accessed by CPU and I²C (R/W)

In Port Based VLAN Mode

Bits [7:0]:	VLAN Mask for port 0 (Default 0xFF)

This register indicates the legal egress ports. A "1" on bit 3 means that the packet can be sent to port 3. A "0" on bit 3 means that any packet destined to port 3 will be discarded. This register works with registers 1 to form a 10 bit mask to all egress ports.

In Tag based VLAN Mode

Bits [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 0 Configuration Register 1

I²C Address: h034, CPU Address: h0103

Accessed by CPU and I²C (R/W)

In Port based VLAN Mode

Bits [1:0]:	VLAN Mask for ports 9 to 8 (Default 0x3)
Bits [7:2]:	Reserved (Default 0x3F)

In Tag based VLAN Mode

Bits [3:0]:	PVID [11:8] (Default is 0xF)
Bit [4]:	Untrusted Port. 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 (Default) 0: Keep VLAN priority field

Zarlink Semiconductor Inc.

Bits [7:5]:	Untag Port Priority **(Default 0x7)**
	Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)

12.3.2.5   PVMAP00_3 – Port 0 Configuration Register 3

I²C Address: h3E, CPU Address: h0105

Accessed by CPU and I²C (R/W)

In Port Based VLAN Mode

Bits [2:0]:	Reserved
Bits [5:3]:	Default transmit priority. Used when Bit [7]=1 (Default 0) Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bit [6]:	Default drop precedence. Used when Bit [7]=1 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest)
Bit [7]:	Enable Fix Priority (Default 0) 0 - Disable. All frames are analysed. Transmit Priority and Drop Precedence are based on VLAN Tag, TOS or Logical Port. 1 - Enable. Transmit Priority and Drop Precedence 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.

Bits [5:3]:	Default transmit priority. Used when Bit [7]=1 (Default 0) Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bit [6]:	Default drop precedence. Used when Bit [7]=1 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest)
Bit [7]:	Enable Fix Priority (Default 0) 0 - Disable. All frames are analysed. Transmit Priority and Drop Precedence are based on VLAN Tag, TOS or Logical Port. 1 - Enable. Transmit Priority and Drop Precedence are based on values programmed in bit [6:3]

12.3.2.6   PVMAPnn_0,1,3 – Ports 1~9 Configuration Registers

PVMAP01_0,1,3 - I²C Address h02B, 035, 03F; CPU Address:h0106, 0107, 0109 (Port 1)

PVMAP02_0,1,3 - I²C Address h02C, 036, 040; CPU Address:h010A, 010B, 010D (Port 2)

PVMAP03_0,1,3 - I²C Address h02D, 037, 041; CPU Address:h010E, 010F, 0111 (Port 3)

PVMAP04_0,1,3 - I²C Address h02E, 038, 042; CPU Address:h0112, 0113, 0115 (Port 4)

PVMAP05_0,1,3 - I²C Address h02F, 039, 043; CPU Address:h0116, 0117, 0119 (Port 5)

PVMAP06_0,1,3 - I²C Address h030, 03A, 044; CPU Address:h011A, 011B, 011D (Port 6)

PVMAP07_0,1,3 - I²C Address h031, 03B, 045; CPU Address:h011E, 011F, 0121 (Port 7)

PVMAP08_0,1,3 - I²C Address h032, 03C, 046; CPU Address:h0122, 0123, 0125 (Port CPU)

PVMAP09_0,1,3 - I²C Address h033, 03D, 047; CPU Address:h0126, 0127, 0129 (Port MMAC)

12.3.2.7   PVMODE

I²C Address: h048, CPU Address: h0170

Accessed by CPU and I²C (R/W)

Bit [0]:	VLAN Mode 0: Port based VLAN Mode (Default) 1: Tag based VLAN Mode
Bit [1]:	Slow learning (Default = 0) Same function as SE_OPMODE 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 01-80-C2-00-00-01 to 0x01-80-C2-00-00-0F. 0: Drop all frames in this range (Default) 1: Disable dropping of frames in this range

Zarlink Semiconductor Inc.

Bit [3]:	Flooding control in secure mode 0: Enable - Learning disabled port will not receive any flooding packets (Default) 1: Disable

Bit [4]:	Support MAC address 0
	0: MAC address 0 is not learned. (Default)
	   This means packet with destination MAC address 0 is forwarded as unknown
	   destination. It is subjected to unicast to multicast rate control.
	1: MAC address 0 is learned.

Bit [5]:	Disable IEEE multicast control frame (01-80-C2-00-00-00 to 01-80-C2-00-00-0F) to CPU in managed mode. In unmanaged mode, frame is forwarded as multicast (except PAUSE frame). 0: Frame is forwarded to CPU (Default) 1: Frame is forwarded as multicast (except PAUSE frame)
Bit [6]:	IP Multicast Enable 0: Disable (default) 1: Enable In general, this bit is equal to ^FEN[4].
Bit [7]:	Enable logical port match in secure mode 0: Disable (Default) 1: Enable - When Well Known or User Define logical port force discard enabled, force any IP packet with logical port number matching logical port numbers to CPU.

12.3.2.8 PVLAN_Pn – Per-Port Private VLAN Edge (Protected Ports)

CPU Address: h0220+n (n = RMAC port number)

Accessed by CPU (R/W)

Only applicable if Private VLAN Edge feature is enable (FEN[1]='1').

NOTE: These registers overlap TRUNK6/7_HASHxy registers, thus, trunk groups 6 and 7 must be disabled (TRUNK6=TRUNK7=0x00) if the PVLAN feature is enabled.

Bits [7:0]	Port 9,7-0 bit map of protected port n (Default 0x00)
	
	Bits [7:0] represent ports 9,7-0, excluding the source port. For example:
	   For port 0, [7:0] represent ports 9,7..1
	   For port 1, [7:0] represent ports 9,7..2,0
	   For port 2, [7:0] represent ports 9,7..3,1..0

12.3.3 (Group 2 Address) Port Trunking Groups

Trunk Group – Up to eight RMAC ports can be selected for each trunk group.

83

12.3.3.1   TRUNKn– Trunk Group 0~7

CPU Address: h0200+n (n = trunk group)

Accessed by CPU (R/W)

Bit [7:0] Port 7-0 bit map of trunk n. (Default 0)

<div align="center">

B   B
i   i
t   t
7   0

TRUNK0

P   P
o   o
r   r
t   t
7   0

</div>
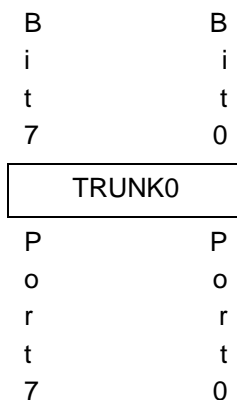
12.3.3.2   TRUNKn_HASH10 – Trunk group n hash result 1/0 destination port number

CPU Address: h0208+4n (n = trunk group)

Accessed by CPU (R/W)

Bits [3:0]	Hash result 0 destination port number (Default 0)
Bits [7:4]	Hash result 1 destination port number (Default 0)

12.3.3.3   TRUNKn_HASH32 – Trunk group n hash result 3/2 destination port number

CPU Address: h0209+4n (n = trunk group)

Accessed by CPU (R/W)

Bits [3:0]	Hash result 2 destination port number (Default 0)
Bits [7:4]	Hash result 3 destination port number (Default 0)

12.3.3.4   TRUNKn_HASH54 – Trunk group n hash result 5/4 destination port number

CPU Address: h020A+4n (n = trunk group)

Accessed by CPU (R/W)

Bits [3:0]	Hash result 4 destination port number (Default 0)
Bits [7:4]	Hash result 5 destination port number (Default 0)

Zarlink Semiconductor Inc.

12.3.3.5 TRUNKn_HASH76 – Trunk group n hash result 7/6 destination port number

CPU Address: h020B+4n (n = trunk group)

Accessed by CPU (R/W)

Bits [3:0]	Hash result 6 destination port number (Default 0)
Bits [7:4]	Hash result 7 destination port number (Default 0)

Multicast Hash Registers

Multicast Hash registers are used to distribute multicast traffic. 16 registers are used to form a 8-entry array; each entry has 10 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	HASH0-1	HASH0-0
Hash Value =1	HASH1-1	HASH1-0
Hash Value =2	HASH2-1	HASH2-0
Hash Value =3	HASH3-1	HASH3-0
Hash Value =4	HASH4-1	HASH4-0
Hash Value =5	HASH5-1	HASH5-0
Hash Value =6	HASH6-1	HASH6-0
Hash Value =7	HASH7-1	HASH7-0

Port 9 Port 8 Port 7   Port 0

12.3.3.6 MULTICAST_HASHn-0 – Multicast hash result 0~7 mask byte 0

CPU Address: h0228+2n (n = hash value)

Accessed by CPU (R/W)

Bits [7:0]:	Port 7-0 bit map for multicast hash. (Default 0xFF)

12.3.3.7 MULTICAST_HASHn-1 – Multicast hash result 0~7 mask byte 1

CPU Address: h0229+2n (n = hash value)

Accessed by CPU (R/W)

Bits [1:0]:	Port 9-8 bit map for multicast hash. (Default 0x3)

85

Zarlink Semiconductor Inc.

Bit [2]:	MULTICAST_HASH1-1 Unknown IP Multicast Filter Enable 1: Enable (default) 0: Disable If this feature is enabled, unknown IP multicast packets or unknown multicast packets can be either filtered or sent to the CPU, based on bits [13:12] in the per VLAN Port Association table. This feature is only applicable in tagged-based VLAN mode (PVMODE[0]='1'). In port-based VLAN mode (PVMODE[0]='0'), this bit is ignored. MULTICAST_HASH[7:2,0]-1 Reserved (Default 0x1)
Bits [5:3]:	Reserved (Default 0x7)
Bits [7:6]:	MULTICAST_HASH0-1 Hash Select. The hash algorithm selected is valid for all trunks 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 Port Number for hashing (Default) MULTICAST_HASH[7:1]-1 Reserved (Default 0x3)

12.3.4   (Group 3 Address) CPU Port Configuration Group

47   0   (MC bit)

MAC5	MAC4	MAC3	MAC2	MAC1	MAC0
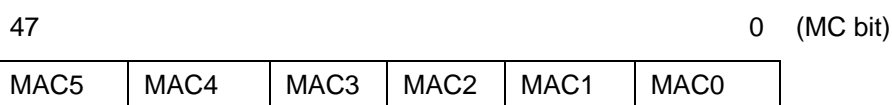
MAC5 to MAC0 registers form the CPU MAC address. When a packet with destination MAC address match MAC [5:0], the packet is forwarded to the CPU. The default MAC address is 00-00-00-00-00-00.

12.3.4.1   MAC0 – CPU MAC address byte 0

CPU Address: h0300

Accessed by CPU (R/W)

Bits [7:0]:	Byte 0 (bits [7:0]) of the CPU MAC address (Default 0)

12.3.4.2   MAC1 – CPU MAC address byte 1

CPU Address: h0301

Accessed by CPU (R/W)

Bits [7:0]:	Byte 1 (bits [15:8]) of the CPU MAC address (Default 0)

Zarlink Semiconductor Inc.

12.3.4.3   MAC2 – CPU MAC address byte 2

CPU Address: h0302

Accessed by CPU (R/W)

Bits [7:0]:	Byte 2 (bits [23:16]) of the CPU MAC address (Default 0)

12.3.4.4   MAC3 – CPU MAC address byte 3

CPU Address: h0303

Accessed by CPU (R/W)

Bits [7:0]:	Byte 3 (bits [31:24]) of the CPU MAC address (Default 0)

12.3.4.5   MAC4 – CPU MAC address byte 4

CPU Address: h0304

Accessed by CPU (R/W)

Bits [7:0]:	Byte 4 (bits [39:32]) of the CPU MAC address (Default 0)

12.3.4.6   MAC5 – CPU MAC address byte 5

CPU Address: h0305

Accessed by CPU (R/W)

Bits [7:0]:	Byte 5 (bits [47:40]) of the CPU MAC address (Default 0) Note: Bits [42:40] are set on a per port basis using MAC01, MAC23, MAC45, MAC67 registers. For port 9, this register is ignored and MAC9 is used for bits [47:40].

12.3.4.7   INT_MASK0 – Interrupt Mask

CPU Address: h0306

Accessed by CPU (R/W)

The CPU can dynamically mask the interrupt when it is busy and doesn't want to be interrupted. (Default 0x00)

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

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
Bits [6:3]:	Reserved
Bit [7]:	Device Timeout Detected interrupt

Zarlink Semiconductor Inc.

