# 8-Port 10/100 + 2-Port 10/100/1000Mb Ethernet Switch

## Features List

- **8-Port 10/100Mb + 2-Port 10/100/1000Mb Ethernet Switch**
  - Built-in 8 10/100Mb PHYs
  - RGMII interface for 1 10/100/1000Mb PHYs
  - RGMII/MII interface for external CPU
  - Support IEEE802.3az
  - Support 10/100Mb full/half duplex
  - Support 1000Mb full duplex
  - 100Mb TP and Fiber dual mode, selected by Signal Detection (SD) level
- **Store & Forward, Share Memory Non-blocking Architecture**
  - Built-in 4Mb SRAM for packet buffer
  - Support 16K Jumbo packet
  - Max. length 1664B without supporting jumbo packet
- **Wire-Speed Operation On Every Port**
- **Head Of Line Blocking Prevention**
- **Flow Control**
  - 802.3x compliant flow control in full duplex
  - Collision/Carrier_ sense based backpressure in half duplex
- **Internal 16K MAC Address Entities**
  - CRC/Direct hashing algorithm
  - Aging timer programmable (55s~500hrs)
  - Wire speed address learning and resolution
  - CPU accessible for security and static MAC
  - Learning enable/disable
- **IGMP/MLD Snooping**
  - IGMP Version 1, 2, 3 / MLD Version 1, 2
  - MLD snooping. ( 4 sets IP address per MLD list )
  - Snooping by Switch ASIC or CPU
- **Two Trunk Group**
  - Two trunk groups, trunk A~B up to 4 ports
  - Load balance based on (Port ID, DA, SA, DA/SA, IP, TCP/UTP)
  - Link fault recovery
- **Per Port 41 MIB Counters/port**
  - RMON/ Ethernet/ MIB II
- **Hardware Auto loop detection**
- **VLAN**
  - 10 Port based
  - 4K Tag based
  - Support Tag remove/add/modify
  - IVL/SVL learning mode
  - Support Protocol VLAN
  - Support Q-in-Q (double tag)
  - 64 configurable VID for Q-in-Q tag stacking
- **Class Of Service (CoS)**
  - Support FIFS/SP/WRR/SP+WRR/WFQ/ TWRR in Output Queue Schedule Modes
  - Port based priority
  - 802.1Q priority tag based
  - IP TOS based (IPv4/IPv6)
  - TCP/UDP port number based
  - Source MAC address based
  - ACL based
  - Privilege priority
  - 8 levels per port
- **Priority queuing decision**
  - DSCP/ TAG priority remarking
  - 802.1Qad PCP, DEN insert / modify
  - 802.1Q PCP, CFI insert/ modify
  - RX &TX Priority re-mapping
  - LLQ + latency
- **Broadcast/multicast/DLF/ARP/ICMP Storm Control**
  - 256 levels resolution for all storm control
  - Storm control can be enabled for per port
  - With option to drop all ARP to CPU
  - Unit time can be selected for all storm control
- **Sniffer Function**
  - Ingress、egress、ingress/egress methods
  - Two Sniffer destination port group configurations
  - Add/ Remove Tag option for packet routing to mirroring ports
  - ACL and special tag for sniffer application
- **Port Security**
  - MAC based
  - TCP/UDP port based
  - SIP based
  - 802.1x Port based

- **Bandwidth Control**
  - From 64K bps to wire speed (resolution 64k)
  - Support Flow control on/off
  - Queue based bandwidth control (resolution 64k/1M/2M/4M)
- **SMI Interface Auto-Polling**
  - Speed, Duplex, Flow control, Link
  - CPU accessible and interrupt
  - CPU R/W PHY register
  - Support MMD access
- **Out Queue**
  - Frame buffer/ queue/ port based aging
  - Bandwidth assured/ limited
  - Latency assured
  - WRED(Weight Random Early Drop)
  - Support LLQ (Low latency queue)
  - Dorm mode (WAN/LAN dual schedule mode)
- **Spanning Tree Protocol Port State**
  - Support Discard/Block/Learning/Forwarding four states
  - Forwarding STP frame to CPU port
  - Support RSTP/MSTP
- **Access Control List**
  - 128 ACL Entities
  - Ingress port
  - VLAN
  - Destination/Source MAC address
  - Destination/Source IP (specific or range)
  - TCP/UDP Destination/Source port number (specific or range)
  - IP protocol, DSCP, TCP flag
  - Action : forward, to CPU, drop, priority, Q-in-Q tag, remarking, redirect, bandwidth limited
- **Configuration**
  - Pin initial setting
  - 2-wire serial interface for accessing EEPROM
  - Advanced EEPROM program mode
  - 2-wire serial interface for low cost smart system application
- **Programmable serial LED Display Function**
  - 8 modes/ flashing speed selectable
  - Power on cable diagnosis indication
  - Loop indication
- **Auto Test Function For Mass-production**
  - Auto generate test frames
  - Show the result on LED output
- **Interrupt Pin For PHY Mode/Link/SMI R/W**
  - Complete notification
- **Special TAG**
- **OAM (IEEE 802.3 ah)**
  - Auto discovery
  - Fault indication
  - Remote LoopBack test
- **IEEE1588 stamp**
  - 16 time stamps for Ingress/Egress per port
  - 8 time stamps for input event trigger
  - Support Event trigger stamp
  - Support Pulse Per Second (PPS) output
  - Support PTP trigger out
- **IC+ Remote Management Protocol (IRMP)**
- **Build-in SRAM Self Test (BIST)**
- **sFlow**
- **EoC cable failure auto detection and isolation hardware**
- **L3/L2 Protocol packets forwarding to CPU or broadcast or drop**
  - LACP,LLDP,IGMP,MLD,ICMP,BPDU,802.1x, GARP
- **IPv6 function**
  - TCP/UDP port number and FLAG
  - Finding out next header is based on ICMPv6, authentication, encapsulation, fragment or user-defined header
- **Only need one 25MHz Crystal**
- **Adjustable IO voltage**
  - (2.5V~3.3V of RGMII/MII for port 10, 2.0V~3.3V of RGMII for port 9)
- **Built-in 2.0V regulator**
- **128 LQFP , 129 E-Pad Ground**

## General Description

IP1810I is a non-blocking, store-and-forward architecture switch controller, which builds 8-port 10/100Mbps Fast Ethernet MAC and PHY, 1-port GMAC with RGMII interface and 1-port with RGMII/MII for external CPU in a single chip.

IP1810I embeds a 4Mb SRAM for the use of packet buffer. It also provides various 2-wire interfaces, such as CPU interface, SMI, and EEPROM interface, which allow the user to access the internal register, external PHY's registers and EEPROM data. The serial LED can show the status of each port (such as Link, Speed, Activity and so on) by using 74HC164 or IP403 external device through 2-wire (LEDCLK, LEDDAT) signals driving from IP1810I.

For avoiding loop occur, IP1810I supports STP, RSTP and MSTP. Even a hardware loop detection mechanism is supported.

There are 16K entries in Lookup table. Hashing method can be selected either direct or CRC hashing for MAC address learning. Lookup Table aging time can be adjustable ranging from 55 seconds to 500 hours. IP1810I also provides the MAC learning threshold to limit the number of addresses learning.

An independent Multicast table supports 256 entries. For the Internet Group Management Protocol, IP1810I supports IGMP v1/v2/v3 for IPv4 and MLD v1/v2 for IPv6 packet snooping and aging time function. For the IGMPv3 and MLDv2 application, IP1810I provides up to 10 sets IPv4 or 4 sets IPv6 address per entry for source list table.

IP1810I has 4k entries for 802.1Q VLAN, which provides more security for VLAN member, and supports Independent VLAN Learning (IVL) and Share VLAN Learning (SVL). Further, IP1810I also has add/remove C-tag, S-tag, VLAN remarking, uplink link port and inactive VID redirect functions for the flexible VLAN application.

IP1810I has 128 entries for ACL table. The ACL (Address Control List) function provides advanced and flexible control mechanism to decide what packet should do (action) if packet matches ACL rule. ACL Rule and action:

- Rules: (Port ID, DA, SA, DA/SA, DIP, SIP, TCP/UDP Port Number (DP and SP) )
- Actions: (Sniffer, Insert/remove Tag, Priority, Redirection, Trap to CPU, BW limited, DSCP, sFlow, Drop, PTP, MIB Counter)

For the Quality of Service application, IP1810I supports port-based, 802.1Q tag based, ACL based, IP CoS/DSCP based, SMAC based, IP address based, IGMP based and WAN/LAN based to decide the priority of output frame. For the Remarking application, IP1810I supports remarking for the DSCP, Ctag and Stag value of transmit packet.

For the link aggregation application, IP1810I provides eight hashing methods (Port ID, DA, SA, DA/SA, DIP, SIP, DP, and SP) and support two trunk groups' sets.

For meeting the various network applications, IP1810I also supports storm control, bandwidth rate control, Output Queues functions, Sniffer, sFlow, PTP, MIB counters and eight LED display modes. Therefore, IP1810I has the best adaptability and capability for the future network conditions.

IP1810I supports the web based management interface, such as Microsoft Internet Explorer or Mozilla Firefox, Google chrome, user can more easily configure IP1810I's functions by these browsers. Besides, IP1810I also provides IRMP(IC+ remote management protocol) to let the users remotely configure the registers of IP1810I.

# Table of Contents

# List of Figures

# List of Tables

## Revision History

| Revision # | History Description | Date |
|---|---|---|
| IP1810I-DS-R01 | Initial release | 2016/10/19 |
| IP1810I-DS-R02 | Updated Power On Sequence of source power (page 165)<br>Added description of page C register 3[13][8] for port 10 MII PHY mode (page 159)<br>Updated block diagram of port 10 MII PHY mode (page 26) | 2017/01/20 |
| IP1810I-DS-R03 | Updated LED display behavior and sequence (page 30)<br>Updated power on sequence of 3.3V/1.15V (page 169)<br>Added 10BaseT selection (page 75) | 2017/07/18 |
| IP1810I-DS-R04 | Updated description of BGRES (page 17)<br>Updated power on sequence of 3.3V/1.15V (page 169) | 2018/04/10 |
| IP1810I-DS-R05 | Updated power on sequence (page 169)<br>Updated description of page 0xC register 0x3[14] (page 163)<br>Updated description of VDDIOR and VDDIOP (page 19)<br>Added description of Power on Setting (page 18,19)<br>Updated page 0x1 register 0x2 default value (page 93)<br>Updated diagram of RGMII timing (page 172) | 2018/06/04 |
| IP1810I-DS-R06 | Updated enable/disable of EEE AN ability (page 121)<br>Updated hold time of MDIO to 0ns (page 170) | 2018/07/31 |
| IP1810I-DS-R07 | Updated LED I/F PAD power from VDDIOP to VDDIOR (page 19)<br>Added description of external clock specifications and thermal data (page 176, 177) | 2018/09/20 |
| IP1810I-DS-R08 | Updated RESET voltage reference VDDIOP (page 173)<br>Added TXER/RXER interface to MII diagrom (page 26)<br>Updated behavior of start command and command complete (page 117, 121, 124, 165)<br>Updated Pull-down resistor to Pull-up/down resistor (page 173)<br>Updated enable/disable of EEE fuction of power on setting (page 18) | 2019/02/21 |
| IP1810I-DS-R09 | Added the parameters of power consumption (page 175)<br>Added EEPROM timing (page 173)<br>Updated power on sequence (page 168)<br>Added thermal data of IC Plus 2 layers PCB (page 177) | 2020/05/07 |
| IP1810I-DS-R10 | Removed Max. clock cycle time of CPU I/F (page 170)<br>Updated Group 0 description of Table 23 (page 42)<br>Updated default value of page 0x0 register 0xB5 (page 92)<br>Updated Max. of Supply Voltage (page 167)<br>Updated TWRR tickle unit (page 138)<br>Updated description of forwarded IGMPv3 packet (page 53)<br>Updated power on sequence and reset timing diagram (page 168)<br>Updated specification of Receiver common mode voltage (page 175) | 2020/09/25 |
| IP1810I-DS-R11 | Updated Crystal & X1 Clock Input Specification (page 176) | 2022/01/04 |