12.3.4.8   INTP_MASK0 – Interrupt Mask for MAC Port 0,1

CPU Address: h0310

Accessed by CPU (R/W)

The CPU can dynamically mask the interrupt when it is busy and doesn't want to be interrupted (Default 0x00)

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

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 [2]:	Port 0 module detect mask
Bit [3]:	Reserved
Bit [4]:	Port 1 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 [5]:	Port 1 link change mask
Bit [6]:	Port 1 module detect mask
Bit [7]	Reserved

12.3.4.9   INTP_MASKn – Interrupt Mask for MAC Ports 2~9 Registers

INTP_MASK1 - CPU Address: h0311 (Ports 2,3)

INTP_MASK2 - CPU Address: h0312 (Ports 4,5)

INTP_MASK3 - CPU Address: h0313 (Ports 6,7)

INTP_MASK4 - CPU Address: h0314 (Port CPU,MMAC)

12.3.4.10   RQS – Receive Queue Select

CPU Address: h0323

Accessed by CPU (RW)

Select which receive queue is being used by the CPU port.

Bit [0]:	Select Queue 0 0: Not selected (Default) 1: Selected
Bit [1]:	Select Queue 1
Bit [2]:	Select Queue 2
Bit [3]:	Select Queue 3
Bit [4]:	Select Multicast Queue 0
Bit [5]:	Select Multicast Queue 1

Bit [6]:	Select Multicast Queue 2
Bit [7]:	Select Multicast Queue 3

Note: Strict priority applies between different selected queues (UQ3>UQ2>UQ1>UQ0>MQ3>MQ2>MQ1>MQ0).

12.3.4.11   RQSS – Receive Queue Status

CPU Address: h0324

Accessed by CPU (RO)

CPU receive queue status

Bits [3:0]:	Unicast Queue 3 to 0 not empty 0: Empty 1: Not Empty
Bits [7:4]:	Multicast Queue 3 to 0 not empty

12.3.4.12   MAC01 – Increment MAC port 0,1 address

CPU Address: h0325

Accessed by CPU (RW)

Bits [2:0]:	Bits [42:40] of Port 0 CPU MAC address
Bit [3]:	Reserved
Bits [6:4]:	Bits [42:40] of Port 1 CPU MAC address
Bit [7]:	Reserved

MAC01, MAC23, MAC45, MAC67, and MAC9 registers are used with the MAC0~5 registers to form the CPU MAC address on a per port basis.

12.3.4.13   MAC23 – Increment MAC port 2,3 address

CPU Address: h0326

Accessed by CPU (RW)

Bits [2:0]:	Bits [42:40] of Port 2 CPU MAC address
Bit [3]:	Reserved
Bits [6:4]:	Bits [42:40] of Port 3 CPU MAC address
Bit [7]:	Reserved

12.3.4.14   MAC45 – Increment MAC port 4,5 address

CPU Address: h0327

Accessed by CPU (RW)

Bits [2:0]:	Bits [42:40] of Port 4 CPU MAC address
Bit [3]:	Reserved
Bits [6:4]:	Bits [42:40] of Port 5 CPU MAC address
Bit [7]:	Reserved

12.3.4.15   MAC67 – Increment MAC port 6,7 address

CPU Address: h0328

Accessed by CPU (RW)

Bits [2:0]:	Bits [42:40] of Port 6 CPU MAC address
Bit [3]:	Reserved
Bits [6:4]:	Bits [42:40] of Port 7 CPU MAC address
Bit [7]:	Reserved

12.3.4.16   MAC9 – Increment MAC port 9 address

CPU Address: h0329

Accessed by CPU (RW)

Bits [7:0]:	Bits [47:40] of Port 9 CPU MAC address

12.3.4.17   CPUQINS0 - CPUQINS6 – CPU Queue Insertion Command

CPU Address: h0330-0336

Accessed by CPU, (R/W)

55   0

CQ6	CQ5	CQ4	CQ3	CQ2	CQ1	CQ0
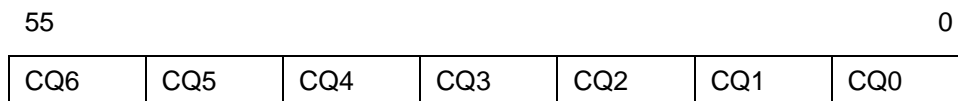
CPU Queue insertion command

CPUQINS0	
Bit[7:0]:	Destination Map (port 7-0).
CPUQINS1	
Bit[9:8]:	Destination Map (MMAC, CPU).
Bits [13:10]	Priority

90

Zarlink Semiconductor Inc.

Bits [15:14]	Number of granules for the frame
CPUQINS2	
Bits [20:16]	Number of granules for the frame (con't)
Bits [23:21]	Tail pointer
CPUQINS3	
Bits [31:24]	Tail pointer (con't)
CPUQINS4	
Bits [35:32]	Tail pointer (con't)
Bits [39:36]	Header Pointer
CPUQINS5	
Bits [47:40]	Header Pointer (con't)
CPUQINS6	
Bits [50:48]	Header Pointer (con't)
Bit [51]	Multicast frame (has to be one if more than one destination port)
Bits [54:52]	Reserved
Bit [55]	Command valid (will be processed on the rising edge of the signal)

12.3.4.18   CPUQINSRPT – CPU Queue Insertion Report

CPU Address: h0337

Accessed by CPU, (RO)

CPU command queue status

Bit [0]:	The command is under processing.
Bit [1]:	Insertion Fail (May be due to queue full, WRED or filtering)

12.3.4.19   CPUGRNHDL0 - CPUGRNHDL1 – CPU Allocated Granule Pointer

CPU Address: h0338-339

Accessed by CPU, (RO)

15   0

CG1	CG0
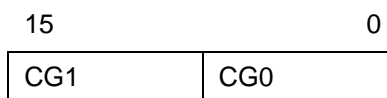
CPU Queue insertion command

Bits [14:0]:	Granule pointer.
Bit [15]:	Pointer valid

Zarlink Semiconductor Inc.

12.3.4.20 CPURLSINFO0 - CPURLSINFO4 – Receive Queue Status

CPU Address: h033A-033E

Accessed by CPU, (R/W)

0

CR4	CR3	CR2	CR1	CR0

CPU Queue insertion command

Bits [14:0]:	Header pointer
Bits [30:15]	Tail pointer
Bits [38:32]	Number of granules for the release

12.3.4.21 CPUGRNCTR – CPU Granule Control

CPU Address: h033f

Accessed by CPU, (R/W)

CPU receive queue status

Bit [0]:	Allocate granule to the CPU if set to one. Otherwise, do not allocate any resource.
Bit [1]:	Read allocated granule (at rising edge only)
Bit [2]:	Release info valid (will be processed at rising edge only)

12.3.5 (Group 4 Address) Search Engine Group

12.3.5.1 AGETIME_LOW – MAC address aging time Low

I²C Address: h049; CPU Address: h0400

Accessed by CPU and I²C (R/W)

Used in conjuction with AGETIME_HIGH. The ZL50411 removes the MAC address from the data base and sends a Delete MAC Address Control Command to the CPU.

Bits [7:0]:	Low byte of the MAC address aging timer (Default 0x5C)

12.3.5.2 AGETIME_HIGH –MAC address aging time High

I²C Address: h04A; CPU Address: h0401

Accessed by CPU and I²C (R/W)

Bits [7:0]:	High byte of the MAC address aging timer (Default 0x00)

Zarlink Semiconductor Inc.

The default setting of AGETIME_LOW/HIGH provides 300 seconds aging time. Aging time is based on the following equation:

{AGETIME_HIGH,AGETIME_LOW} X (# of MAC entries in the memory X 800 μsec). Number of MAC entries = 4 K.

12.3.5.3   SE_OPMODE – Search Engine Operation Mode

CPU Address: h0403

Accessed by CPU (R/W)

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

Bit [0]:	Reserved. Must be 0.
Bit [1]:	Protocol filtering mode 0 – Inclusive (Default) 1 – Exclusive
Bit [2]:	Delete MAC report control 0 – Report MAC address deletion (MAC address is deleted from MCT after aging time) (Default) 1 – Disable report MAC address deletion
Bit [3]:	Delete Control 0 – MAC address entry is removed when it is old enough to be aged (Default) 1 – Disable aging logic from removing MAC during aging However, a report is still sent to the CPU in both cases, when bit [2] = 0
Bit [4]:	Enable RSVP Packet trapping 0 - Disable RSVP Packet trapping. (Default) 1 - Enable RSVP Packet trapping. IP Multicast also needs to be enabled for this function.
Bit [5]	Reserved
Bit [6]:	Disable MCT speed-up aging 0 – Enable speed-up aging when MCT resource is low. (Default) 1 – Disable speed-up aging when MCT resource is low.
Bit [7]:	Slow Learning 0 – Learning is performed independent of search demand (Default) 1 – Enable slow learning. Learning is temporary disabled when search demand is high

12.3.6   (Group 5 Address) Buffer Control/QOS Group

12.3.6.1   QOSC – QOS Control

I²C Address: h04B; CPU Address: h0500

Accessed by CPU and I^{2}C (R/W)

Bit [0]:	Enable TX rate control (on RMAC ports only) 0 – Disable (Default) 1 – Enable

Zarlink Semiconductor Inc.

Bit [1]:	Enable RX rate control (on RMAC ports only) 0 – Disable (Default) 1 – Enable
Bits [4:2]:	Reserved
Bit [5]:	Select VLAN tag or TOS (IP packets) to be preferentially picked to map transmit priority and drop precedence 0 – Select VLAN Tag priority field over TOS (Default) 1 – Select TOS over VLAN tag priority field
Bit [6]:	Select TOS bits for transmission priority 0 – Use TOS [4:2] bits to map the transmit priority (Default) 1 – Use TOS [7:5] bits to map the transmit priority
Bit [7]:	Select TOS bits for drop precedence 0 – Use TOS [4:2] bits to map the drop precedence (Default) 1 – Use TOS [7:5] bits to map the drop precedence

12.3.6.2   PCC – Packet Congestion Control

I^{2}C Address: h068; CPU Address: h0510

Accessed by CPU and I^{2}C (R/W)

Bits [7:0]:	Packet congestion control on the granule count of P0 queue.
	Used to trigger RED at B% when destination port's P0 queue granule count reaches PCC and shared pool is all in use. Granularity is 16 granule (Default 0x6).
See Programming QoS Registers application note, ZLAN-42, for more information	

12.3.6.3   MCC – Multicast Congestion Control

I²C Address: h069; CPU Address: h0511

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Global congestion control on the multicast granule count.
	Used to trigger 100% drop (or flow control, if FC enabled on source port) when total multicast granule count reaches MCC. Granularity is 16 granule (Default 0x6).
See Programming QoS Registers application note, ZLAN-42, for more information	

12.3.6.4   MCCTH – Multicast Threshold Control

CPU Address: h0512

Accessed by CPU (R/W)

Bits [7:0]:	Global congestion control threshold on the multicast granule count. Exceeding the MCCTH threshold is considered as an indication of multicast resources being low.
	Used to trigger RED at B% when when total multicast granule count reaches MCCTH. Granularity is 16 granule (Default 0x3).
See Programming QoS Registers application note, ZLAN-42, for more information	

12.3.6.5   RDRC0 – WRED Rate Control 0

I²C Address: h090; CPU Address: h0513

Accessed by CPU and I^{2}C (R/W)

Bits [3:0]:	Corresponds to the frame drop percentage Y% for WRED. Granularity 6.25%.
Bits [7:4]:	Corresponds to the frame drop percentage X% for WRED. Granularity 6.25%.
See Programming QoS Registers application note, ZLAN-42, for more information	

12.3.6.6   RDRC1 – WRED Rate Control 1

I²C Address: h091; CPU Address: h0514

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Corresponds to the frame drop percentage B% for RED. Granularity 6.25%.
Bits [7:4]:	Corresponds to the frame drop percentage Z% for WRED. Granularity 6.25%.
See Programming QoS Registers application note, ZLAN-42, for more information	

12.3.6.7   RDRC2 – WRED Rate Control 2

CPU Address: h0515

Accessed by CPU (R/W)

Bits [3:0]:	Corresponds to the frame drop percentage RB% for ingress rate control. Granularity 6.25%.
Bits [7:4]:	Corresponds to the frame drop percentage RA% for ingress rate control. Granularity 6.25%.
See Rate Control application note, ZLAN-33, for more information	

12.3.6.8   SFCB – Share FCB Size

I²C Address: h074; CPU Address: h0518

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Expressed in multiples of 16 granules. Buffer reservation for shared pool.

12.3.6.9   C1RS – Class 1 Reserve Size

I²C Address: h075; CPU Address: h0519

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Class 1 FCB Reservation

Buffer reservation for class 1. Granularity 16 granules**. (Default 0)**

12.3.6.10   C2RS – Class 2 Reserve Size

I²C Address: h076; CPU Address: h051A

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Class 2 FCB Reservation

Buffer reservation for class 2. Granularity 16 granules**. (Default 0)**

12.3.6.11   C3RS – Class 3 Reserve Size

I²C Address: h077; CPU Address: h051B

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Class 3 FCB Reservation

Buffer reservation for class 3. Granularity 16 granules**. (Default 0)**

12.3.6.12   AVPML – VLAN Tag Priority Map

I²C Address: h056; CPU Address: h0530

Accessed by CPU and I²C (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 ZL50411. When the packet goes out it carries the original priority.

Zarlink Semiconductor Inc.

Bits [2:0]:	Priority when the VLAN tag priority field is 0 (Default 0) Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bits [5:3]:	Priority when the VLAN tag priority field is 1 (Default 0)
Bits [7:6]:	Priority when the VLAN tag priority field is 2 (Default 0)

12.3.6.13   AVPMM – VLAN Priority Map

I²C Address: h057; CPU Address: h0531

Accessed by CPU and I²C (R/W)

Map VLAN priority into eight level transmit priorities:

Bit [0]:	Priority when the VLAN tag priority field is 2 (Default 0)
Bits [3:1]:	Priority when the VLAN tag priority field is 3 (Default 0)
Bits [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)

12.3.6.14   AVPMH – VLAN Priority Map

I²C Address: h058; CPU Address: h0532

Accessed by CPU and I²C (R/W)

Map VLAN priority into eight level transmit priorities:

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

12.3.6.15   AVDM – VLAN Discard Map

I²C Address: h05C; CPU Address: h0533

Accessed by CPU and I²C (R/W)

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

Bit [0]:	Frame drop precedence when VLAN Tag priority field is 0 (Default 0) 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest)

Zarlink Semiconductor Inc.

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

12.3.6.16   TOSPML – TOS Priority Map

I²C Address: h059; CPU Address: h0540

Accessed by CPU and I²C (R/W)

Map TOS field in IP packet into eight level transmit priorities

Bits [2:0]:	Priority when the TOS field is 0 (Default 0) Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bits [5:3]:	Priority when the TOS field is 1 (Default 0)
Bits [7:6]:	Priority when the TOS field is 2 (Default 0)

12.3.6.17   TOSPMM – TOS Priority Map

I²C Address: h05A; CPU Address: h0541

Accessed by CPU and I²C (R/W)

Map TOS field in IP packet into eight level transmit priorities

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

Zarlink Semiconductor Inc.

12.3.6.18   TOSPMH – TOS Priority Map

I²C Address: h05B; CPU Address: h0542

Accessed by CPU and I²C (R/W)

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

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

12.3.6.19   TOSDML – TOS Discard Map

I²C Address: h05D; CPU Address: h0543

Accessed by CPU and I²C (R/W)

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

Bit [0]:	Frame drop precedence when TOS field is 0 (Default 0) 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest)
Bit [1]:	Frame drop precedence when TOS field is 1 (Default 0)
Bit [2]:	Frame drop precedence when TOS field is 2 (Default 0)
Bit [3]:	Frame drop precedence when TOS field is 3 (Default 0)
Bit [4]:	Frame drop precedence when TOS field is 4 (Default 0)
Bit [5]:	Frame drop precedence when TOS field is 5 (Default 0)
Bit [6]:	Frame drop precedence when TOS field is 6 (Default 0)
Bit [7]:	Frame drop precedence when TOS field is 7 (Default 0)

12.3.6.20   USER_PROTOCOL_n – User Define Protocol 0~7

I²C Address: h0B3+n; CPU Address: h0550+n

Accessed by CPU and I²C (R/W)

(Default 00) This register is duplicated eight times from PROTOCOL 0~7 and allows the CPU to define eight separate protocols.

Bits [7:0]:	User Define Protocol

Zarlink Semiconductor Inc.

12.3.6.21 USER_PROTOCOL_FORCE_DISCARD – User Define Protocol 0~7 Force Discard

I²C Address: h0BB; CPU Address: h0558

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable Protocol 0 Force Discard 1 – Enable 0 – Disable
Bit [1]:	Enable Protocol 1 Force Discard
Bit [2]:	Enable Protocol 2 Force Discard
Bit [3]:	Enable Protocol 3 Force Discard
Bit [4]:	Enable Protocol 4 Force Discard
Bit [5]:	Enable Protocol 5 Force Discard
Bit [6]:	Enable Protocol 6 Force Discard
Bit [7]:	Enable Protocol 7 Force Discard

User Defined Logical Ports and Well Known Ports

The ZL50411 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.

Zarlink Semiconductor Inc.

12.3.6.22   WELL_KNOWN_PORT[1:0]_PRIORITY- Well Known Logic Port 1 and 0 Priority

I²C Address: h0A8; CPU Address: h0560

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for Well known port 0 (23 for telnet)
	Bit[0] 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest) Bits[3:1] Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bits [7:4]:	Priority setting, transmission + dropping, for Well known port 1 (512 for TCP/UDP)

12.3.6.23   WELL_KNOWN_PORT[3:2]_PRIORITY- Well Known Logic Port 3 and 2 Priority

I²C Address: h0A9; CPU Address: h0561

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for Well known port 2 (6000 for XWIN)
Bits [7:4]:	Priority setting, transmission + dropping, for Well known port 3 (443 for HTTP sec)

12.3.6.24   WELL_KNOWN_PORT[5:4]_PRIORITY- Well Known Logic Port 5 and 4 Priority

I²C Address h0AA, CPU Address 562

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for Well known port 4 (111 for sun remote procedure call)
Bits [7:4]:	Priority setting, transmission + dropping, for Well known port 5 (22555 for IP Phone call setup)

12.3.6.25   WELL_KNOWN_PORT[7:6]_PRIORITY- Well Known Logic Port 7 and 6 Priority

I²C Address: h0AB; CPU Address: h0563

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for Well known port 6 (22 for ssh)
Bits [7:4]:	Priority setting, transmission + dropping, for Well known port 7 (554 for rtsp)

Zarlink Semiconductor Inc.

12.3.6.26   WELL_KNOWN_PORT_ENABLE – Well Known Logic Port 0 to 7 Enables

I²C Address: h0AC; CPU Address: h0564

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable Well Known Port 0 Priority 1 – Enable 0 – Disable
Bit [1]:	Enable Well Known Port 1 Priority
Bit [2]:	Enable Well Known Port 2 Priority
Bit [3]:	Enable Well Known Port 3 Priority
Bit [4]:	Enable Well Known Port 4 Priority
Bit [5]:	Enable Well Known Port 5 Priority
Bit [6]:	Enable Well Known Port 6 Priority
Bit [7]:	Enable Well Known Port 7 Priority

12.3.6.27   WELL_KNOWN_PORT_FORCE_DISCARD – Well Known Logic Port 0~7 Force Discard

I²C Address: h0AD; CPU Address: h0565

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable Well Known Port 0 Force Discard 1 – Enable 0 – Disable
Bit [1]:	Enable Well Known Port 1 Force Discard
Bit [2]:	Enable Well Known Port 2 Force Discard
Bit [3]:	Enable Well Known Port 3 Force Discard
Bit [4]:	Enable Well Known Port 4 Force Discard
Bit [5]:	Enable Well Known Port 5 Force Discard
Bit [6]:	Enable Well Known Port 6 Force Discard
Bit [7]:	Enable Well Known Port 7 Force Discard

Zarlink Semiconductor Inc.

12.3.6.28   USER_PORT[7:0]_[LOW/HIGH] – User Define Logical Port 0~7

I²C Address: h092+n(Low); CPU Address: h0570+2n(Low) (n = logical port number)

I²C Address: h09A+n(High); CPU Address: h0571+2n(High)

Accessed by CPU and I²C (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.

```
7                                                    0
┌────────────────────────────────────────────────────┐
│              TCP/UDP Logic Port Low                 │
└────────────────────────────────────────────────────┘
7                                                    0
┌────────────────────────────────────────────────────┐
│              TCP/UDP Logic Port High                │
└────────────────────────────────────────────────────┘
```
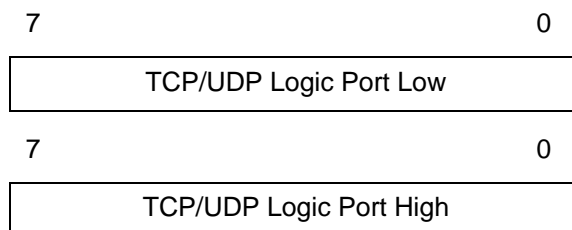
12.3.6.29   USER_PORT_[1:0]_PRIORITY - User Define Logic Port 1 and 0 Priority

I²C Address: h0A2; CPU Address: h0590

Accessed by CPU and I²C (R/W)

The chip allows the CPU to define the priority

Bits [3:0]:	Priority setting, transmission + dropping, for logic port 0
	
	 Bit[0] 0 – Drop Precedence Level 0 (Lowest) (Default)
	               1 – Drop Precedence Level 1 (Highest)
	
	 Bits[3:1] Transmit Priority Level 0 (Lowest)
	               Transmit Priority Level 1
	               ...
	               Transmit Priority Level 6
	               Transmit Priority Level 7 (Highest)

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

12.3.6.30   USER_PORT_[3:2]_PRIORITY - User Define Logic Port 3 and 2 Priority

I²C Address: h0A3; CPU Address: h0591

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for logic port 2
Bits [7:4]:	Priority setting, transmission + dropping, for logic port 3 (Default 00)

12.3.6.31 USER_PORT_[5:4]_PRIORITY - User Define Logic Port 5 and 4 Priority

I²C Address: h0A4; CPU Address: h0592

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for logic port 4
Bits [7:4]:	Priority setting, transmission + dropping, for logic port 5 (Default 00)

12.3.6.32 USER_PORT_[7:6]_PRIORITY - User Define Logic Port 7 and 6 Priority

I²C Address: h0A5; CPU Address: h0593

Accessed by CPU and I²C (R/W)

Bits [3:0]:	Priority setting, transmission + dropping, for logic port 6
Bits [7:4]:	Priority setting, transmission + dropping, for logic port 7 (Default 00)

12.3.6.33 USER_PORT_ENABLE[7:0] – User Define Logic Port 0 to 7 Enables

I²C Address: h0A6; CPU Address: h0594

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable User Port 0 Priority 1 – Enable 0 – Disable
Bit [1]:	Enable User Port 1 Priority
Bit [2]:	Enable User Port 2 Priority
Bit [3]:	Enable User Port 3 Priority
Bit [4]:	Enable User Port 4 Priority
Bit [5]:	Enable User Port 5 Priority
Bit [6]:	Enable User Port 6 Priority
Bit [7]:	Enable User Port 7 Priority

12.3.6.34 USER_PORT_FORCE_DISCARD[7:0] – User Define Logic Port 0~7 Force Discard

I²C Address: h0A7; CPU Address: h0595

Accessed by CPU and I²C (R/W)

Bit [0]:	Enable User Port 0 Force Discard 1 – Enable 0 – Disable
Bit [1]:	Enable User Port 1 Force Discard

Zarlink Semiconductor Inc.

Bit [2]:	Enable User Port 2 Force Discard
Bit [3]:	Enable User Port 3 Force Discard
Bit [4]:	Enable User Port 4 Force Discard
Bit [5]:	Enable User Port 5 Force Discard
Bit [6]:	Enable User Port 6 Force Discard
Bit [7]:	Enable User Port 7 Force Discard

12.3.6.35   RLOWL – User Define Range Low Bit 7:0

I²C Address: h0AE; CPU Address: h05A0

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Lower 8 bit of the User Define Logical Port Low Range

12.3.6.36   RLOWH – User Define Range Low Bit 15:8

I²C Address: h0AF; CPU Address: h05A1

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Upper 8 bit of the User Define Logical Port Low Range

12.3.6.37   RHIGHL – User Define Range High Bit 7:0

I²C Address: h0B0; CPU Address: h05A2

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Lower 8 bit of the User Define Logical Port High Range

12.3.6.38   RHIGHH – User Define Range High Bit 15:8

I²C Address: h0B1; CPU Address: h05A3

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Upper 8 bit of the User Define Logical Port High Range

Zarlink Semiconductor Inc.

12.3.6.39   RPRIORITY – User Define Range Priority

I²C Address: h0B2; CPU Address: h05A4

Accessed by CPU and I²C (R/W)

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

Bit [0]:	Drop Precedence (inclusive only) 0 – Drop Precedence Level 0 (Lowest) (Default) 1 – Drop Precedence Level 1 (Highest)
Bits [3:1]	Transmit Priority (inclusive only) Transmit Priority Level 0 (Lowest) Transmit Priority Level 1 ... Transmit Priority Level 6 Transmit Priority Level 7 (Highest)
Bits [5:4]	Reserved
Bits [7:6]	00 - No Filtering 01 - Exclusive Filtering (x<=RLOW or x>=RHIGH) 10 - Inclusive Filtering (RLOW<x<RHIGH) 11 - Invalid

12.3.7   (Group 6 Address) MISC Group

12.3.7.1   MII_OP0 – MII Register Option 0

I²C Address: h0BC; CPU Address: h0600

Accessed by CPU and I²C (R/W)

Bits [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
Bits [5]	Disable jabber detection. This is for HomePNA applications or any serial operation slower than 10 Mbps. 0 = Enable 1 = Disable
Bits [6]	Reserved
Bit [7]:	Half duplex flow control feature 0 = Half duplex flow control always enable 1 = Half duplex flow control by negotiation

Zarlink Semiconductor Inc.