# 1 Pin Diagram

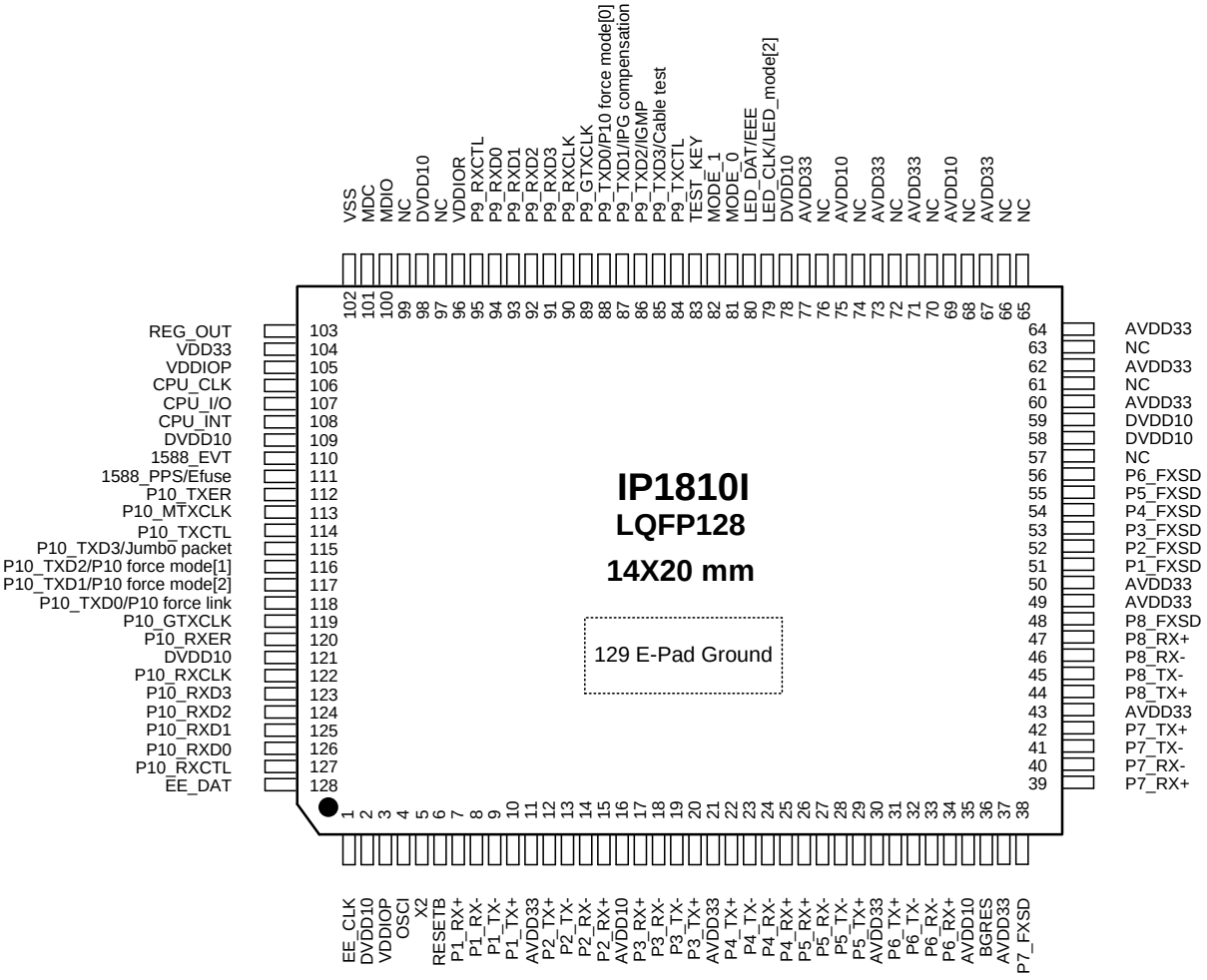



Figure 1 IP1810I 128 pin out Diagram

# 2 Block Diagram

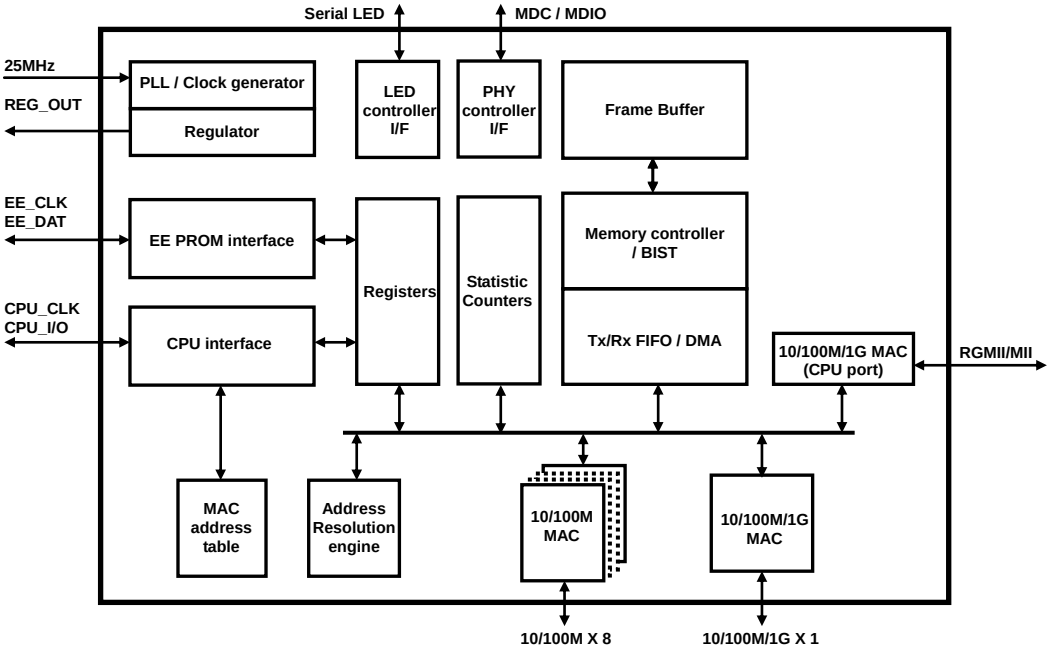



Figure 2 Block Diagram

# 3 Application Diagram

## 8 10/100M + 2G Combo Dumb Switch Application

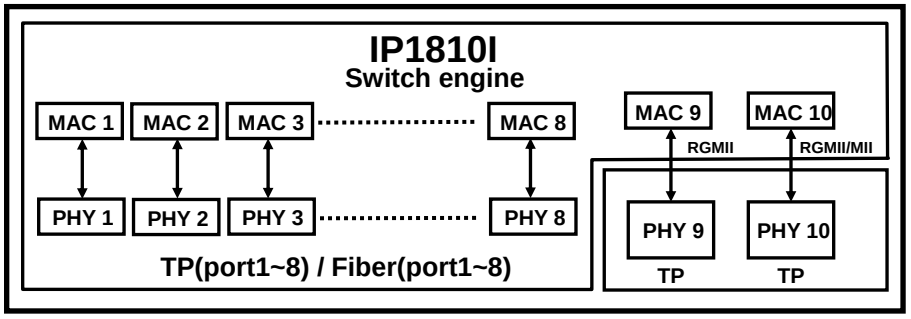



**Figure 3 Dumb Switch Application Diagram**

## 8 10/100M + 1G Combo Smart / Management Switch Application

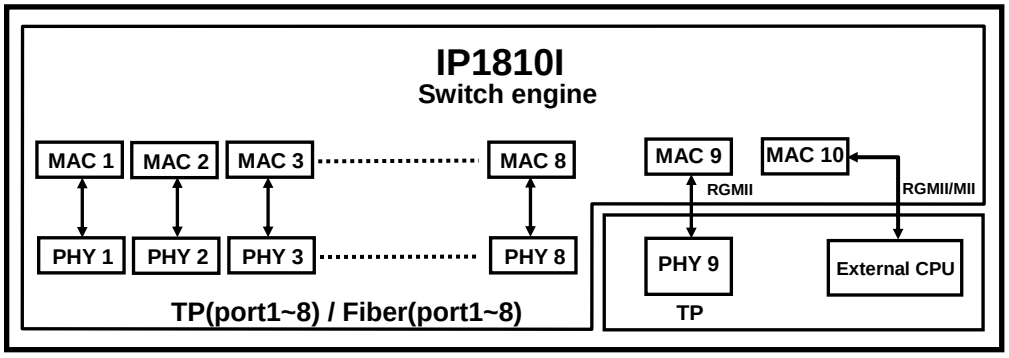



**Figure 4 Smart / Management Switch Application Diagram**

# 4 Pin Description

**Table 1 Pin Symbol Abbreviation**

| Type | Description | Type | Description |
|---|---|---|---|
| I | Input pin | IL | Input latched upon reset |
| O | Output pin | PD | Pulled down with internal resistor |
| I/O | Bi-direction input/output | PU | Pulled up with internal resistor |
| P | Power or ground | | |

**Table 2 MDI Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| **Medium Dependent Interface (port 1~8 ,PHY address 8~15)** | | | |
| 10<br>9<br>7<br>8 | P1_TX+<br>P1_TX-<br>P1_RX+<br>P1_RX- | O<br>O<br>I<br>I | **Port 1 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 1. Auto MDI/MDIX can reverse the pairs P1_TX+/P1_TX- and P1_RX+/P1_RX-. |
| 12<br>13<br>15<br>14 | P2_TX+<br>P2_TX-<br>P2_RX+<br>P2_RX- | O<br>O<br>I<br>I | **Port 2 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 2. Auto MDI/MDIX can reverse the pairs P2_TX+/P2_TX- and P2_RX+/P2_RX-. |
| 20<br>19<br>17<br>18 | P3_TX+<br>P3_TX-<br>P3_RX+<br>P3_RX- | O<br>O<br>I<br>I | **Port 3 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 3. Auto MDI/MDIX can reverse the pairs P3_TX+/P3_TX- and P3_RX+/P3_RX-. |
| 22<br>23<br>25<br>24 | P4_TX+<br>P4_TX-<br>P4_RX+<br>P4_RX- | O<br>O<br>I<br>I | **Port 4 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 4. Auto MDI/MDIX can reverse the pairs P4_TX+/P4_TX- and P4_RX+/P4_RX-. |
| 29<br>28<br>26<br>27 | P5_TX+<br>P5_TX-<br>P5_RX+<br>P5_RX- | O<br>O<br>I<br>I | **Port 5 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 5. Auto MDI/MDIX can reverse the pairs P5_TX+/P5_TX- and P5_RX+/P5_RX-. |
| 31<br>32<br>34<br>33 | P6_TX+<br>P6_TX-<br>P6_RX+<br>P6_RX- | O<br>O<br>I<br>I | **Port 6 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 6. Auto MDI/MDIX can reverse the pairs P6_TX+/P6_TX- and P6_RX+/P6_RX-. |
| 42<br>41<br>39<br>40 | P7_TX+<br>P7_TX-<br>P7_RX+<br>P7_RX- | O<br>O<br>I<br>I | **Port 7 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 7. Auto MDI/MDIX can reverse the pairs P7_TX+/P7_TX- and P7_RX+/P7_RX-. |

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 44<br>45<br>47<br>46 | P8_TX+<br>P8_TX-<br>P8_RX+<br>P8_RX- | O<br>O<br>I<br>I | **Port 8 Media Dependent Interface**<br><br>The differential data from the media is transmitted and received on two pairs for port 8. Auto MDI/MDIX can reverse the pairs P8_TX+/P8_TX- and P8_RX+/P8_RX-. |

| | | | |
|---|---|---|---|
| 51<br>52<br>53<br>54<br>55<br>56<br>38<br>48 | P1_FXSD<br>P2_FXSD<br>P3_FXSD<br>P4_FXSD<br>P5_FXSD<br>P6_FXSD<br>P7_FXSD<br>P8_FXSD | I | **Port 1~ Port 8 fiber signal detect**<br><br>The Fiber Signal Detect input voltage level and behavior are as below table :<br><br>**Voltage on Pxx_FXSD** / **TP port** / **Fiber port** / **Fiber signal detect** / **Condition**<br>< 0.4V / V / -- / -- / TP mode<br>> 1.2 V ; < 1.7 V / -- / V / Off / Fiber mode / unlink<br>> 1.95 V ; < 3.3 V / -- / V / On / Fiber mode / link |

**Table 3 RGMII/MII Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| **RGMII (port 9: PHY address 2,;port 10: PHY address 1)**<br>Port 9 I/O configuration by Page C register 8<br>Port 10 I/O configuration by Page C register A | | | |
| 85<br>86<br>87<br>88 | P9_TXD3<br>P9_TXD2<br>P9_TXD1<br>P9_TXD0 | O | **RGMII Transmit Data**<br><br>When it works in 10BASE-T or 100BASE-TX mode, TXD[3:0] present the transmit data at the rising edge of GTXCLK, and when it works in 1000BASE-T mode, TXD[3:0] present the low nibble of transmit data byte at the rising edge of GTXCLK, and present the high nibble of transmit data byte at the falling edge of GTXCLK. |
| 115<br>116<br>117<br>118 | P10_TXD3<br>P10_TXD2<br>P10_TXD1<br>P10_TXD0 | O | **RGMII/MII Transmit Data**<br><br>In RGMII mode, when it works in 10BASE-T or 100BASE-TX mode, TXD[3:0] present the transmit data at the rising edge of GTXCLK, and when it works in 1000BASE-T mode, TXD[3:0] present the low nibble of transmit data byte at the rising edge of GTXCLK, and present the high nibble of transmit data byte at the falling edge of GTXCLK.<br>In MII mode, TXD[3:0] present the transmit data at the rising edge of MTXCLK. |
| 84 | P9_TXCTL | O | **RGMII Transmit Enable**<br><br>RGMII transmit enable. |
| 114 | P10_TXCTL | O | **RGMII/MII Transmit Enable**<br><br>RGMII transmit enable in RGMII.<br>TXCTL is regarded as TXEN in MII. |

| | | | |
|---|---|---|---|
| 112 | P10_TXER | O | **MII Transmit Error**<br><br>MII transmit error in MII.<br>A “high” state present on this pin indicates transmit data error. It is synchronous to MTXCLK. It’s recommended to pull low in RGMII mode. |
| 113 | P10_MTXCLK | I/O | **MII Transmit Clock**<br><br>It is a 25MHz clock in 100BASE-TX mode, or a 2.5MHz clock in 10BASE-T mode. IP1810I uses the clock to sample TXER, TXEN and TXD[3:0]. |
| 89<br>119 | P9_GTXCLK<br>P10_GTXCLK | O | **RGMII Transmit Clock**<br><br>It is a 125MHz clock in 1000BASE-T mode, 25MHz clock in 100BASE-TX mode, or a 2.5 MHz clock in 10BASE-T mode. IP1810I uses the clock to sample TXCTL and TXD[3:0]. |
| 91<br>92<br>93<br>94 | P9_RXD3<br>P9_RXD2<br>P9_RXD1<br>P9_RXD0 | I | **RGMII Receive Data**<br><br>When the PHY works in 10BASE-T or 100BASE-TX mode, RXD[3:0] latch the receive data at the rising edge of RXCLK, and when it works in 1000BASE-T mode, RXD[3:0] latch the low nibble of receive data byte at the rising edge of RXCLK, and present the high nibble of transmit data byte at the falling edge of RXCLK. |
| 123<br>124<br>125<br>126 | P10_RXD3<br>P10_RXD2<br>P10_RXD1<br>P10_RXD0 | I | **RGMII/MII Receive Data**<br><br>In RGMII mode, when the PHY works in 10BASE-T or 100BASE-TX mode, RXD[3:0] latch the receive data at the rising edge of RXCLK, and when it works in 1000BASE-T mode, RXD[3:0] latch the low nibble of receive data byte at the rising edge of RXCLK, and present the high nibble of transmit data byte at the falling edge of RXCLK.<br>In MII mode, RXD[3:0] latch the receive data at the rising edge of RXCLK. |
| 120 | P10_RXER | I | **MII Receive Error**<br><br>MII receive error in MII.<br>A “high” state present on this pin indicates received data error. It is synchronous to RXCLK. It’s recommended to pull low in RGMII mode. |
| 95 | P9_RXCTL | I | **RGMII Receive Valid**<br><br>RGMII receive data valid. |
| 127 | P10_RXCTL | I | **RGMII/MII Receive Valid**<br><br>RGMII receive data vaild in RGMII.<br>RXCTL is regarded as RXDV in MII. |
| 90 | P9_RXCLK | I | **RGMII Receive Clock**<br><br>It is a 125 MHz, 25MHz, or 2.5MHz reference clock from a Gigabit PHY working in 1000BASE-T, 100BASE-TX, or 10BASE-T. |

| | | | |
|---|---|---|---|
| 122 | P10_RXCLK | I/O | **RGMII/MII Receive Clock**<br><br>It is a 125 MHz, 25MHz, or 2.5MHz reference clock from a Gigabit PHY working in 1000BASE-T, 100BASE-TX, or 10BASE-T. |

**Table 4 MDC/MDIO Pin**

| | | | |
|---|---|---|---|
| 101 | MDC | O | **Serial management bus clock output**<br><br>It's recommended to add a 30pF capacitor to ground for noise filtering. |
| 100 | MDIO | I/O | **Serial management bus data input/output**<br><br>It's recommended to add a 1.5K ohm pull up resistor connecting to 3.3V VCC and a 30pF capacitor connecting to ground. |

**Table 5 EEPROM Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 1 | EE_CLK | O | **Serial EEPROM clock output** |
| 128 | EE_DAT | I/O | **Serial EEPROM data input/output** |

**Table 6 CPU R/W Interface Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 106 | CPU_CLK | I | **Serial CPU access clock input** |
| 107 | CPU_I/O | I/O | **Serial CPU data input/output** |
| 108 | CPU_INT | O | **CPU interrupt** |

**Table 7 Serial LED Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 79 | LED_CLK | O | **Serial LED clock output**<br><br>LED clock rate setting by Page 3 register 0x17 bit[7:6] :<br>00 : 781 KHz<br>01 : 2.5 MHz<br>10 : 5.2 MHz<br>11 : 10.4 MHz (default) |
| 80 | LED_DAT | O | **Serial LED data output** |

**Table 8 IEEE-1588 PTP Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 110 | 1588_EVT | I | **1588 Event**<br><br>Event trigger source input |
| 111 | 1588_PPS | O | **1588 PPS**<br><br>Pulse per second reference output or clock/trigger output, refer to page 0x9 register 0x02 |

**Table 9 Miscellaneous Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 4 | OSCI | I | **Crystal/Oscillator 25MHz input** |
| 5 | X2 | O | **Crystal output** |
| 6 | RESETB | I | **System reset (low active)**<br><br>Should be kept at “low” for at least 10 microseconds. The input voltage should be not higher than VDD33 |
| 83 | TEST_KEY | I | **Auto Factory Test enable key input** |
| 81<br>82 | MODE_0<br>MODE_1 | I, PD | **Test mode select**<br><br>00: Normal (default)<br>Reserved for test, should be “00” for normal operation |
| 36 | BGRES | I | **Band gap reference voltage external resistor**<br><br>It must be connected to ground through a 6.19K resistor. |
| 57,61,63,<br>65,66,68,<br>70,72,74,<br>76,97,99 | NC | | **No connect** |

## Power on Setting

The state of these pins will be latched upon reset.

**Table 10 Power on Setting**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 79 | LED_CLK/<br>**LED_mode[2]** | IL, PU | **LED mode [2]**<br><br>1 : 2-bit mono-color (Giga-Spd, Link/Act) (100-Spd, Link/Act)<br>0 : 2-bit bi-color (operate at GE port: Spd and Link/Act) and<br>1-bit mono-color (operate at FE port: Link/Act)<br>The setting can be updated by writing page 0x3 register 0x17[2]. |
| 80 | LED_DAT/**EEE** | IL, PU | **Polling PHY MMD register for EEE function**<br><br>0: EEE disable<br>1: EEE enable<br>The setting can be updated by writing page 0x3 register 0x2F[7:0] and 0x30[12][8]. |
| 88 | P9_TXD0/<br>**P10 force mode[0]** | IL, PD | **P10 force mode [0]** (please refer to **P10 force mode [1]** and **P10 force mode [2]** )<br><br>P10 force mode [2:0]<br>000 : p10 operation mode is not forced<br>001 : p10 operation mode is not forced<br>010 : 1000Mb/ Full<br>011 : Reserved<br>100 : 100Mb / Full<br>101 : 100Mb / Half<br>110 : 10Mb/ Full<br>111 : 10Mb/ Half |
| 87 | P9_TXD1/<br>**IPG compensation** | IL, PD | **IPG compensation**<br><br>0: IPG compensation 40ppm<br>1: IPG compensation 160ppm<br>The setting can be updated by writing page 0x0 register 0x1[1]. |
| 86 | P9_TXD2/ **IGMP EN** | IL, PD | **IGMP enable**<br><br>0: IGMP snooping disable<br>1: IGMP snooping enable<br>The setting can be updated by writing page 0x1 register 0x29[0]. |
| 85 | P9_TXD3/<br>**Initial cable test** | IL, PD | **Intial cable test**<br><br>0: Initial cable test disable<br>1: Initial cable test enable<br>The setting can be updated by writing page 0x3 register 0x16[14]. |
| 118 | P10_TXD0/<br>**P10 force link** | IL, PD | **P10 force link**<br><br>0: P10 force link disable<br>1: P10 force link enable<br>The setting can be updated by writing page 0x3 register 0x12[0]. |

| | | | |
|---|---|---|---|
| 117 | P10_TXD1/<br>**P10 force mode [2]** | IL, PD | **P10 force mode [2]**<br><br>(please refer to **p10 force mode [0]** description) |
| 116 | P10_TXD2/<br>**P10 force mode [1]** | IL, PD | **P10 force mode [1]**<br><br>(please refer to **P10 force mode [0]** description) |
| 115 | P10_TXD3/<br>**Jumbo packet** | IL, PD | **Jumbo packet**<br><br>0 : Jumbo packet disable<br>1 : Jumbo packet enable<br>The setting can be updated by writing page 0x0 register 0x2[7:0] and 0x3[12][8]. |