12.3.7.2   MII_OP1 – MII Register Option 1

I²C Address: h0BD; CPU Address: h0601

Accessed by CPU and I²C (R/W)

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 Enable Register

I²C Address: h0BE; CPU Address: h0602

Accessed by CPU and I²C (R/W)

Bit [0]:	Statistic Counter 0 – Disable (Default) 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.
Bit [1]:	Private VLAN Edge Support 0: Disable (default) 1: Enable If this feature is enabled, use registers PVLAN_Pn to set up the egress protected port map. This feature is only applicable in tagged-based VLAN mode (PVMODE[0]='1'). In port-based VLAN mode (PVMODE[0]='0'), this bit must be 0. See Private VLAN Edge application note, ZLAN-130, for more information.
Bit [2]:	Support DS EF Code. 0 – Disable (Default) 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 ID hashing 0 – Disable (Default) 1 – Enable If this is enabled, the VLAN ID will be used along with the MAC address when determining the HASH index for learning/searching. This is to allow Independent VLAN Learning (IVL). This feature is only applicable in tagged-based VLAN mode (PVMODE[0]='1'). In port-based VLAN mode (PVMODE[0]='0'), this bit must be '0'.

Zarlink Semiconductor Inc.

Bit [4]:	Disable IP Multicast Support 0 – Enable IP Multicast Support (Must also set PVMODE[6]=1) 1 – Disable IP Multicast Support (Default) 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]:	Report to CPU 0 – Disable (Default) 1 – Enable When disable new VLAN port association report, new MAC address report or aging reports are disable for all ports. When enable, register SE_OPMODE is used to enable/disable selectively each function.
Bit [6]:	MII Management State Machine 0: Enable (Default) 1: Disable This bit must be set so that there is no contention on the MDIO bus between MII Management state machine and MIIC & MIID PHY register accesses.
Bit [7]:	MCT Link List structure 0 – Enable (Default) 1 – Disable

12.3.7.4 MIIC0 – MII Command Register 0

CPU Address: h0603

Accessed by CPU (R/W)

Bits [7:0]:	MII Command Data [7:0]

Note: Before programming MII command: set FEN[6]; check MIIC3, making sure RDY; then program MII command.

12.3.7.5 MIIC1 – MII Command Register 1

CPU Address: h0604

Accessed by CPU (R/W)

Bits [7:0]:	MII Command Data [15:8]

Zarlink Semiconductor Inc.

12.3.7.6   MIIC2 – MII Command Register 2

CPU Address: h0605

Accessed by CPU (R/W)

Bits [4:0]	REG_AD – Register PHY Address
Bits [6:5]	OP – Operation code "10" for read command "01" for write command
Bits [7]	Reserved

12.3.7.7   MIIC3 – MII Command Register 3

CPU Address: h0606

Accessed by CPU (R/W)

Bits [4:0]	PHY_AD – 5-bit PHY address
Bit [5]	Reserved
Bit [6]	VALID – Data valid from PHY (Read Only)
Bit [7]	RDY – MII management interface ready (Read Only)

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

12.3.7.8   MIID0 – MII Data Register 0

CPU Address: h0607

Accessed by CPU (RO)

Bits [7:0]:	MII Data [7:0]

12.3.7.9   MIID1 – MII Data Register 1

CPU Address: h0608

Accessed by CPU (RO)

Bits [7:0]:	MII Data [15:8]

Zarlink Semiconductor Inc.

12.3.7.10 USD – One Micro Second Divider

CPU Address: h0609

Accessed by CPU (R/W)

Bits [5:0]:	Divider to get one micro second from M_CLK (only used when not in standard RMII mode)
	In a MII or GPSI system, a 50 MHz M_CLK may not be available. The system designer can decide to use another frequency on the M_CLK signal. To compensate for this, this register is required to be programmed.
	For example. If 20 MHz is used on M_CLK, to compensate for the difference, this register is programmed with 20 to provide 1 usec for internal reference.
Bits [7:6]:	Reserved

12.3.7.11 DEVICE - Device Mode

CPU Address: h060A

Accessed by CPU (R/W)

Bit [0]:	Reserved
Bit [1]:	CPU Interrupt Polarity 0: Negative Polarity 1: Positive Polarity (Default)
Bits [4:2]:	Reserved
Bits [7:5]:	Device ID (Default 0x0). See application note ZLAN-26, Processor Interface, for usage.

12.3.7.12 CHECKSUM - EEPROM Checksum

I²C Address: 0FF; CPU Address: h060B

Accessed by CPU and I²C (R/W)

Bits [7:0]:	Checksum content (Default 0)

This register is used in unmanaged mode only. Before requesting that the ZL50411 updates the EEPROM device, the correct checksum needs to be calculated and written into this checksum register.

The checksum formula is:

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

When the ZL50411 boots from the EEPROM the checksum is calculated and the value must be zero. If the checksum is not zeroed the ZL50411 does not start and pin CHECKSUM_OK is set to zero.

12.3.7.13   LHBTimer – Link Heart Beat Timeout Timer

CPU Address: h0610

Accessed by CPU (R/W)

In slot time (512 bit time). LHB packet will be sent out to the remote device if no other packet is transmitted in half this period. The receiver will trigger LHB timeout interrupt if not receiving any good packet in this period.

12.3.7.14   LHBReg0, LHBReg1 - Link Heart Beat OpCode

CPU Address: h0611, h6012

Accessed by CPU (R/W)

The LHB frame uses MAC control frame format (same as flow control frame.) The register here defines the operation code (we recommend h00-12).

12.3.7.15   fMACCReg0, fMACCReg1 - MAC Control Frame OpCode

CPU Address: h0613, h0614

Accessed by CPU (R/W)

The registers define the operation code if MAC control frame is forced out by processor.

12.3.7.16   FCB Base Address Register 0

I²C Address: h0BF; CPU Address: h0620

Accessed by CPU and I²C (R/W)

Bits [7:0]	FCB Base address bit 7:0 (Default 0)

12.3.7.17   FCB Base Address Register 1

I²C Address: h0C0; CPU Address: h0621

Accessed by CPU and I²C (R/W)

Bits [7:0]	FCB Base address bit 15:8 (Default 0x60)

12.3.7.18   FCB Base Address Register 2

I²C Address: h0C1; CPU Address: h0622

Accessed by CPU and I²C (R/W)

Bits [7:0]	FCB Base address bit 23:16 (Default 0)

12.3.8 (Group 7 Address) Port Mirroring Group

12.3.8.1 MIRROR CONTROL – Port Mirror Control Register

CPU Address: h070C

Accessed by CPU (R/W) (Default 00)

Bits [3:0]:	Destination port to be mirrored to.
Bit [4]	Mirror Flow from MIRROR_SRC_MAC[5:0] to MIRROR_DEST_MAC[5:0]
Bit [5]	Mirror Flow from MIRROR_DEST_MAC[5:0] to MIRROR_SRC_MAC[5:0]
Bit [6]:	Mirror when address is destination
Bit [7]:	Mirror when address is source

12.3.8.2 MIRROR_DEST_MAC[5:0] – Mirror Destination MAC Address 0~5

CPU Address: h0700-0705

Accessed by CPU (R/W)

DEST_MAC5	DEST_MAC4	DEST_MAC3	DEST_MAC2	DEST_MAC1	DEST_MAC0
[47:40] (Default 00)	[39:32] (Default 00)	[31:24] (Default 00)	[23:16] (Default 00)	[15:8] (Default 00)	[7:0] (Default 00)

12.3.8.3 MIRROR_SRC _MAC[5:0] – Mirror Source MAC Address 0~5

CPU Address: h0706-070B

Accessed by CPU (R/W)

SRC_MAC5	SRC_MAC4	SRC_MAC3	SRC_MAC2	SRC_MAC1	SRC_MAC0
[47:40] (Default 00)	[39:32] (Default 00)	[31:24] (Default 00)	[23:16] (Default 00)	[15:8] (Default 00)	[7:0] (Default 00)

12.3.8.4 RMII_MIRROR0 – RMII Mirror 0

CPU Address: h0710

Accessed by CPU (R/W)

Bits [2:0]:	Source port to be mirrored
Bit [3]:	Mirror path 0: Receive 1: Transmit
Bits [6:4]:	Destination port for mirrored traffic
Bit [7]:	Mirror enable

Zarlink Semiconductor Inc.

12.3.8.5 RMII_MIRROR1 – RMII Mirror 1

CPU Address: h0711

Accessed by CPU (R/W)

Bits [2:0]:	Source port to be mirrored
Bit [3]:	Mirror path 0: Receive 1: Transmit
Bits [6:4]:	Destination port for mirrored traffic
Bit [7]:	Mirror enable

12.3.9   (Group 8 Address) Per Port QOS Control

12.3.9.1   FCRn – Port 0~9 Flooding Control Register

I²C Address: h04C+n; CPU Address: h0800+n (n = port number)

Accessed by CPU and I²C (R/W)

Bits [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 from Port n. 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 = 0)
Bits [6:4]:	Time Base for Unicast to Multicast, Multicast and Broadcast rate control of Port n: (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 = 12.8 ms
Bit [7]:	Reserved

12.3.9.2   BMRCn - Port 0~9 Broadcast/Multicast Rate Control

I²C Address: h05E+n; CPU Address: h0820+n (n = port number)

Accessed by CPU and I²C (R/W)

This broadcast and multicast rate defines for Port n, 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 FCR0 [6:4]

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

Zarlink Semiconductor Inc.

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

12.3.9.3   PR100_n – Port 0~7 Reservation

I²C Address: h06A+n; CPU Address: h0840+n (n = port number)

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. (Default 0x6)

12.3.9.4   PR100_CPU – Port CPU Reservation

I²C Address: h073; CPU Address: h0848

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. (Default 0x6)

12.3.9.5   PRM – Port MMAC Reservation

I²C Address: h072; CPU Address: h0849

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. (Default 0x24)

12.3.9.6   PTH100_n – Port 0~7 Threshold

I²C Address: h0C2+n; CPU Address: h0860+n (n = port number)

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. More than this number used on a source port will trigger either random drop or flow control (Default 0x3)

12.3.9.7   PTH100_CPU – Port CPU Threshold

I²C Address: h0CB; CPU Address: h0868

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. More than this number used on a source port will trigger either random drop or flow control (Default 0x3)

12.3.9.8   PTHG – Port MMAC Threshold

I²C Address: h0CA; CPU Address: h0869

Accessed by CPU and I²C (R/W)

Expressed in multiples of 16 granules. More than this number used on a source port will trigger either random drop or flow control (Default 0x12)

12.3.9.9   QOSC00, QOSC01 - Classes Byte Limit port 0

I²C Address: h078-079; CPU Address: h0880-0881

Accessed by CPU and I²C (R/W)

- QOSC00 – Port 0 L1 threshold for queue 1
- QOSC01 – Port 0 L2 threshold for queue 1

Multiple of 16 granules. The two numbers set the two level for WRED on the high priority queue. When the queue size exceeds the L1 threshold, received frame will subject to X% (high drop) or Y% (low drop) WRED. When the queue size exceeds L2 threshold, received frame will either be filtered (high drop) or subject to Z% WRED.

12.3.9.10   QOSC02, QOSC15 - Classes Byte Limit port 1-7

I²C Address: h07A-087; CPU Address: h0882-088F

Accessed by CPU and I²C (R/W)

Same as QOSC00, QOSC01 for ports 1-7

12.3.9.11   QOSC16 - QOSC21 - Classes Byte Limit CPU port

I^{2}C Address: h088-08D; CPU Address: h0890-0895

Accessed by CPU and I^{2}C (R/W):

- QOSC16 – Port 8 (CPU) L1 threshold for queue 1
- QOSC17 – Port 8 (CPU) L2 threshold for queue 1
- QOSC18 – Port 8 (CPU) L1 threshold for queue 2
- QOSC19 – Port 8 (CPU) L2 threshold for queue 2
- QOSC20 – Port 8 (CPU) L1 threshold for queue 3
- QOSC21 – Port 8 (CPU) L2 threshold for queue 3

Multiple of 16 granules. The two numbers set the two level for WRED on the high priority queue. When the queue size exceeds the L1 threshold, received frame will subject to X% (high drop) or Y% (low drop) WRED. When the queue size exceeds L2 threshold, received frame will either be filtered (high drop) or subject to Z% WRED.

12.3.9.12   QOSC22 - QOSC27 - Classes Byte Limit MMAC port

I²C Address: h08E-08F; CPU Address: h0896-089B

Accessed by CPU and I²C (R/W)

- QOSC22 – Port 9 (uplink) L1 threshold for queue 1
- QOSC23 – Port 9 (uplink) L2 threshold for queue 1
- QOSC24 – Port 9 (uplink) L1 threshold for queue 2
- QOSC25 – Port 9 (uplink) L2 threshold for queue 2
- QOSC26 – Port 9 (uplink) L1 threshold for queue 3
- QOSC27 – Port 9 (uplink) L2 threshold for queue 3

Multiple of 16 granules. The two numbers set the two level for WRED on the high priority queue. When the queue size exceeds the L1 threshold, received frame will subject to X% (high drop) or Y% (low drop) WRED. When the queue size exceeds L2 threshold, received frame will either be filtered (high drop) or subject to Z% WRED.

12.3.9.13   QOSC28 - QOSC31 - Classes WFQ Credit For MMAC

CPU Address: h089C-089F

Accessed by CPU (R/W)

- QOSC28 – CREDIT_C00
- QOSC29 – CREDIT_C01
- QOSC30 – CREDIT_C02
- QOSC31 – CREDIT_C03

Bits [5:0] in QOSC28 through QOSC31 represents one set of WFQ parameters for MMAC port. The granularity of the numbers is 1, and their sum must be 64. QOSC31 corresponds to queue 3, that is the highest priority, and QOSC27 corresponds to queue 0.

Default scheduling method will be strict priority across all queues. Only when the bit [7] is set, the queue will be scheduled as WFQ. The credit number also works as shaper credit if bit [6] is set. A queue with shaper enabled will be scheduled by strict priority when the token is available. The shaper setting override the WFQ (bit [7]) setting.

Bits [5:0]:	Class scheduling credit
Bit [6]:	Traffic Shaper Enable 0: Disable 1: Enable
Bit [7]:	Enable WFQ Scheduling 0: Strict Priority 1: WFQ

12.3.9.14   QOSC36 - QOSC39 - Shaper Control Port MMAC

CPU Address: h08A4-08A7

Accessed by CPU (R/W)

- QOSC36 – TOKEN_LIMIT_C00
- QOSC37 – TOKEN_LIMIT_C01
- QOSC38 – TOKEN_LIMIT_C02
- QOSC39 – TOKEN_LIMIT_C03

QOSC36 through QOSC39 represents one set of token limit on the shaper of MMAC port. The granularity of the numbers is 64 bytes. The shaper is implemented as leaky bucket and the limit here works as bucket size. Since the hardware implementation can keep negative number, the limit can be as small as one and still can transmit oversized frame, as long as one byte token is available.

12.3.10   (Group E Address) System Diagnostic

NOTE: Device Manufacturing test registers.

12.3.10.1   DTSRL – Test Output Selection

CPU Address: h0E00

Accessed by CPU (R/W)

Test group selection for testout[7:0].

12.3.10.2   DTSRM – Test Output Selection

CPU Address: h0E01

Accessed by CPU (R/W)

Test group selection for testout[15:8].

12.3.10.3   TESTOUT0, TESTOUT1 – Testmux Output [7:0], [15:8]

CPU Address: h0E02-0E03

Accessed by CPU (RO)

12.3.10.4   MASK0-MASK4 – Timeout Reset Mask

CPU Address: h0E10-E14

Accessed by CPU (R/W)

Disable timeout reset on selected state machine status.

See Programming Timeout Reset application note, ZLAN-41, for more information.

12.3.10.5   BOOTSTRAP0 – BOOTSTRAP3

CPU Address: h0E80-E83

Accessed by CPU (RO)

31	23	15	0
BT3	BT2	BT1	BT0

Bits [15:0]:	Bootstrap value from TSTOUT[15:0]:
	    Bit [6:0]: TSTOUT[6:0]
	    Bit [8:7]: Invert of TSTOUT[8:7]
	    Bit [9]: TSTOUT[11]
	    Bit [10]: TSTOUT[9]
	    Bit [11]: TSTOUT[10]
	    Bit [14:12]: TSTOUT[14:12]
	    Bit [15]: Always 0

Bits [23:16]:	Bootstrap value from M[7:0]_TXEN
	    Bit [16]: M0_TXEN
	    Bit [17]: M1_TXEN
	    ...
	    Bit [23]: M7_TXEN

Bits [31:24]:	Reserved

Zarlink Semiconductor Inc.

12.3.10.6   PRTFSMST0~9

CPU Address: h0E90+n

Accessed by CPU (RO)

Bit [0]:	TX FSM NOT idle for 5 sec
Bit [1]:	TX FIFO control NOT idle for 5 sec
Bit [2]:	RX SFD detection NOT idle for 5 sec
Bit [3]:	RXINF NOT idle for 5 sec
Bit [4]:	PTCTL NOT idle for 5 sec
Bit [5]:	Reserved
Bit [6]	LHB frame detected
Bit [7]:	LHB receiving timeout

12.3.10.7   PRTQOSST0-PRTQOSST7

CPU Address: h0EA0+n

Accessed by CPU (RO)

Bit [0]:	Source port reservation low
Bit [1]:	No source port buffer left
Bit [2]:	Unicast congestion detected on best effort (P0) queue
Bit [3]:	Reserved
Bit [4]:	High priority queue reach L1 WRED level
Bit [5]:	High priority queue reach L2 WRED level
Bit [6]:	Low priority MC queue full
Bit [7]:	High priority MC queue full

12.3.10.8   PRTQOSST8A, PRTQOSST8B (CPU port)

CPU Address: h0EA8 – 0EA9

Accessed by CPU (RO)

15   0

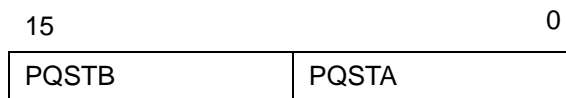| PQSTB	PQSTA

Zarlink Semiconductor Inc.

Bit [0]:	Source port reservation low
Bit [1]:	No source port buffer left
Bit [2]:	Unicast congestion detected on best effort (P0) queue
Bit [3]:	Reserved
Bit [4]:	priority queue 1 reach L1 WRED level
Bit [5]:	priority queue 1 reach L2 WRED level
Bit [6]:	priority queue 2 reach L1 WRED level
Bit [7]:	priority queue 2 reach L2 WRED level
Bit [8]:	priority queue 3 reach L1 WRED level
Bit [9]:	priority queue 3 reach L2 WRED level
Bit [10]:	priority 0 MC queue full
Bit [11]:	priority 1 MC queue full
Bit [12]:	priority 2 MC queue full
Bit [13]:	Priority 3 MC queue full
Bits [15:14]:	Reserved

12.3.10.9   PRTQOSST9A, PRTQOSST9B (MMAC port)

CPU Address: h0EAA – 0EAB

Accessed by CPU (RO)

15	0
PQSTB	PQSTA

Bit [0]:	Source port reservation low
Bit [1]:	No source port buffer left
Bit [2]:	Unicast congestion detected on best effort (P0) queue
Bit [3]:	Reserved
Bit [4]:	Priority queue 1 reach L1 WRED level
Bit [5]:	Priority queue 1 reach L2 WRED level
Bit [6]:	Priority queue 2 reach L1 WRED level
Bit [7]:	Priority queue 2 reach L2 WRED level
Bit [8]:	Priority queue 3 reach L1 WRED level
Bit [9]:	Priority queue 3 reach L2 WRED level
Bit [10]:	Priority 0 MC queue full
Bit [11]:	Priority 1 MC queue full
Bit [12]:	Priority 2 MC queue full

Zarlink Semiconductor Inc.

Bit [13]:	Priority 3 MC queue full
Bits [15:14]:	Reserved

12.3.10.10   CLASSQOSST

CPU Address: h0EAC

Accessed by CPU (RO)

Bit [0]:	No share buffer
Bit [1]:	No class 1 buffer
Bit [2]:	No class 2 buffer
Bit [3]:	No class 3 buffer
Bits [7:4]:	Reserved

12.3.10.11   PRTINTCTR

CPU Address: h0EAD

Accessed by CPU (R/W)

Bit [0]:	Interrupt when source buffer low
Bit [1]:	Interrupt when no source buffer
Bit [2]:	Interrupt when UC congest
Bit [3]:	Interrupt when L1 WRED level
Bit [4]:	Interrupt when L2 WRED level
Bit [5]:	Interrupt when MC queue full
Bit [6]:	Interrupt when LHB timeout
Bit [7]:	Interrupt when no class buffer

12.3.10.12   QMCTRL0~9

CPU Address: h0EB0+n

Accessed by CPU (R/W)

Bit [0]:	Suspend port scheduling (no departure)
Bit [1]:	Reset queue
Bits [4:2]:	Reserved
Bit [5]:	Force out MAC control frame
Bit [6]:	Force out XOFF flow control frame
Bit [7]:	Force out XON flow control frame

12.3.10.13   QCTRL

CPU Address: h0EBA

Accessed by CPU (R/W)

Bit [0]:	Stop QM FSM at idle
Bit [1]:	Stop MCQ FSM at idle
Bit [2]:	Stop new granule grant to any source
Bit [3]:	Stop release granule from any source
Bits [7:4]:	Reserved

12.3.10.14   BMBISTR0, BMBISTR1

CPU Address: h0EBB - 0EBC

Accessed by CPU (RO)

12.3.10.15   BMControl

CPU Address: h0EBD

Accessed by CPU (R/W)

Bits [3:0]:	Block Memory redundancy control 0: Use hardware detected value All others: Overwrite the hardware detected memory swap map
Bits [7:4]:	Reserved

Zarlink Semiconductor Inc.

12.3.10.16   BUFF_RST

CPU Address: h0EC0

Accessed by CPU (R/W)

Bits [3:0]	Assign a value that the pool to be reset 0: port 0 pool 1: port 1 pool 2: port 2 pool 3: port 3 pool 4: port 4 pool 5: port 5 pool 6: port 6 pool 7: port 7 pool 8: port MMAC pool 9: shared pool 10: class 1 pool 11: class 2 pool 12: class 3 pool 13: multicast pool 14: cpu pool 15: reserved
Bit [4]	If this bit is 1, then all the pools are assigned
Bit [5]	Set 1 to reset the pools that are assigned
Bits [7:6]	Reserved

If CPU wants to reset pools again, CPU has to clear bit 5 and then set bit 5.

Note: Before CPU doing so, CPU should set QCTRL (CPU Address EBA) bit 2 and bit 3 to one. After reset the pools, CPU shall reprogram free granule link list (CPU address EC1, EC2, EC3, EC4, EC5, EC6). Then clear QCTRL (EBA).

12.3.10.17   FCB_HEAD_PTR0, FCB_HEAD_PTR1

CPU address: h0EC1

Accessed by CPU (R/W)

Bits [7:0]	Fcb_head_ptr[7:0]. The head pointer of free granule link that CPU assigns.

CPU address: h0EC2

Accessed by CPU (R/W)

Bits [6:0]	Fcb_head_ptr[14:8]. The head pointer of free granule link that CPU assigns.
Bit [7]	Set 1 to write

If CPU wants to write again, CPU has to clear bit 15 and then set bit 15.

Zarlink Semiconductor Inc.

12.3.10.18   FCB_TAIL_PTR0, FCB_TAIL_PTR1

CPU address: h0EC3

Accessed by CPU (R/W)

Bits [7:0]	Fcb_tail_ptr[7:0]. The tail pointer of free granule link that CPU assigns.

CPU address: h0EC4

Accessed by CPU (R/W)

Bits [6:0]	Fcb_tail_ptr[14:8]. The tail pointer of free granule link that CPU assigns.
Bit [7]	Set 1 to write

If CPU wants to write again, CPU has to clear bit 15 and then set bit 15.

12.3.10.19   FCB_NUM0, FCB_NUM1

CPU address: h0EC5

Accessed by CPU (R/W)

Bits [7:0]	Fcb_number[7:0]. The total number of granules that CPU assigns.

CPU address: h0EC6

Accessed by CPU (R/W)

Bits [6:0]	Fcb_number[14:8]. The total number of granules that CPU assigns.
Bit [7]	Set 1 to write

If CPU wants to write again, CPU has to clear bit 15 and then set bit 15.

Note: There are two ways to reprogram the free granules.

1. CPU links all the granules: CPU writes memory directly, at last write head pointer (address EC1, EC2), tail pointer (address EC3, EC4) and granule number (address EC5, EC6).
2. CPU tells Buffer Manager to link: CPU clear head pointer (address EC1, EC2), clear tail pointer (address EC3, EC4), then write granule number that tells Buffer Manager to link (address EC5, EC6).