**Table 11 Power & Ground Pin**

| Pin No. | Label | Type | Description |
|---|---|---|---|
| 2,58, 59,78, 98,109,121 | DVDD10 | P | **1.15V power for core circuit** |
| 16,35, 69,75 | AVDD10 | P | **1.15V power for analog circuit** |
| 103 | REG_OUT | P | **2.0V regulator output for RGMII I/O** |
| 96 | VDDIOR | P | **2.0V/2.5V/3.3V input for LED I/F and port 9 I/O PAD power** |
| 102 | VSS | P | **Digital power ground for regulator** |
| 104 | VDD33 | P | **3.3V power for digital circuit** |
| 3,105 | VDDIOP | P | **2.5V/3.3V input for EEPROM, CPU I/F, RESET and port 10 I/O PAD power** |
| 11,21,30,37, 43,49,50,60, 62,64,67,71, 73,77 | AVDD33 | P | **3.3V power for analog circuit** |
| 129 | E-Pad GND | P | **Digital power ground** |

I/O Power configuration refer Table 13

# 5 Function Description

## 5.1 Switch Engine and Queue Management

### 5.1.1 Packet forwarding

IP1810I utilizes the “store & forward” method to handle packet transfer. IP1810I begins to forward a packet to a destination port after the entire packet is received. A received packet will be forwarded to the destination port only if it is error free; otherwise, it will be discarded.

#### 5.1.1.1 Address Learning and Hashing

| Related registers | 0x02[3:0], 0x03[7:0], 0x04[12][8] | PAGE 0x01 |
|---|---|---|

IP1810I has 16384 entries for MAC address learning tables. It provides two hashing methods to access the MAC address table. One is the direct mapping and the other is the CRC algorithm. For the direct mapping method, if page 0x01 register 0x02[3] is set to “0”, the MAC address directly employs the least significant 13 bits of the MAC address. For the CRC algorithm method, if page 0x01 register 0x02[3] is set to “1”, IP1810I uses CRC algorithm to obtain 13bits MAC address from the 48-bits MAC.

The MAC address learning function can be either enabled or disabled for each port by programming page 0x01 register 0x03[7:0] and 0x04[12][8], and MAC address with all 0s can be learned to MAC address table by programming page 0x01 register 0x02[2] if the MAC address learning function is enabled.

IP1810I provides three kinds of methods for the MAC address learning. The first method is enabled by programming page 0x01 register 0x02[1:0] to “00”, means LUT can be written even when the entry is valid. The second method is enabled by programming page 0x01 register 0x02[1:0] to “01 or 10”, means LUT can not be written before aging out. The third method is enabled by programming page 0x01 register 0x02[1:0] to “11”, means only the second layer LUT can be written before aging out,.

The packet with the following conditions will not be learned in MAC address table.

- Erroneous packet
- 802.3x pause packet(option)
- 802.1D Reserved Group packet(option)
- Multicast source MAC address

IP1810I provides the MAC address learning count threshold function to limit source MAC address number learning in MAC address table. The number of learning MAC address must not be more than threshold, the extra MAC address will not be learnt in MAC address table if learning MAC address number higher than the threshold. The threshold values can be set by programming page 0x01 register 0x07[8:0] and each port is enabled individually by programming the corresponding bits of page 0x01 register 0x05[7:0] and 0x06[12][8].

#### 5.1.1.2 Port Based Address Flush

| Related registers | 0x1A[7:0], 0x1B[13:12][8] | PAGE 0x01 |
|---|---|---|

IP1810I can clear the MAC address that bind to specific port. The specific port can be enabled by programming the corresponding bits of page 0x01 register 0x1A[7:0] and 0x1B[12][8] to “1”, and trigger the port based address flush function by programming page 0x01 register 0x1B[13] to “1”.

#### 5.1.1.3 Aging Time

Address Aging

| Related registers | 0x01[15], 0x01[14:0] | PAGE 0x01 |
|---|---|---|

IP1810I supports programmable aging time to meet various system requirements, ranging from 55 sec to 1802295 sec ±3.8%. The user can set aging time by programming page 0x01 register 0x01[14:0]. The address aging function can be also disabled by programming page 0x01 register 0x01[15] to "1".

Packet Aging

| Related registers | 0x66[7:0], 0x67 to 0x6A, 0x73, 0x75 | PAGE 0x08 |
|---|---|---|

IP1810I supports packet aging (out queue aging). (If a packet stays in Queue too long, IP1810I will age out according to aging time defined in page 0x08 register 0x66[7:0]). IP1810I will drop the packet to improve the efficiency of packet buffer. The packet aging function can be enabled individually for each port by programming page 0x08 register 0x67 to 0x6A, 0x73 and 0x75.

#### 5.1.1.4 802.1D packet forwarding

| Related registers | 0x06[15:0], 0x07[4:0] | PAGE 0x00 |
|---|---|---|
| | 0x03[0] | PAGE 0x0C |

Except that the erroneous packet and the IEEE802.3x pause packet will not be forwarded, 802.1D Reserved Group packet with MAC address from 01-80-c2-00-00-00 to 01-80-c2-00-00-3F can be forwarded, dropped or "To CPU port" by programming page 0x00 register 0x06[15:0] and 0x07[3:0]. If the corresponding bits of page 0x00 register 0x06[15:0] and 0x07[3:0] are set to "11", the 802.1D packets for each port will be dropped. Under this situation, if page 0x00 register 0x07[4] is also set to "1", the 802.1D packet coming from CPU port can be forwarded.

Inter Frame Gap Compensation

| Related registers | 0x01[1:0] | PAGE 0x00 |
|---|---|---|

IP1810I supports an option to transmit a packet with IPG shrank 40 ppm, 80 ppm or 160 ppm to compensation the data accumulation due to TX clock frequency difference between local machine and link partner. Programming page 0x00 register 0x01[1:0] can enable this function.

### 5.1.2 Flow Control

IP1810I supports two kinds of flow control mechanisms, backpressure for half duplex operation and IEEE 802.3x for full duplex operation.

#### 5.1.2.1 IEEE802.3x

| Related registers | 0x08[7:0] to 0x0B[12][8] | PAGE 0x03 |
|---|---|---|

When operating in full duplex mode, IP1810I supports IEEE802.3x flow control, both symmetric pause and asymmetric pause function. Each port's flow control function can be enabled individually by programming page 0x03 register 0x08[7:0] to 0x0B[12][8]. When the packets in buffer reach the threshold, IP1810I generates a "Xoff" pause packet immediately or right after the current packet has been transmitted. When receiving a pause packet, the link partner stops transmission for a period of time defined in the pause packet. This prevents the buffer of IP1810I from overrun. When the packets in buffer lower than threshold, IP1810I generates a "Xon" pause packet to notify the link partner the receive buffer is available.

#### 5.1.2.2 Backpressure

| Related registers | 0x01[3:2] | PAGE 0x00 |
|---|---|---|
| | 0x0C[7:0], 0x0D[12][8] | PAGE 0x03 |

When operating in half duplex mode, IP1810I supports backpressure flow control. Each port's backpressure function can be enabled individually by programming page 0x03 register 0x0C[7:0] and 0x0D[12][8]. When the packets in buffer reach the threshold, IP1810I generates a jam pattern to back off the link partner. IP1810I supports the collision based and carrier-based backpressure. When the collision based backpressure is enabled, page 0x00 register 0x01[2] set to "1", IP1810I generates a jam pattern only when the link partner is transmitting data and the receive buffer in IP1810I is not available. When detecting a collision on line, the link partner stops transmission until a back off time expires. When the carrier based backpressure is enabled, page 0x00 register 0x01[2] set to "0", IP1810I transmits null packets continuously to prevent link partner's transmission when the buffer is not available.

To prevent the packet loss due to excessive collision caused by backpressure mechanism, user can program page 0x00 register 0x01[3] to "0" to disable the drop function due to 16 consecutive collisions defined in IEEE802.3.

#### 5.1.2.3 Flow control auto off for high priority packet

| Related registers | 0x01[5] | PAGE 0x00 |
|---|---|---|

To prevent the flow control function from blocking the high priority traffic, each port of IP1810I can auto turn off flow control function for a period of time automatically when receiving a high priority packet. This function can be enabled by programming page 0x00 register 0x01[5] to "1".

### 5.1.3 Bandwidth Control

| | | |
|---|---|---|
| **Related registers** | **0x87 to 0x8E, 0x9F, 0xA3 (Ingress)** | **PAGE 0x00** |
| | **0x85 to 0x8C, 0x9D, 0xA1 (Egress), 0xB1[6:4]** | **PAGE 0x08** |

IP1810I implements a sophisticated data rate control mechanism, which is very useful for the bandwidth-limited network. By controlling both the ingress and the egress data rate, IP1810I provides a variety of bandwidth configurations. It limits the maximum byte counts in one second, for each port to send or receive packets. If the transmit byte counter or receive byte counter of a port reaches a pre-defined threshold, it will stop transmitting or receiving data in one second.

Each port's Ingress/egress data rate can be programmable individually. The bandwidth rate for port 1 to port 10 is defined in page 0x00 register 0x87 to 0x8E, 0x9F and 0xA3(Ingress) and page 0x08 register 0x85 to 0x8C, 0x9D and 0xA1(Egress). The egress bandwidth parameter is defined in page 0x08 register 0xB1[6:4]. Note that its bandwidth should not be more than the link speed of a port.

### 5.1.4 Broadcast Storm Control

| | | |
|---|---|---|
| **Related registers** | **0x0D[9:0] to 0x0F[12][8]** | **PAGE 0x01** |

To prevent the broadcast storm, the IP1810I implements a broadcast storm control mechanism. By enabling this function, a port begins to drop the incoming broadcast packets if the number of received broadcast packets reach the threshold in selected period, which is defined in page 0x01 register 0x0D[7:0] and 0x0D[9:8]. Each port's broadcast storm protection function can be enabled individually by programming page 0x01 register 0x0E[7:0] and 0x0F[12][8]. IP1810I also provides four kinds of time period, which can be selected by programming page 0x01 register 0x0D[9:8] for 1G/100M/10M. The detailed configuration is shown in the following table.

#### 5.1.4.1 Multicast and DLF Storm Control

| | | |
|---|---|---|
| **Related registers** | **0x0D[9:0], 0x10[7:0] to 0x13[12][8]** | **PAGE 0x01** |

Multicast storm and DLF storm use the same threshold setting defined in page 0x01 register 0x0D[7:0] . Multicast means the bit[40] of the destination MAC address is set to "1". DLF(Destination Lookup Failure) means the MAC address doesn't exist in MAC address table, is regarded as broadcast.
multicast storm is configured by programming page 0x01 register 0x10[7:0] and 0x11[12][8] and DLF storm is configured by programming page 0x01 register 0x12[7:0] and 0x13[12][8].

#### 5.1.4.2 ARP Storm Control

| | | |
|---|---|---|
| **Related registers** | **0x14[9:0] to 0x16[12][8]** | **PAGE 0x01** |

To prevent the ARP storm, IP1810I implements an ARP storm control mechanism. By enabling this function, a port begins to drop the incoming ARP packets if the number of received ARP packets reach the threshold in selected period, which is defined in page 0x01 register 0x14[7:0] and 0x14[9:8].

Each port's ARP storm protection function can be enabled individually by programming page 0x01 register 0x15[7:0] and 0x16[12][8]. IP1810I also provides four kinds of time period, which can be selected by programming page 0x01 register 0x14[9:8] for 1G/100M/10M. The detailed configuration is shown in the following table.

#### 5.1.4.3 ICMP Storm Control

| Related registers | 0x17[9:0] to 0x19[12][8] | PAGE 0x01 |
|---|---|---|

To prevent the ICMP storm, IP1810I implements an ICMP storm control mechanism. By enabling this function, a port begins to drop the incoming ICMP packets if the number of received ICMP packets reach the threshold in selected period, which is defined in page 0x01 register 0x17[7:0] and 0x17[9:8].

Each port's ICMP storm protection function can be enabled individually by programming page 0x01 register 0x18[7:0] and 0x19[12][8]. IP1810I also provides four kinds of time period, which can be selected by programming page 0x01 register 0x17[9:8] for 1G/100M/10M. The detailed configuration is shown in the following table.

**Table 12 Broadcast /ARP/ICMP Storm Counter Clear Period Selection Table**

| 0x0D[9:8]<br>0x14[9:8]<br>0x17[9:8] | 1000Mbps | 100Mbps | 10Mbps |
|---|---|---|---|
| 00 | 200us | 2ms | 20ms |
| 01 | 1ms | 10ms | 100ms |
| 10 | 10ms | 10ms | 10ms |
| 11 | 100ms | 100ms | 100ms |

### 5.1.5 Interface

#### 5.1.5.1 MDI (port 1 ~ port 8)

IP1810I builds up 8 100M/10Mb internal PHYs, which individually connect with each TP port by TX/RX differential signals, ranging from port 1 to port 8.

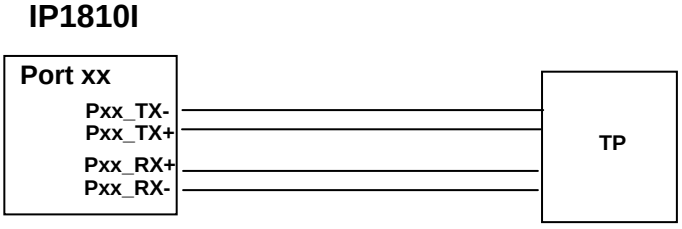



Figure 5 MDI Application diagram

#### 5.1.5.2 RGMII(port 9 ~ port 10) / MII(port 10)

IP1810I sends out data TXD[3:0] and control signal TXCTL at the rising and falling edge of GTXCLK. Two GMII like signals TXEN and TXER are embedded in the TXCTL. GMII like information TXD[7:0] is embedded in the TXD[3:0]. By recognizing the decoded TXEN, TXD[7:0] and TXER, a PHY can capture the correct data stream.

A PHY sends out data RXD[3:0] and control signal RXCTL at the rising and falling edge of RXCLK. Two GMII like signals RXDV and RXER are embedded in the RXCTL. GMII like information RXD[7:0] is embedded in the RXD[3:0]. By recognizing the decoded RXDV, RXD[7:0] and RXER, IP1810I can capture the correct data stream. IP1810I samples the correct data at the rising edge of RXCLK.

To fit the timing requirement, the delay on GTXCLK and RXCLK can be adjusted by programming the bit[5:0] of page 0x0C register 0x08 and 0x0A. The driving current can be also adjusted by programming the bit[8:6] of page 0x0C register 0x08 and 0x0A.

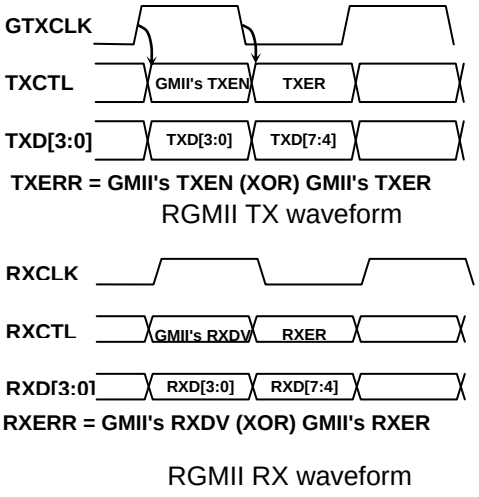



**Figure 6 Waveform of RGMII**

IP1810I's port 10 uses RGMII/MII interface to connect with external PHY

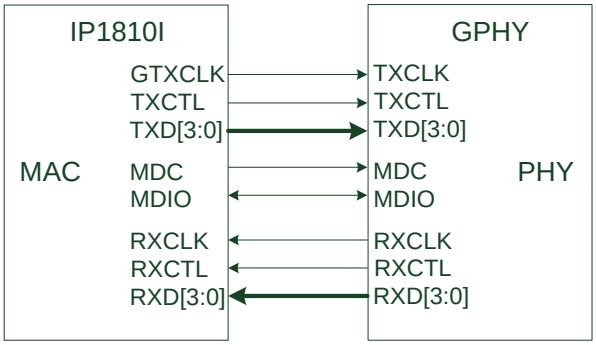



**Figure 7 RGMII Application diagram**

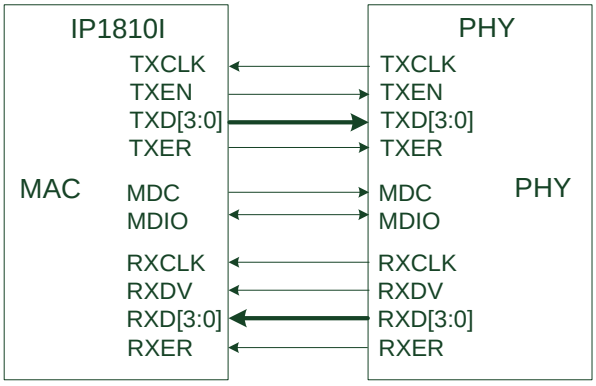



**Figure 8 MII Application diagram**

IP1810I's port 10 uses RGMII/MII interface to connect with external CPU.

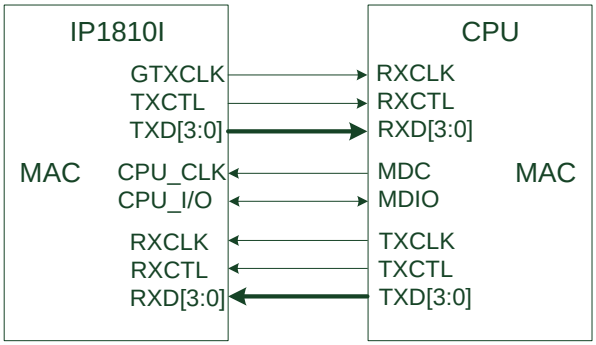



**Figure 9 RGMII Application diagram**

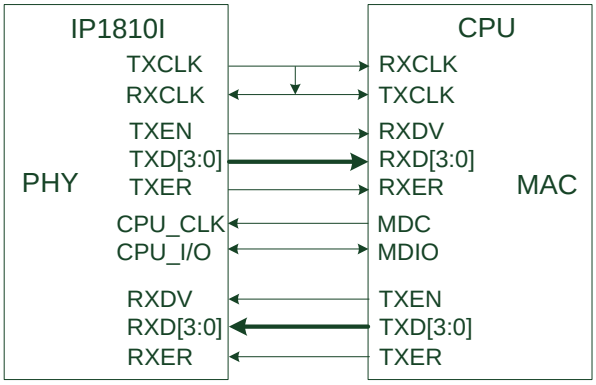



**Figure 10 MII Application diagram**

**Table 13 I/O Power Configuration Table**

| I/O Power \ Ports & I/O | | Port 9 | | Port 10 | | Other I/Os | | |
|---|---|---|---|---|---|---|---|---|
| VDDIOR | VDDIOP | RGMII | SMI | RGMII | MII | CPUx | EEPROM | LEDx |
| 2.0V | 3.3V | 2.0V | 2.0V | 3.3V | 3.3V | 3.3V | 3.3V | 2.0V |
| 2.5V | 3.3V | 2.5V | 2.5V | 3.3V | 3.3V | 3.3V | 3.3V | 2.5V |
| 2.5V | 2.5V | 2.5V | 2.5V | 2.5V | 2.5V | 2.5V | 2.5V | 2.5V |
| 3.3V | 2.5V | 3.3V | 3.3V | 2.5V | 2.5V | 2.5V | 2.5V | 3.3V |
| 3.3V | 3.3V | 3.3V | 3.3V | 3.3V | 3.3V | 3.3V | 3.3V | 3.3V |

### 5.1.6 External CPU Read / Write interface

There is no need to program the register of the IP1810I for the generic application. However it's probably necessary to program the internal register to fit some special applications. The interface between the IP1810I and the external CPU is a serial bus, which comprises a clock and an I/O signal. Like the access cycle of the serial management interface, the serial interface comprises the switch ID, the read/write command, the address and the data. The access cycle is depicted as below.

The access cycle is much like the access cycle of MDC, MDIO. Care should be taken that the switch ID is 2-bit wide rather than 5-bit wide.

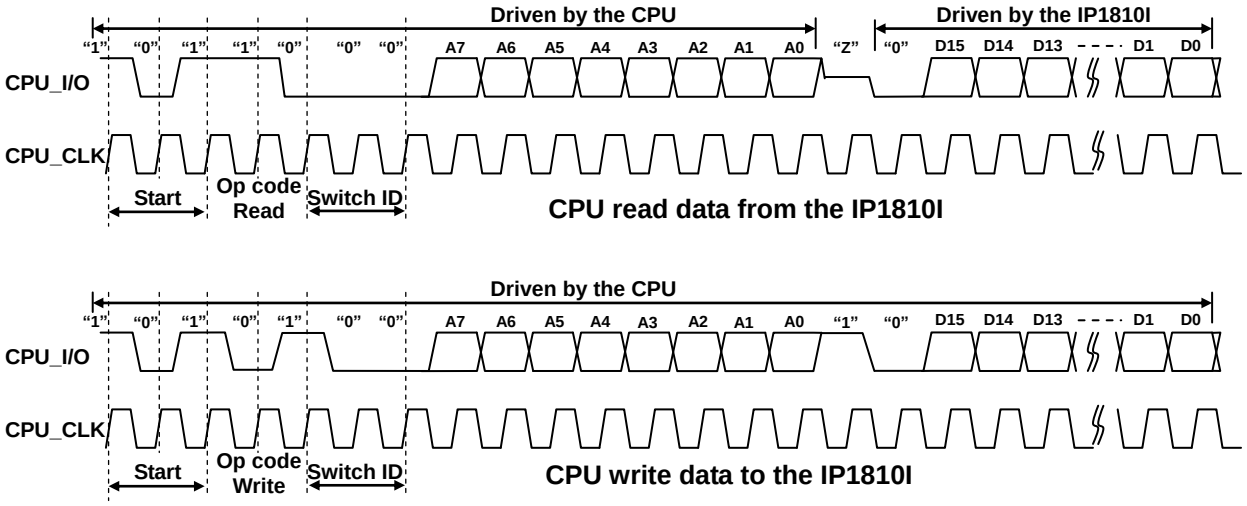



**Figure 11 External CPU Read / Write interface diagram**

### 5.1.7 EEPROM Interface

| Related registers | 0x01 to 0x0F | PAGE 0x0B |
|---|---|---|

IP1810I supports EEPROM I/F to access 24C01 to 24C16. After detecting the rising edge of a reset input, IP1810I will start to read the content of EEPROM (acting like an EEPROM master). Being an EEPROM master, IP1810I provides command mode to download the content of EEPROM. The content of the first byte should be “00h” to be identified as IP1810I EEPROM. If the content of the second byte is “0Bh” for command mode of 24C01 to 24C16 EEPROM capacity.