12.3.10.20   BM_RLSFF_CTRL

CPU address: h0EC7

Accessed by CPU (R/W)

Bit [0]	Read BM release FIFO.
Bits [7:1]	Reserved

The information of BM release FIFO is relocated to registers BM_RLSFF_INFO (address ECD, ECC, ECB, ECA, EC9 and EC8). If the FIFO is not empty, CPU can read out the next by setting the bit 0. Read only happens when bit 0 is changing from 0 to 1.

Zarlink Semiconductor Inc.

12.3.10.21   BM_RSLFF_INFO[5:0]

CPU address: h0EC8

Accessed by CPU (RO)

Bits [7:0]	Rls_head_ptr[7:0].

CPU address: h0EC9

Accessed by CPU (RO)

Bits [6:0]	Rls_head_ptr[14:8].
Bit [7]	Rls_tail_ptr[0]

CPU address: h0ECA

Accessed by CPU (RO)

Bits [7:0]	Rls_tail_ptr[8:1]

CPU address: h0ECB

Accessed by CPU (RO)

Bits [5:0]	Rls_tail_ptr[14:9]
Bits [7:6]	Rls_count[1:0]

CPU address: h0ECC

Accessed by CPU (RO)

Bits [4:0]	Rls_count[6:2]
Bit [5]	If 1, then It is multicast packet.
Bits [7:6]	Rls_src_port[1:0[

CPU address: h0ECD

Accessed by CPU (RO)

Bits [1:0]	Rls_src_port[3:2]
Bits [3:2]	Class[1:0]
Bit [4]	This release request is from QM directly.
Bits [7:5]	Entries count in release FIFO, 0 means FIFO is empty

Zarlink Semiconductor Inc.

12.3.11 (Group F Address) CPU Access Group

12.3.11.1 GCR - Global Control Register

CPU Address: h0F00

Accessed by CPU (R/W)

Bit [0]:	Store configuration (Default = 0)
	Write '1' followed by '0' to store configuration into external EEPROM
Bit [1]:	Store configuration and reset (Default = 0)
	Write '1' to store configuration into external EEPROM and reset chip
Bit [2]:	Start BIST (Default = 0)
	Write '1' followed by '0' to start the device's built-in self-test. The result is found in the DCR register.
Bit [3]:	Soft Reset (Default = 0)
	Write '1' to reset chip
Bit [4]:	Initialization Completed (Default = 0)
	This bit is reserved in unmanaged mode.
	In managed mode, the CPU writes this bit with '1' to indicate initialization is completed and ready to forward packets. The '0' to '1' transition will toggle TSTOUT[2] from low to high.
Bits [7:5]:	Reserved

Zarlink Semiconductor Inc.

12.3.11.2   DCR - Device Status and Signature Register

CPU Address: h0F01

Accessed by CPU (RO)

Bit [0]:	1: Busy writing configuration to I²C 0: Not busy (not writing configuration to I²C)
Bit [1]:	1: Busy reading configuration from I²C 0: Not busy (not reading configuration from I²C)
Bit [2]:	1: BIST in progress 0: BIST not running
Bit [3]:	1: RAM Error 0: RAM OK
Bits [5:4]:	Device Signature 10: ZL50411 device
Bits [7:6]:	Revision 00: Initial Silicon 01: Second Silicon 10: Third Silicon

12.3.11.3   DCR1 - Device Status Register 1

CPU Address: h0F02

Accessed by CPU (RO)

Bits [6:0]	Reserved
Bit [7]	Chip initialization completed

12.3.11.4   DPST – Device Port Status Register

CPU Address: h0F03

Accessed by CPU (R/W)

Bits [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 CPU Operating mode and Negotiation status
	  - 5'b01001 - Port MMAC Operating mode and Negotiation status

Bits [7:5]:	Reserved

Zarlink Semiconductor Inc.

12.3.11.5   DTST – Data read back register

CPU Address: h0F04

Accessed by CPU (RO)

This register provides various internal information as selected in DPST bit [4:0]. Refer to the PHY Port Control Application Note, ZLAN-37.

Bit [0]	Flow control enable 1: Flow control 0: No flow control
Bit [1]	Full duplex port 1: Full duplex 0: Half duplex
Bit [2]	Fast Ethernet port 1: FE Port
Bit [3]	Link is down 1: Link down 0: Link up
Bit [4]	Auto negotiation disabled 1: Disable 0: Enable
Bit [5]	Reserved
Bit [6]	Reserved
Bit [7]	Module detected (for hot swap purpose) 0: No module 1: Module detected Note: If Module Detect feature is disabled (bootstrap TSTOUT[9]='0'), this bit will always be '1'.

12.3.11.6   DA – Dead or Alive Register

CPU Address: h0FFF

Accessed by CPU (RO)

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

Bits [7:0]	Always return DA

Zarlink Semiconductor Inc.

13.0   Characteristics and Timing

13.1   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}	+2.95 V to +3.65 V
Supply Voltage V_{DD} with Respect to V_{SS}	+1.60 V to +2.00 V
Voltage on 5 V Tolerant Input Pins	-0.5 V to (V_{CC} + 2.5 V)
Voltage on Other Pins	-0.5 V to (V_{DD} + 0.3 V)

Caution: Stress above those listed may damage the device. Exposure to the Absolute Maximum Ratings for extended periods may affect device reliability. Functionality at or above these limits is not implied.

13.2   DC Electrical Characteristics

V_{CC} = 3.3 V +/- 10% $T_{AMBIENT}$ = -40 C to +85 C

V_{DD} = 1.8 V +/- 5%

Zarlink Semiconductor Inc.

...

13.3   Recommended Operating Conditions

Symbol	Parameter Description	Min.	Typ.	Max.	Unit
f_{osc}	Frequency of Operation (SCLK)		50	100	MHz
I_{CC}	V_{CC} (3.3V) Supply Current @ 50 MHz			50	mA
I_{DD}	V_{DD} (1.8V) Supply Current @ 50 MHz			180	mA
Pwr	Power @ SCLK = 50MHz			0.49	W
V_{OH}	Output High Voltage (CMOS)	2.4			V
V_{OL}	Output Low Voltage (CMOS)			0.4	V
V_{IH}	Input High Voltage (TTL 5 V tolerant)	2.0		$V_{CC} + 2.0$	V
V_{IL}	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	μA
I_{OL}	Output Leakage Current (0.1 V < V_{OUT} < V_{CC})			10	μA
C_{IN}	Input Capacitance			5	pF
C_{OUT}	Output Capacitance			5	pF
$C_{I/O}$	I/O Capacitance			7	pF
θ_{ja}	Thermal resistance with 0 air flow			24.3	C/W
θ_{ja}	Thermal resistance with 1 m/s air flow			20.0	C/W
θ_{ja}	Thermal resistance with 2 m/s air flow			18.1	C/W
θ_{jc}	Thermal resistance between junction and case			4.6	C/W

Zarlink Semiconductor Inc.

13.4 AC Characteristics and Timing

13.4.1 Typical Reset & Bootstrap Timing Diagram
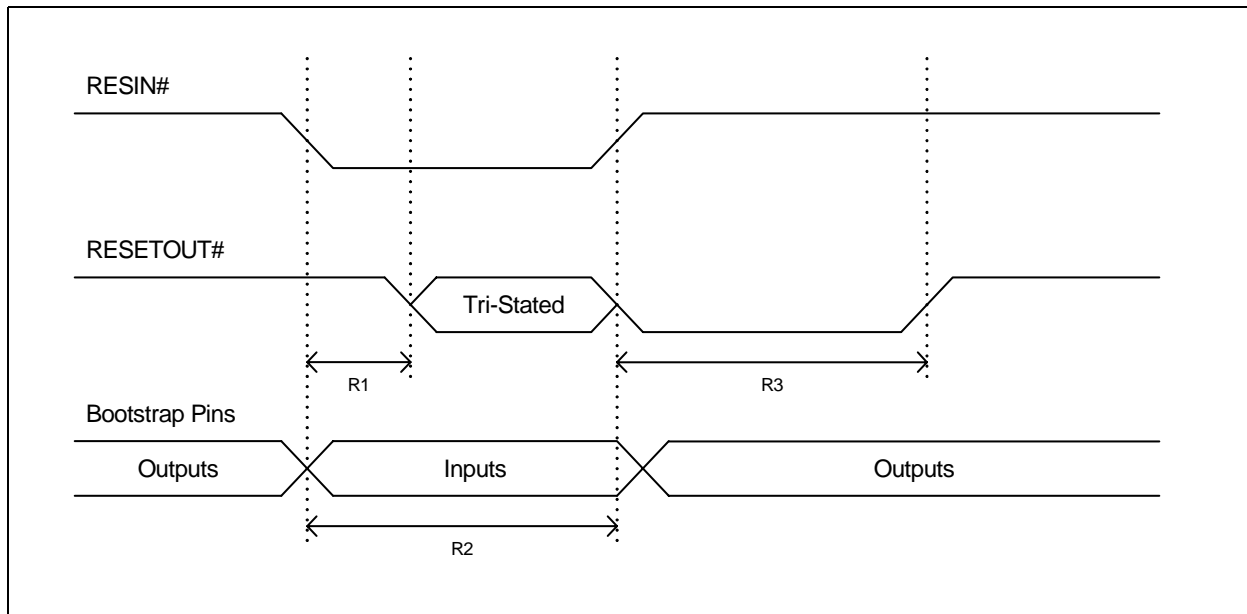
Figure 14 - Typical Reset & Bootstrap Timing Diagram

Symbol	Parameter	Min.	Typ.	Refer to Figure 14
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 μs	10 μs	Bootstrap pins sampled on rising edge of RESIN#
R3	RESETOUT# assertion		2 ms	

Table 13 - Reset Timing

Zarlink Semiconductor Inc.

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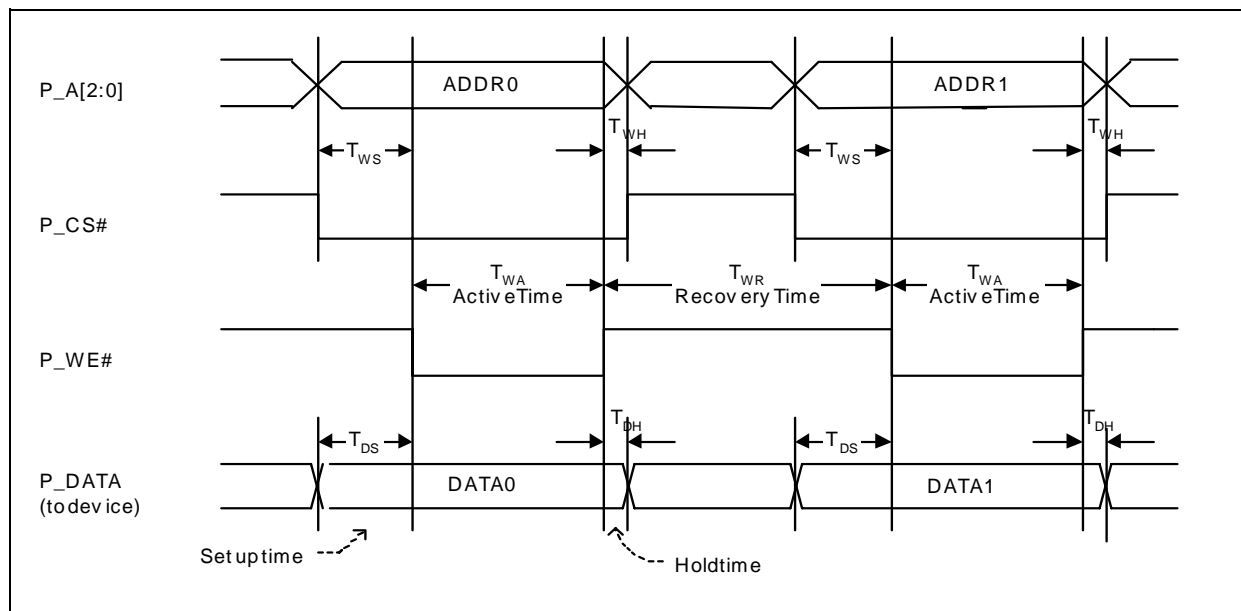
Figure 15 - Typical CPU Timing Diagram for a CPU Write Cycle

Description		(SCLK=100 Mhz)		(SCLK=50 Mhz)		
Write Cycle	**Symbol**	**Min. (ns)**	**Max. (ns)**	**Min. (ns)**	**Max. (ns)**	**Refer to Figure 15**
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		40		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		60		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#