In the command mode, one command occupies 4 EEPROM bytes. The content includes 1 bytes command and 1 bytes register address and 2 bytes data. If command byte < 0x40, it stands the register page address. If command byte >=0x40, it stands for the advance commands. (The detail explanation for advance commands please refer to appendix application notes)

The following is a normal EEPROM example for IP1810I.

**Table 14 Command Mode Format of EEPROM**

| Command mode | | |
|---|---|---|
| **EEPROM address** | **EEPROM content** | **Description** |
| 0x00 | 0x00 | EEPROM start |
| 0x01 | 0x0B | ( EEPROM capacity at Command Mode) |
| 0x02 | 0x00 | Don’t care |
| 0x03 | 0x00 | Don’t care |
| 0x04 | 0x03 | Page address or advance Command<br>In this case, page address is 0x03 |
| 0x05 | 0x0A | Register address<br>In this case, register address is 0x0A |
| 0x06 | 0x34 | bit[15:8] Register data<br>In this case, write 0x34 to register 0x0A[15:8] |
| 0x07 | 0x12 | bit[7:0] Register data<br>In this case, write 0x12 to register 0x0A[7:0] |
| 0x08 | 0xXX | Page address or advance Command |
| 0x09 | 0xXX | Register address |
| 0x0a | 0xXX | bit[15:8] Register data |
| 0x0b | 0xXX | bit[7:0] Register data |
| 0x0c | 0xXX | Page address or advance Command |
| 0x0d | 0xXX | Register address |
| 0x0e | 0xXX | bit[15:8] Register data |
| 0x0f | 0xXX | bit[7:0] Register data |
| ⋮ | | |
| XX | 0xFF (or 0x00) | Page address or advance Command |
| XX | 0xFF (or 0x00) | Register address |
| XX | (Do not care) | Stop Initial Reading (Follow by the command byte and register address are both 0xFF) |
| ⋮ | (Do not care) | |

### 5.1.8 MIB Counter

| Related registers | 0x01[9], 0x64 to 0x66 | PAGE 0x00 |
|---|---|---|

IP1810I provides a set of RX and TX counters for statistic, IP1810I can monitor a variety of packets, including different length packet, illegal packet(over size, symbol error), unicast packet, multicast packet broadcast packet, and even byte count for packet, etc. To enable the counter, page 0x00 register 0x01[9] must set to “1”. The user can read the content of the counter through page 0x00 register 0x64~0x66. The detailed description is shown in the following table.

**Table 15 Address Format of Port MIB Counter**

| address | RX counter | TX counter |
|---|---|---|
| 00 | RX_64B | TX_64B |
| 01 | RX_65_127B | TX_65_127B |
| 02 | RX_128_255B | TX_128_255B |
| 03 | RX_256_511B | TX_256_511B |
| 04 | RX_512_1023B | TX_512_1023B |
| 05 | RX_1024_1518 | TX_1024_1518 |
| 06 | RX_Oversize | TX_Oversize |
| 07 | RX_Bcst | TX_Bcst |
| 08 | RX_Mcst | TX_Mcst |
| 09 | RX_Ucst | TX_Ucst |
| 0A | RX_Pause | TX_Pause |
| 0B | RX_Pkt | TX_Pkt |
| 0C | Byte_L | Bytes_L |
| 0D | Byte_H | Bytes_H |
| 0E | RX_Drop (buff) | TX_Drop (buff) |
| 0F | RX_DROP others | Single Col |
| 10 | RX_CRC | Multiple Col |
| 11 | RX_Alignment | Late Col |
| 12 | RX_Runt | Defered TX |
| 13 | RX_Frag | Excessive Col |
| 14 | RX_Jabber | |
| 15 | Symbol Error | |
| 16 | ACL | |
| 17 | ACL | |

### 5.1.9 Auto Factory Test (AFT)

IP1810I implements an AFT mechanism, which is very useful for switch pre-test. When this function is triggered by hardware pin 83 TEST_KEY, IP1810I’s physical ports are connected either port-pairs LoopBack or self-LoopBack and then IP1810I will generator frames for TX of all ports .

In the AFT mode, the light-on period is within 1 second, then show the test result on the LED. The status of test result will keep the LED status while this hardware pin AFT is still triggered.

For example, port 1 and port 2 are connected to each other (LoopBack). Check the LED of port 1 and port 2 is light-on, and then trigger this AFT function. If the test result is passed (no CRC or packet loss), show LED of port 1 and port 2 light-on. Otherwise, show LED light-off when CRC or packet loss is happened.

### 5.1.10 LED display mode

| Related registers | 0X17 to 0X1F[12][8] | PAGE 0x03 |
|---|---|---|

The bit stream is output sequentially through LED_DAT and LED_CLK and its sequence starts from port 10 to port 1. In the other word, port 1 LED status is present on the latest LED bit, as shown in the table. To store the serial LED stream, a serial-to-parallel shift register should be used.

**Table 16 LED display behavior vs. LED mode**

| Pin Setting | 0x17 bit[2:0] | LED mode | LED display behavior | |
|---|---|---|---|---|
| | | | Giga port | 10/100M port |
| - | 000 | 3-bit M | Giga-Spd, L/A, Dupx/Col | 100-Spd, L/A, Dupx/Col |
| - | 001 | 3-bit M | Giga-L/A, 100/10-L/A, Dupx/Col | 100-L/A, 10-L/A, Dupx/Col |
| - | 010 | 3-bit M | Giga-Spd, 100/10-Spd, /A | 100-Spd, 10-Spd, /A |
| 1 | 011 | 2-bit M | Giga-Spd, L/A | 100-Spd, L/A |
| - | 100 | 2-bit B, 1-bit M | Spd & L/A, PSE | Spd1 & L/A, PSE |
| - | 101 | 2-bit M | L, /A | |
| - | 110 | 2-bit B | Spd & L/A | Spd1 & L/A |
| 0 | 111 | 2-bit B 1-bit M | Spd & L/A | L/A |

( B : bi-color; M : mono-color)
( Spd & L/A : 1G & 100M/10M bi-color; Spd1 & L/A : 100M & 10M bi-color)

**Table 17 LED display Sequence**

| 0x17 bit[2:0] | LED mode | LED Display Sequence. [1], [2]..... indicate the bit stream sequence shifted from LED_DAT. |
|---|---|---|
| 000 | 3-bit M | [1] port 10 Giga-Spd, [2] port 10 L/A, [3] port 10 Dupx/Col, …….. ……., [28] port 1 100-Spd, [29] port 1 L/A, [30] port 1 Dupx/Col(last one) |
| 001 | 3-bit M | [1] port 10 Giga-L/A, [2] port 10 100/10-L/A, [3] port 10 Dupx/Col, …… ……., [28] port 1 100-L/A, [29] port 1 10-L/A, [30] port 1 Dupx/Col(last one) |
| 010 | 3-bit M | [1] port 10 Giga-Spd, [2] port 10 100/10-Spd, [3] port 10 Active, ……. ……., [28] port 1 100-Spd, [29] port 1 10-Spd, [30] port 1 Active(last one) |
| 011 | 2-bit M | [1] port 10 Giga-Spd, [2] port 10 L/A, [3] port 9 Giga-Spd, [4] port 9 L/A, ……. ……., [19] port 1 100-Spd, [20] port 1 L/A(last one) |
| 100 | 2-bit B, 1-bit M | [1][2] port 10 Spd & L/A, [3] port 10 PSE, [4][5] port 9 Spd & L/A, [6] port 9 PSE, ……., [28][29] port 1 Spd1 & L/A, [30] port 1 PSE(last one) |
| 101 | 2-bit M | [1] port 10 Link, [2] port 10 Active, [3] port 9 Link, [4] port 9 Active, ……. ……., [19] port 1 Link, [20] port 1 Active(last one) |
| 110 | 2-bit B | [1][2] port 10 Spd & L/A, [3][4] port 9 Spd & L/A, ……. ……., [19][20] port 1 Spd1 & L/A (last one) |
| 111 | 2-bit B 1-bit M | [1][2] port 10 Spd & L/A, [3][4] port 9 Spd & L/A, [5] port 8 L/A, ……. ……., [12]port 1 L/A(last one) |

IP1810I also provides CPU control LED mode, if page 0x03 register 0x17[15] is asserted, then LED will show results according to the settings of page 0x03 register 0x1A[7:0] to 0x1F[12][8].
More detail description, please refer to the LED application note.

Bi-color description


An example for port 1 ~ port 10 using 2-bit stream mono-color LED display mode with IP403 is shown below.

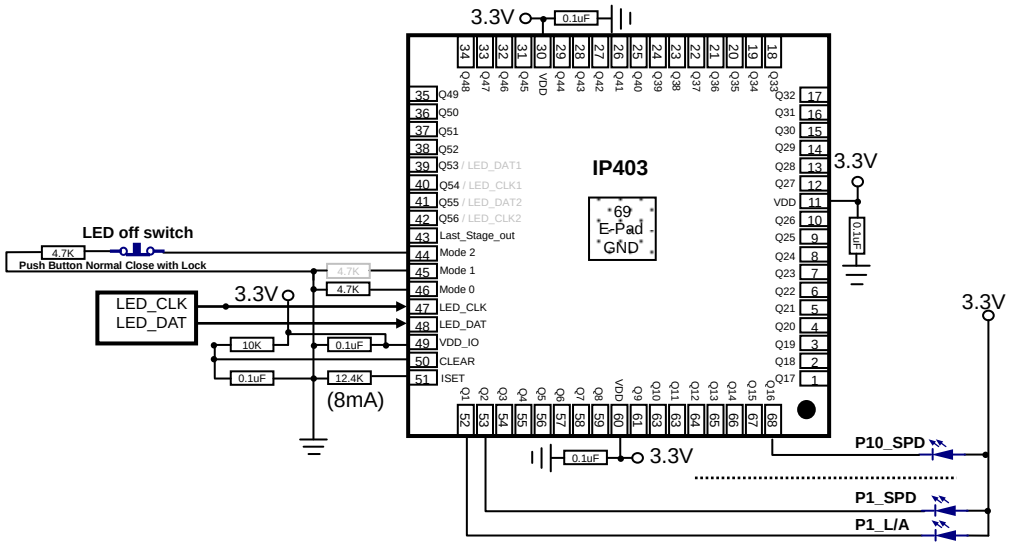



**Figure 12 2-bit stream mono-color LED application diagram**

## 5.2 VLAN

IP1810I supports port based VLAN, tag based VLAN and protocol based VLAN to implement flexible VLAN function, the detailed description is as follows.

### 5.2.1 Port Based VLAN

| Related registers | 0x30[7:0] to 3F[12][8], 0x60[7:0], 0x61[12][8], 0x68[7:0], 0x69[12][8] | PAGE 0x02 |
|---|---|---|

IP1810I provides port-based VLAN configurations. For each port, there are two registers to describe its port-based VLAN configuration. The VLAN group can be defined in page 0x02 registers 0x30[7:0] to 0x3F[12][8], 0x60[7:0], 0x61[12][8], 0x68[7:0] and 0x69[12][8].

Take the following example for the detail description. The data incoming from port 1 can be forwarded to the corresponding port defined in page 0x02 register 0x30 and 0x31. For example, if page 0x02 register 0x30 is written with 16'h000F, the packet from port 1 can be forwarded to port 2/3/4 only. Similarly, the forwarding rule for the data incoming from port 2 will refer to page 0x02 register 0x32 and 0x33. The forwarding rule for the data incoming from port 10 will refer to page 0x02 register 0x68 and 0x69.

### 5.2.2 Tag Based VLAN

| Related registers | 0x06[1], 0x09[7:0], 0x0A[12][8], 0x13 ot 0x1A, 0x2B, 0x2F, 0xD2 to 0xD8 | PAGE 0x02 |
|---|---|---|

IP1810I provides 4096 VLAN entries in VLAN table, the user can configure the VLAN table by programming page 0x02 register 0xD2 to 0xD8. If tag VLAN function is enabled and IP1810I receive a tagged packet, IP1810I will use the VID to lookup the VLAN entry in the VLAN table. If it is invalid, IP1810I drops the packet. If it is valid, IP1810I will forward the packet to destination port which belongs to the members of the VLAN, if this destination port is not a VLAN members, the packet will be dropped. Besides, If the source port is not a member of the VLAN, the packet can be either forwarded or dropped by programming page 0x02 register 0x09[7:0] and 0x0A[12][8].

When an un-tagged packet is received, IP1810I uses the PVID of the source port to look up the VLAN entry from VLAN table, the default PVID of the ports are defined in page 0x02 register 0x13 to 0x1A, 0x2B, 0x2F. IP1810I forwards the packet in the same way as mentioned above. For example, if port 1 receives a un-tag packet, IP1810I adopts the PVID of port 1 defined in bit[11:0] of page 0x02 register 0x13 to search the VLAN table, if the VLAN entry is valid, the packet will be forwarded to destination port which belongs to the members of the VLAN, the behavior of priority tag packet (VID equals to "000h") is the same as an un-tagged packet.

### 5.2.3 Protocol Based VLAN

| Related registers | 0x06[0], 0x6A to 0x71 | PAGE 0x02 |
|---|---|---|

IP1810I can configure a list of protocol types (Ether type, LLC,RFC1042) mapping to VLAN membership by programming page 0x02 register 0x06[0] and page 0x02 register 0x6A to 0x71. IP1810I provides 4 entries for the protocol based VLAN to use, each entry primarily consists of "type for protocol" and "protocol selection" settings.

IP1810I will check whether an incoming packet matches in “protocol selection” and “type for protocol” settings in corresponding register. For example, if the incoming packet format matches page 0x02 register 0x6A and 0x6B (entry 01), the VLAN membership refers to the VID of specific protocol type. (Note that the VID won’t be added to output port, IP1810I only use to the VID to look up the VLAN member for the packets forwarding.

When protocol selection is invalid, the default VLAN membership refers to PVID index.

**Table 18 PVID index search VLAN table**

| Protocol selection | Ingress packet | According to VID index | According to PVID index |
|---|---|---|---|
| **Invalid** | Ether type | --- | By PVID |
| | LLC | --- | By PVID |
| | RFC1042 | --- | By PVID |
| **Ether Type** | Ether type | By VID | --- |
| | LLC | --- | By PVID |
| | RFC1042 | --- | By PVID |
| **LLC** | Ether type | --- | By PVID |
| | LLC | By VID | --- |
| | RFC1042 | --- | By PVID |
| **RFC1042** | Ether type | --- | By PVID |
| | LLC | --- | By PVID |
| | RFC1042 | By VID | --- |

### 5.2.4 VLAN Function

| | | |
|---|---|---|
| **Related registers** | **0x06, 0x13 to 0x1A, 0x2B, 0x2F, 0xCC[7:0] to 0xCF[12][8], 0xD2 to 0xD8** | **PAGE 0x02** |
| | **0x07[7:0], 0x08[12][8]** | **PAGE 0x07** |
| | **0x03[0]** | **PAGE 0x0C** |

#### 5.2.4.1 Add/modify VLAN Tag

IP1810I provides two methods for adding VLAN tag. One is port-based tagging, and the other is VID-based tagging. For port-based tagging, if the corresponding bits of page 0x02 register 0xCC[7:0] and 0xCD[12][8] are set to “1” and page 0x02 register 0x06[1] is set to “1”, the packet will be added tag. For VID-based tagging, if the corresponding bits of “add tag field” of the VLAN entry are set to “1” and page 0x02 register 0x6[3] is set to “1” and page 0x02 register 0x06[1] is set to “1”, the packet will be added tag.

#### 5.2.4.2 Remove VLAN Tag

IP1810I also provides two methods for removing VLAN tag. One is port-based, and the other is VID-based. For port-based, if the corresponding bits of page 0x02 register 0XCE[7:0] and 0xCF[12][8] are set to “1” and page 0x02 register 0x06[1] is set to ”1”, the packet will be stripped tag. For VID-based, if the corresponding bits of “remove tag field” of the VLAN entry are set to “1” and page 0x02 register 0x6[3] is set to “1” and page 0x02 register 0x06[1] is set to “1” ,the packet will be stripped tag”.

Besides, if page 0x02 register 0x06[4] is set to “0” and page 0x0C register 0x03[0] is set to “1”, the packets that send to CPU Port will not add or remove tag. The operation is illustrated as follows. It is note that the VID-based tagging defined in the VLAN entry table is also used for 802.1Q tag-based VLAN.

**Table 19 Forward the Packet with VLAN Configuration Table**

| Frame type of the received packet | The operation on a switch with VLAN configuration | |
|---|---|---|
| | **Forward to a untagged field** | **Forward to a tagged field** |
| Untagged | Forward the packet without modification | 1. Insert a tag using the default VLAN tag value of the source or output port<br>2. Calculate new CRC<br>3. The default VLAN tag value will be added to output port by writing register 0xCC[7:0] and 0xCD[12][8]. |
| Priority-tagged (VLAN ID=0) | 1. Strip tag<br>2. Calculate new CRC | 1. Keep priority field.<br>2. Replace the tag with the default VLAN tag value of the source or output port<br>3. Calculate new CRC<br>4. The default VLAN tag value is defined in the registers bit[11:0] in the 0x13 to 0x1A, 0x2B and 0x2F. |
| VLAN-tagged | 1. Strip tag<br>2. Calculate new CRC | Forward the packet without modification |

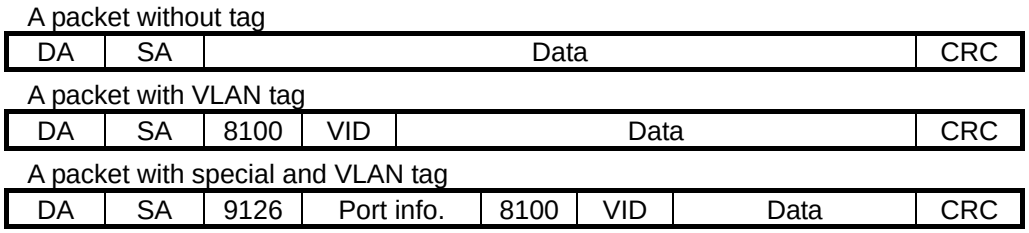



**Figure 13 Frame Format of Inserting VLAN or/and Special Tag**

### 5.2.4.3 VLAN up Link

| Related registers | 0xD0[7:0], 0xD1[12][8] | PAGE 0x02 |
|---|---|---|

When source port forwards packet to destination port in different VLAN group. For example, source port is in VLAN1, destination port is in VLAN2, and the packet will be sent to “uplink port” in VLAN1 if uplink port function is enabled, When source port forwards packet to destination port in different VLAN group, the packet will be dropped if uplink port function is disabled. Uplink port function is enabled by programming page 0x02 register 0xD1[13] to “1”, uplink port is configured by programming the corresponding bits of page 0x02 register 0xD0[7:0] and 0xD1[12][8].

### 5.2.4.4 Force the Incoming Packet Use PVID

| Related registers | 0x13[15] to 0x2A[15], 0x2B[15], 0x2F[15] | PAGE 0x02 |
|---|---|---|

IP1810I provides a kind of the mechanism for VLAN packets forwarding. If the bit[15] of page 0x02 register 0x13 to 0x1A, 0x2B and 0x2F is set to “1” for each port, IP1810I will use PVID for VLAN checking instead of VID for VLAN checking, that is, IP1810I will use PVID to lookup VLAN member for the packets forwarding. The function can be enabled individually for each port by programming each bit[15] form page 0x02 register 0x13 to 0x1A, 0x2B and 0x2F.

#### 5.2.4.5 VLAN Ingress Check

| Related registers | 0x09[7:0] to 0x0B[15:0], 0x0E[9:8][1:0] | PAGE 0x02 |
|---|---|---|

For packets forwarding, IP1810I provides two kinds of mechanism for the VLAN ingress check.

1. By port, IP1810I will check whether the source port of incoming packet belong to the same VLAN group by setting page 0x02 register 0x09[7:0] and 0x0A[12][8] to “1”. If the ingress port is not in VLAN membership, IP1810I will drop it.

2. By frame type, IP1810I will decide which type of the incoming packets should be forwarded by programming the corresponding bits of page 0x02 register 0x0B[15:0] and 0x0E[9:8][1:0], 2 bits for each port, which is represented in as follows,00: “all packet are forwarded”, 01: “only tag packet is forwarded”, 10: “priority and non-tag packet are forwarded”, 11: “priority and tag packet are forwarded”.

#### 5.2.4.6 VLAN Egress Rule

| Related registers | 0x0F[7:0], 0x10[15:13][12][8] | PAGE 0x02 |
|---|---|---|

For egress rule, IP1810I implements a set of VLAN egress rules, which defined port based, unicast, multicast and ARP. If an incoming packet matches specific rule. For example, if page 0x02 register10[15] is set to “1”, the ARP packet will use the default group setting for forwarding.

#### 5.2.4.7 VLAN Exclusive port

| Related registers | 0xCA[7:0], 0xCB[12][8] | PAGE 0x02 |
|---|---|---|

In order to provide flexible network management, IP1810I provides the VLAN exclusive port function which can block any packet for each exclusive port. However, exclusive port can communicate with non exclusive port. For example, if page 0x02 register 0xCA is set to 0x03, port 1 and port 2 can not receive packet for each other, but port 3 can receive packet from both port 1 and port 2.

#### 5.2.4.8 Inactive VID redirect

| Related registers | 0x6[15][14:10][1] | PAGE 0x02 |
|---|---|---|

The inactive VID means the specific VLAN field in the VLAN entry is set to 1’b0. If a packet refers to inactive VID, the packet will be dropped. However, IP1810I can also redirect the packet to specific port, broadcast or drop it. For example, if bit[15] and bit[14:10] of page 0x02 register 0x06 are separately set “1” and “00000”, IP1810I will force the packet to port 1.