Table 14 - CPU Write Timing

131

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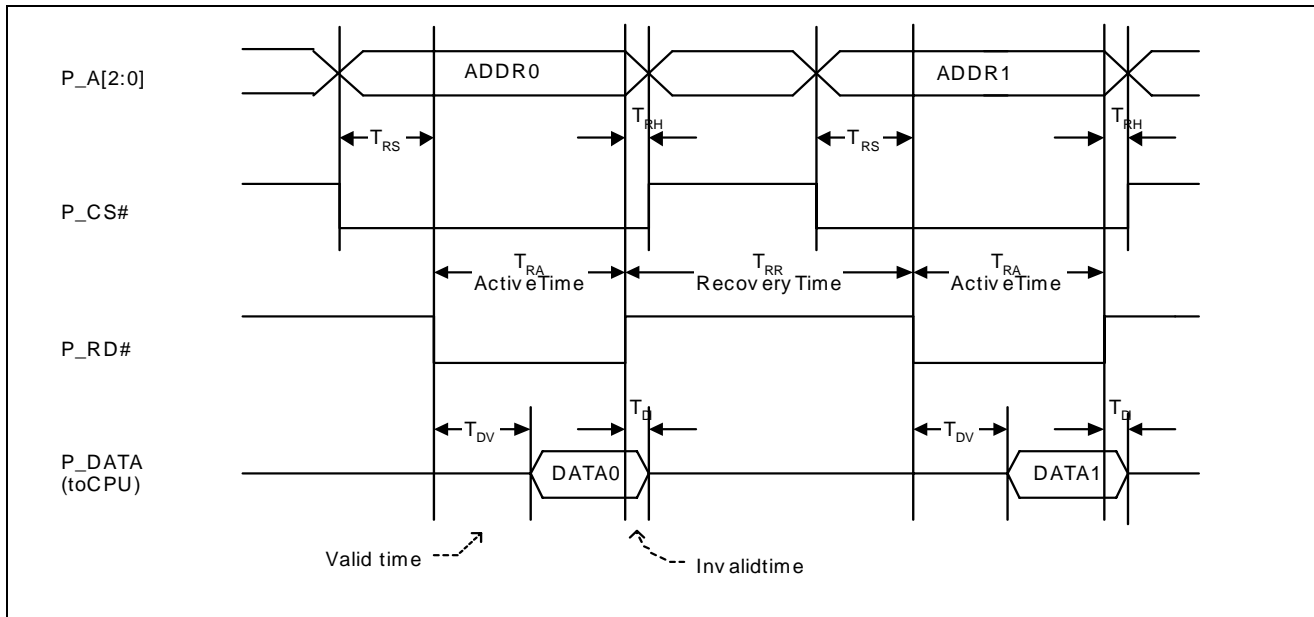
Figure 16 - Typical CPU Timing Diagram for a CPU Read Cycle

Description		(SCLK=100 Mhz)		(SCLK=50 Mhz)		
Read Cycle	**Symbol**	**Min. (ns)**	**Max. (ns)**	**Min. (ns)**	**Max. (ns)**	**Refer to Figure 16**
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		40		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		60		At least 3 SCLK cycles
Data Valid time	T_{DV}		12		12	P_DATA to falling edge of P_RD#
Data Invalid time	T_{DI}		10		10	P_DATA to rising edge of P_RD#

Table 15 - CPU Read Timing

132

Zarlink Semiconductor Inc.

13.4.4   Synchronous Serial Interface (SSI)
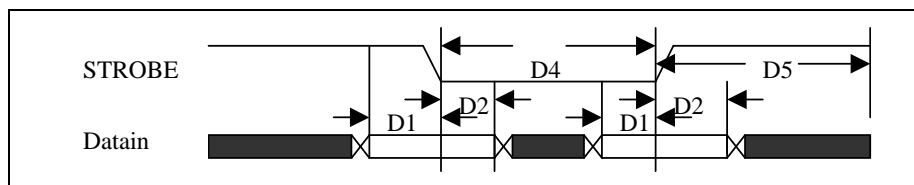
Figure 17 - SSI Setup & Hold Timing
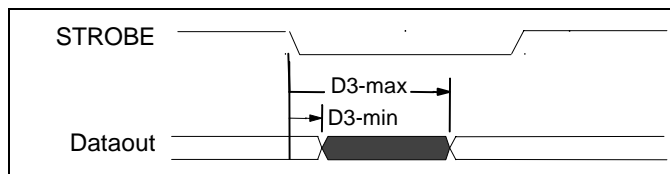
Figure 18 - SSI Output Delay Timing

Symbol	Parameter	Min. (ns)	Max. (ns)	Notes
D1	DATAIN setup time	20		
D2	DATAIN hold time	3 μs		Debounce on
		20		Debounce off
D3*	DATAOUT output delay time	1	50	Debounce on C_L = 100 pf
D3**				Debounce off C_L = 100 pf
D4	STROBE low time	5 μs		Debounce on
		50		Debounce off
D5	STROBE high time	5 μs		Debounce on
		50		Debounce off
	STROBE frequency of operation		100 kHz	Debounce on
			10 MHz	Debounce off
* Open Drain Output. Low to High transition is controlled by external pullup resistor.				
** Totem Pole Output.				

133

Zarlink Semiconductor Inc.

13.4.5   EEPROM Inter-Integrated Circuit (I²C)
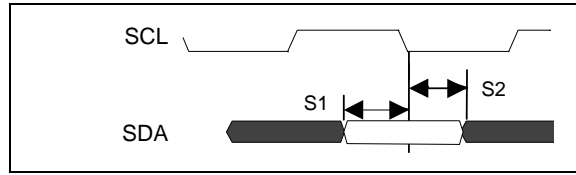
Figure 19 - I²C Setup & Hold Timing
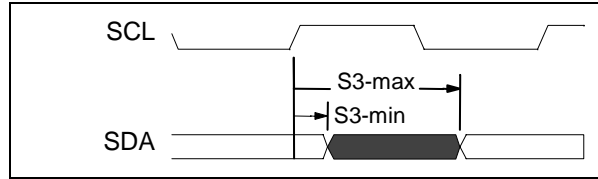
Figure 20 - I²C Output Delay Timing

Symbol	Parameter	SCL=50 KHz		Notes
		Min. (ns)	Max. (ns)	
S1	SDA input setup time	20		
S2	SDA input hold time	1		
S3*	SDA output delay time	4 µs	6 µs	C_L = 30 pf
* Open Drain Output. Low to High transition is controlled by external pullup resistor.				

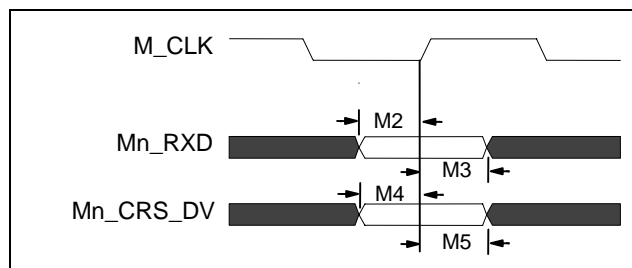### 13.4.6   Reduced Media Independent Interface (RMII)
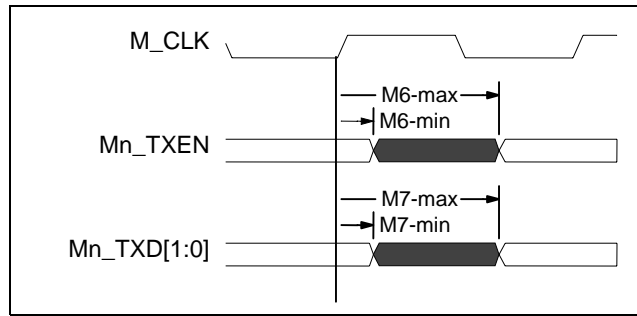
Figure 21 - RMII Transmit Timing

Figure 22 - RMII Receive Timing

Symbol	Parameter	M_CLK=50 MHz		Notes
		Min. (ns)	**Max. (ns)**	
M2	M[7:0]_RXD[1:0] Input Setup Time	4		
M3	M[7:0]_RXD[1:0] Input Hold Time	2		
M4	M[7:0]_CRS_DV Input Setup Time	4		
M5	M[7:0]_CRS_DV Input Hold Time	3		
M6	M[7:0]_TXEN Output Delay Time	2	11	C_L = 20 pF
M7	M[7:0]_TXD[1:0] Output Delay Time	2	11	C_L = 20 pF

Zarlink Semiconductor Inc.

13.4.7   Media Independent Interface (MII)
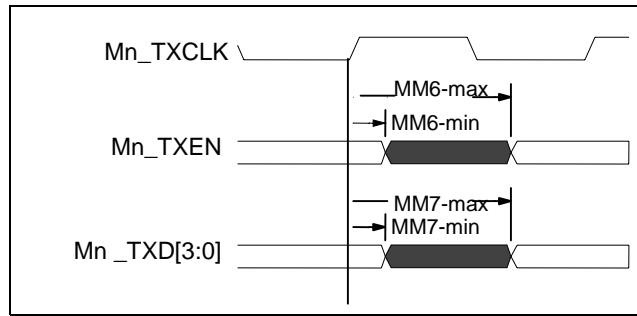
Figure 23 - MII Transmit Timing
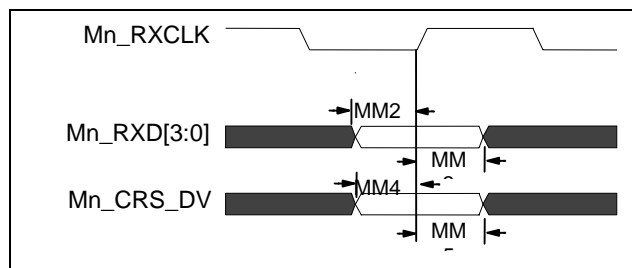
Figure 24 - MII Receive Timing

Symbol	Parameter	25 MHz		Notes
		Min. (ns)	Max. (ns)	
MM2	Mn_RXD[3:0] Input Setup Time	4		
MM3	Mn_RXD[3:0] Input Hold Time	2		
MM4	Mn_CRS_DV Input Setup Time	4		
MM5	Mn_CRS_DV Input Hold Time	2		
MM6	Mn_TXEN Output Delay Time	2	14	C_L = 20 pF
MM7	Mn_TXD[3:0] Output Delay Time	2	14	C_L = 20 pF

Zarlink Semiconductor Inc.

13.4.8   Reverse Media Independent Interface (RvMII)
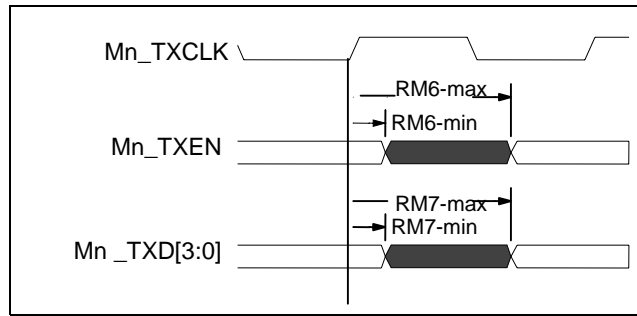
Figure 25 - RvMII Transmit Timing
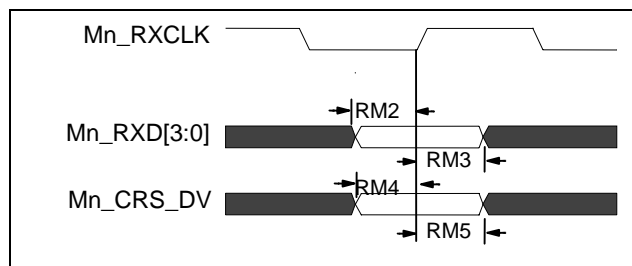
Figure 26 - RvMII Receive Timing

Symbol	Parameter	25 MHz		Notes
		Min. (ns)	Max. (ns)	
RM2	Mn_RXD[3:0] Input Setup Time	4		
	CPU_MII_RXD[3:0] Input Setup Time	10		
RM3	Mn_RXD[3:0] Input Hold Time	2		
	CPU_MII_RXD[3:0] Input Hold Time			
RM4	Mn_CRS_DV Input Setup Time	4		
	CPU_MII_CRS_DV Input Setup Time	10		
RM5	Mn_CRS_DV Input Hold Time	2		
	CPU_MII_CRS_DV Input Hold Time			
RM6*	Mn_TXEN Output Delay Time	2	14	C_L = 20 pF
	CPU_MII_TXEN Output Delay Time	0		
RM7*	Mn_TXD[3:0] Output Delay Time	2	14	C_L = 20 pF
	CPU_MII_TXD[3:0] Output Delay Time	0		
* May need to add external delay depending on other MAC device's min. hold time.				

Zarlink Semiconductor Inc.

13.4.9   General Purpose Serial Interface (GPSI)
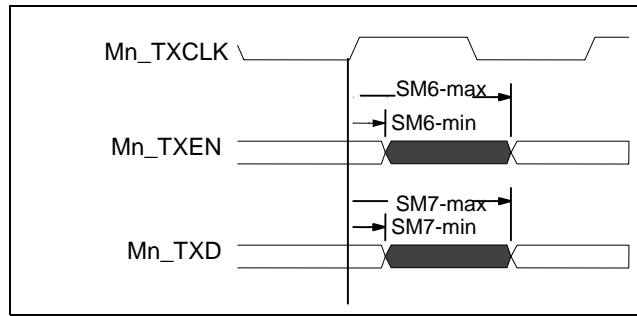
Figure 27 - GPSI Transmit Timing
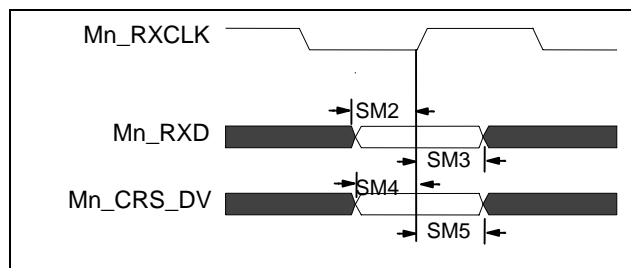
Figure 28 - GPSI Receive Timing

Symbol	Parameter	10 MHz		Notes
		Min. (ns)	**Max. (ns)**	
SM2	M[7:0]_RXD Input Setup Time	4		
SM3	M[7:0]_RXD Input Hold Time	2		
SM4	M[7:0]_CRS_DV Input Setup Time	4		
SM5	M[7:0]_CRS_DV Input Hold Time	2		
SM6	M[7:0]_TXEN Output Delay Time	2	14	C_L = 20 pF
SM7	M[7:0]_TXD Output Delay Time	2	14	C_L = 20 pF

Zarlink Semiconductor Inc.

13.4.10   Reverse General Purpose Serial Interface (RvGPSI)
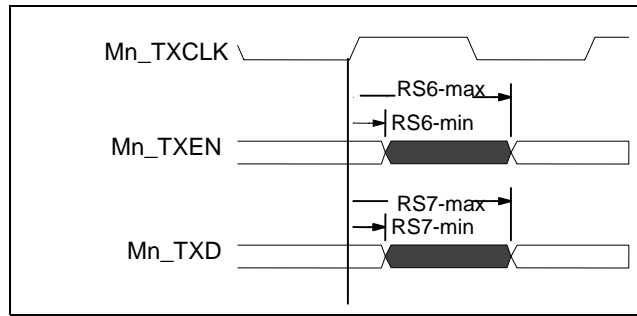
Figure 29 - RvGPSI Transmit Timing
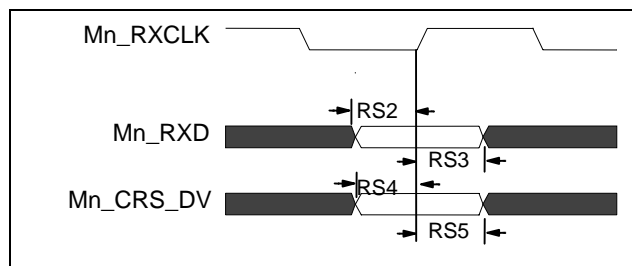
Figure 30 - RvGPSI Receive Timing

Symbol	Parameter	10 MHz		Notes
		Min. (ns)	Max. (ns)	
RS2	M[7:0]_RXD Input Setup Time	4		
RS3	M[7:0]_RXD Input Hold Time	2		
RS4	M[7:0]_CRS_DV Input Setup Time	4		
RS5	M[7:0]_CRS_DV Input Hold Time	2		
RS6*	M[7:0]_TXEN Output Delay Time	2	14	C_L = 20 pF
RS7*	M[7:0]_TXD Output Delay Time	2	14	C_L = 20 pF
* May need to add external delay depending on other MAC device's min. hold time.				

139

Zarlink Semiconductor Inc.

13.4.11   MII Management Data Interface (MDIO/MDC)
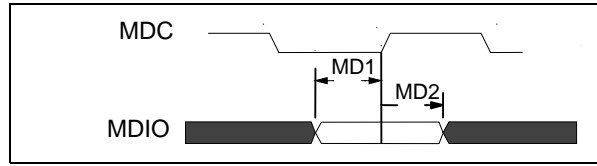
Figure 31 - MDIO Setup & Hold Timing
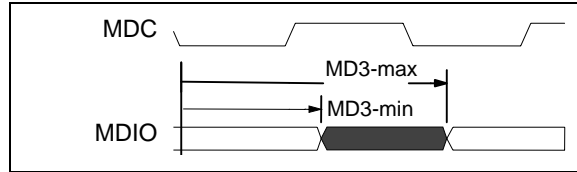
Figure 32 - MDIO Output Delay Timing

Symbol	Parameter	MDC=500 KHz		Notes
		Min. (ns)	Max. (ns)	
MD1	MDIO input setup time	10		
MD2	MDIO input hold time	2		
MD3	MDIO output delay time	1	20	C_L = 50 pf

13.4.12   JTAG (IEEE 1149.1-2001)
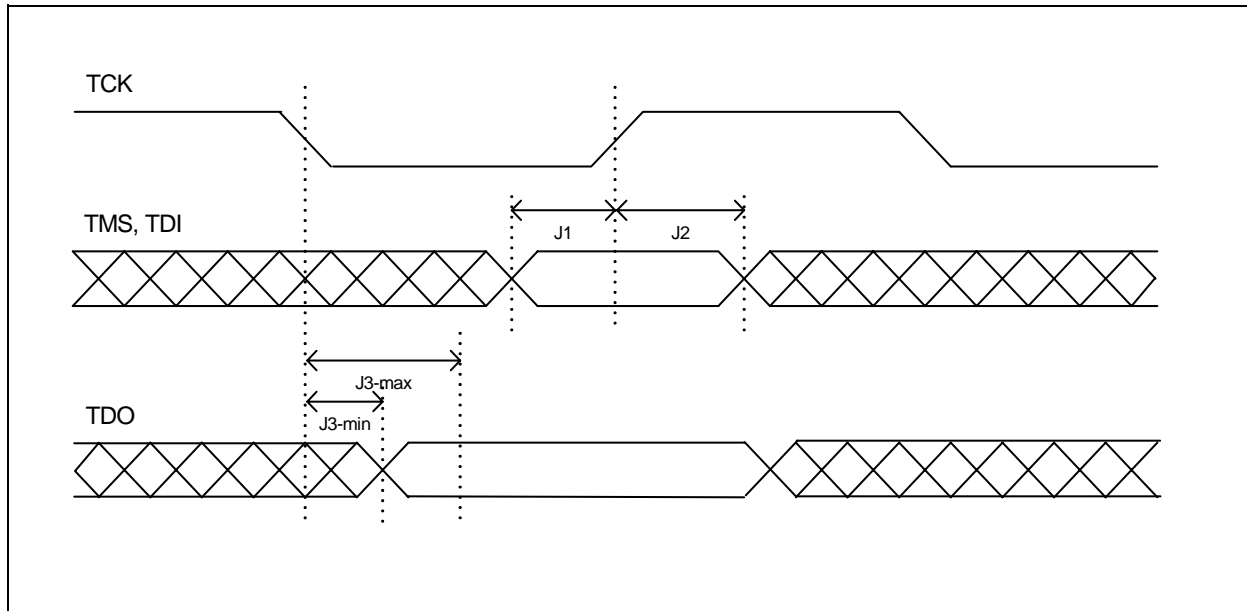
Figure 33 - JTAG Timing Diagram

Symbol	Parameter	Min.	Typ.	Max.	Units	Refer to Figure 33
	TCK frequency of operation	0	10	50	MHz	
	TCK cycle time	20			ns	
	TCK clock pulse width	10			ns	
	TRST# assert time	20		-	ns	TRST is an asynchronous signal
J1	TMS, TDI data setup time	3			ns	
J2	TMS, TDI data hold time	7			ns	
J3	TCK to TDO data valid	0		15	ns	

Zarlink Semiconductor Inc.

TOP VIEW

PIN 1 CORNER

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

A B C D E F G H J K L M N P R T

BOTTOM VIEW

PIN 1 CORNER

16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

A B C D E F G H J K L M N P R T

E

D

e

Dimension	MIN	MAX
A	-	1.40
A1	0.30	0.50
A2	0.53 REF	
D	16.90	17.10
E	16.90	17.10
b	0.40	0.60
e	1.00	
N	208	
Conforms to JEDEC MO-192		

SIDE VIEW

0.2

SEATING PLANE

A2

-C-

b

A1 A
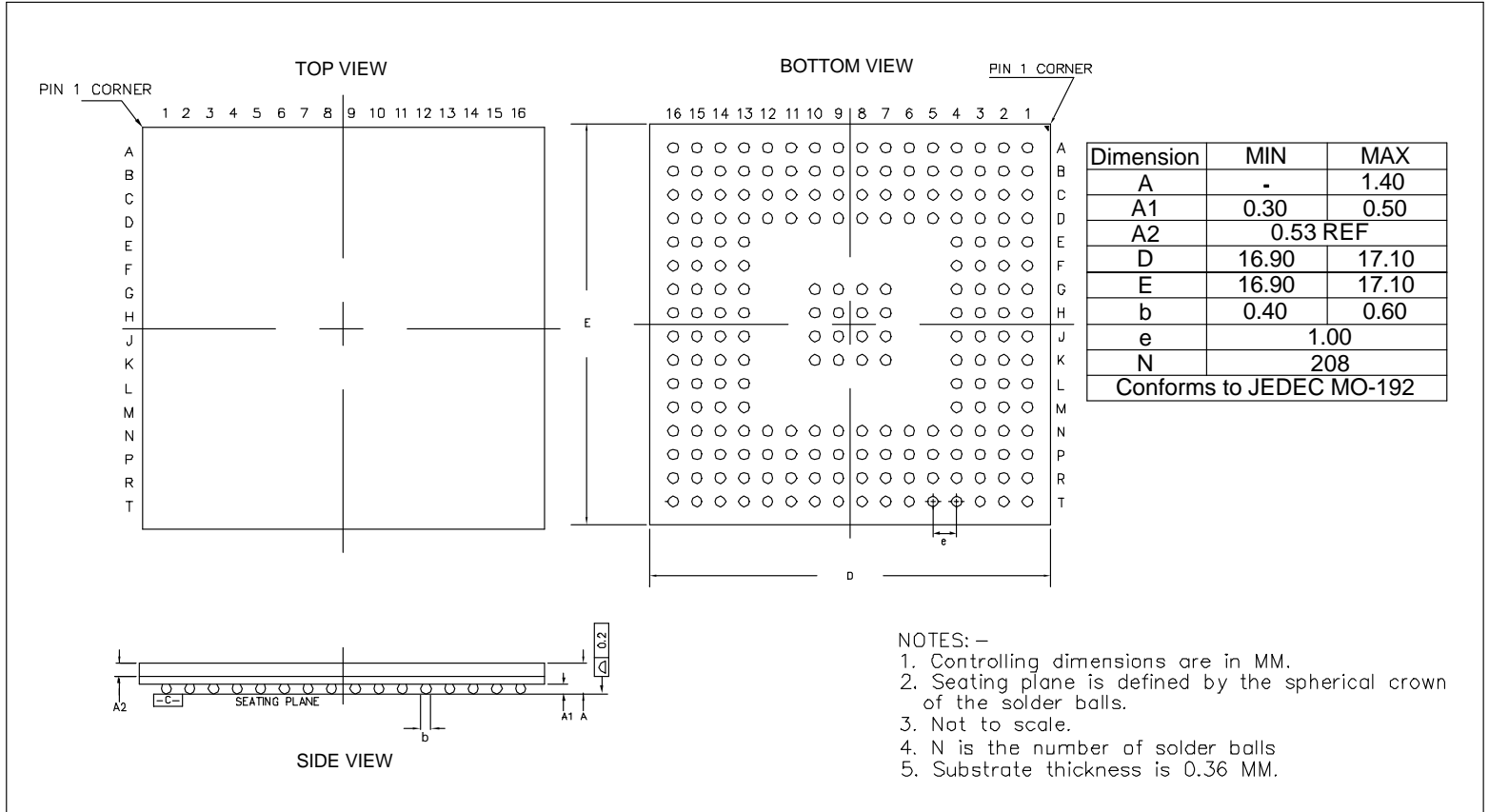
NOTES: —
1. Controlling dimensions are in MM.
2. Seating plane is defined by the spherical crown
   of the solder balls.
3. Not to scale.
4. N is the number of solder balls
5. Substrate thickness is 0.36 MM.

ISSUE	1			
ACN	213730			
DATE	14Nov02			
APPRD.				

ZARLINK
SEMICONDUCTOR

Previous package codes

Package Code GD

Package Outline for
208ball LBGA
17x17x1.4mm Max

GPD00802

ZARLINK
SEMICONDUCTOR

For more information about all Zarlink products
visit our Web Site at

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^2C components conveys a licence under the Philips I^2C Patent rights to use these components in and I^2C System, provided that the system conforms to the I^2C 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