## 5.3 Class Of Service (CoS)

### 5.3.1 Output Queue Schedule Mode with Priority

| Related registers | 0x01 to 0x23, 0x44, 0x45, 0x4C, 0x4D, 0x76 to 0x84 | PAGE 0x08 |
|---|---|---|
| | 0x01[5] | PAGE 0x00 |

IP1810I implements eight levels of priority queues (priority queue 0 to 7). The priority for each packet is based on the following schemes:

1. Port based
2. 802.1Q tag
3. IP TOS/DSCP
4. TCP Flag
5. TCP/UDP port number
6. Supreme priority (including ACL, IP address, source MAC address)

When CoS function is enabled, the following seven scheduling modes can be selectable. If CoS function is not enabled, the first-in/first-out (FIFO) forwarding method will be used to forward packets. The detailed configuration is shown in the following table. For the WRR scheduling mode, which is defined in page 0x08 register 0x01, the order of transmitting packet is sent out from high to low priority queue, Q7 is the highest queue, Q0 is the lowest queue.

Out queue schedule function also can be set for port group A and port group B. The default is group A. If bit[9] of page 0x08 register 0x76 is set to “1”, out queue schedule mode for group B also is enabled. In this condition, group A and group B can individually run their own out queue schedule mode simultaneously.

**Table 20 Output Queue Schedule Mode Description Table**

| Reg 0x01[2:0] | Schedule Mode | Function description |
|---|---|---|
| 00 | FIFO Schedule | All output packets are queued to queue 0, first comes first outs. |
| 01 | WRR/WFQ/BW/TWRR | Queue method is selected by setting page 0x8 register 0x01[6:5] |
| 02 | SPx1+WRR/WFQ/BW/TWRRx7 | Mixed mode, Q7 is SP , queue6~0 are WRR/WFQ/BW/TWRR. |
| 03 | SPx2+WRR/WFQ/BW/TWRRx6 | Mixed mode, Q7 and Q6 are SP , queue5~0 are WRR/WFQ/BW/TWRR. |
| 04 | SPx4+WRR/WFQ/BW/TWRRx4 | Mixed mode, Q7~Q4 are SP , queue3~0 are WRR/WFQ/BW/TWRR. |
| 05 | SPx8 | unless the high priority queue is empty, otherwise, low priority cannot transmit packets. |
| 06 | LLQx1+WFQ/BW/TWRRx7 | Mixed mode, Q7 is LLQ , queue6~0 are WFQ/BW/TWRR. |
| 07 | LLQx2+WFQ/BW/TWRRx6 | Mixed mode, Q7 and Q6 are LLQ , queue5~0 are WFQ/BW/TWRR. |

Note: FIFO(First In First Out), WRR(Weighted Round Robin), WFQ(Weighted Fair Queue), BW(Bandwidth), TWRR(Timing Weighted Round Robin), SP(Strict Priority), LLQ(Low Latency Queue).

IP1810I provides privilege priority to decide which priority of packet should be used. When these priorities are set to the packet of certain port, the order of the privilege priority is ACL > PTP > IGMP > IP address > MAC (LUT) > VID priority > TCP/UDP port number based priority > TCP flag > IP CoS based priority > VLAN tag based priority > port based priority.

IP1810I provides an option to suspend flow control function avoiding the extra delay for a high priority packet. A port's flow control function will be suspended for 1.5sec automatically when IP1810I receives a high priority packet. This function can be enabled by programming page 0x00 register 0x01[5] to "1".

Besides, IP1810I also supports SBM(static bandwidth management) and DBM(dynamic bandwidth management) mechanism to facilitate the use of out queue buffer, SBM and DBM is set for per port, means some port can be configured as either SBM or DBM.

### 5.3.2 Port Based Priority

| Related registers | 0x3A[14:0], 0x3B[8:0], 0x3E[14:12], 0x3F[11:9] | PAGE 0x00 |
|---|---|---|

The port-based priority only concerns the physical port location corresponding to switch. A packet received is regarded as the high priority packet from the high priority port. Each port of IP1810I can be configured as a high priority port individually by programming page 0x00 register 0x3A[14:0], 0x3B[8:0], 0x3E[14:12] and 0x3F[11:9]. For example, page 0x00 register 0x3A[2:0] is corresponding to port 1.

### 5.3.3 802.1Q VLAN Tag Based Priority

| Related registers | 0x40[7:0], 0x41[14][12][8] | PAGE 0x00 |
|---|---|---|

When the 802.1Q VLAN tag-based priority is enabled, the 802.1Q tag priority edition can be selected by programming page 0x00 register 0x41[14]. It will examine 3 bits of the priority field carried by a VLAN tag and map it to the corresponding priority. The priority mapping is shown in the following table. The 802.1Q tag-based priority function for each port can be enabled individually by programming page 0x00 register 0x40[7:0] and 0x41[12][8].

| Priority field | Queue0 | Queue1 | Queue2 | Queue3 | Queue4 | Queue5 | Queue6 | Queue7 |
|---|---|---|---|---|---|---|---|---|
| First Ver. | 2 | 0 | 1 | 3 | 4 | 5 | 6 | 7 |
| Second Ver. | 1 | 0 | 2 | 3 | 4 | 5 | 6 | 7 |

### 5.3.4 IP ToS/ DSCP CoS Based Priority

| Related registers | 0x42[7:0], 0x43[12][8] to 0x49 | PAGE 0x00 |
|---|---|---|

IP1810I provides the IP layer CoS function by recognizing the priority octet and mapping it to the corresponding priority. For an IPv4 packet, it is embedded in the ToS (Type of Service) Octet. For an IPv6 data packet, the Traffic Class Octet is used to differentiate the Class of Service. When this function is enabled, IP1810I will automatically recognize the IP version and capture either the TOS field (IPv4) or the Differentiated Services field (IPv6).

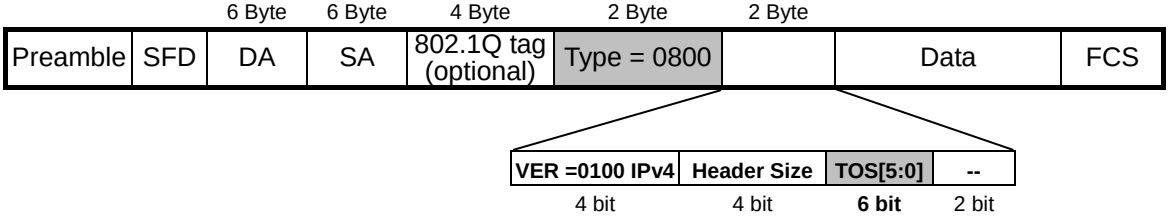



**Figure 14 Frame Format of IPv4 TOS**

The eight bits were allocated to a TOS field in the IP header. They can be divided into 5 sub fields:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Precedence | | | D | T | R | Unused | |

0=Most Significant bit : 7 = Least Significant bit.

The value of the 3 precedence bits are from 0 to 7 and are used to indicate the importance of a datagram (default is 000, higher is better).

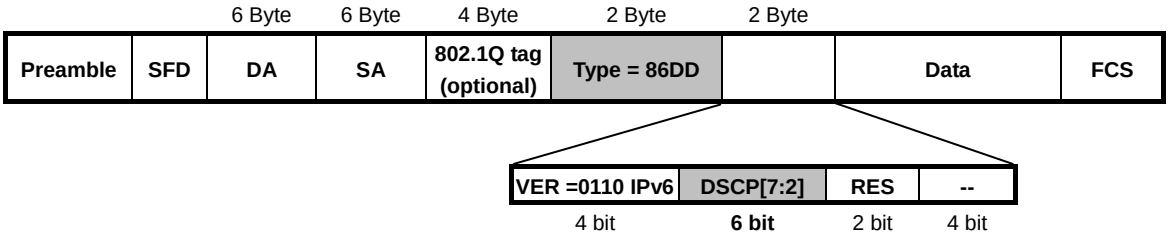



**Figure 15 Frame Format of IPv6 DSCP**

The IP TOS/DS priority function can be enabled individually for each port by programming page 0x00 register 0x42[7:0] and 0x43[12][8], and user can define IP TOS/DS priority setting for seven kinds of code point. If IP1810I receives a packet with IP TOS/DS code point, which is not defined in corresponding register, IP1810I provides two kinds of methods to handle undefined code point for the packet. One is the zero priority method, the packet will be treated as a lowest priority packet by programming page 0x00 register 0x45[9] to “0”. The other is the Tag/Port based method, and the packet will be assigned a priority according to its priority setting on tag/port by programming page 0x00 register 0x45[9] to “1”.

### 5.3.5 TCP/UDP Port Number Based Priority

| Related registers | 0x0A to 0x1C[12][8] | PAGE 0x00 |
|---|---|---|

When the TCP/UDP port number function is enabled, IP1810I will examine the TCP/UDP destination port number of the packet to decide its priority.

The user can define the priority for TCP/UDP packets by programming the corresponding bits of page 0x00 register 0x12 to 0x16. For example, that a packet with TCP/UDP port number equal to 20 or 21 (FTP) will be handled as a highest priority packet if page 0x00 register 0x12[3:0] is set as 4b’0111(Q7). That a packet with TCP/UDP port number equal to 20 or 21 (FTP) will be handled as a lowest priority packet if page 0x00 register 0x12[3:0] is set as 4b’0000.(Q0)

A packet with TCP/UDP port number matches the user-defined port number, which is defined in page 0x00 register 0x0A to 0x11, the packet will be treated as a priority packet depending on the setting in page 0x00 register 0x15[15:12] and 0x16[15:0]. IP1810I provides three groups for user-define port number range. For example, if port number group C is set to “highest priority” by programming page 0x00 register 0x16[7:4], and the range of port number is set from 100 to 1 by programming page 0x00 register 0x0C and 0x0D, the incoming packet(port number=10) will be forwarded to highest priority queue. The TCP/UDP port number based priority function of each port can be enabled individually by programming page 0x00 register 0x1B[7:0] and 0x1C[12][8].

For the setting of TCP/UDP port number, user-defined range setting will override the setting of page 0x00 register 0x12 to 0x15. For example, if the priority of FTP (20, 21) is set to low and the priority of the port number (1~21) is set to high for user-defined range setting, the priority of port number (20, 21) will be treated as high priority rather than low priority.

**Table 21 TCP/UDP Port Number priority Selection Table**

| Register (PAGE 0) | | Port number | Definition |
|---|---|---|---|
| FTP | 0x12[3:0] | 20,21 | 0000: Queue 0<br>0001: Queue 1<br>0010: Queue 2<br>0011: Queue 3<br>0100: Queue 4<br>0101: Queue 5<br>0110: Queue 6<br>0111: Queue 7 |
| SSH | 0x12[7:4] | 22 | |
| TELNET | 0x12[11:8] | 23 | |
| SMTP | 0x12[15:12] | 25 | |
| DNS | 0x13[3:0] | 53 | |
| BOOTP/DHCP | 0x13[7:4] | 67,68 | |
| TFTP | 0x13[11:8] | 69 | |
| HTTP_0,1 | 0x13[15:12] | 80 | |
| POP3 | 0x14[3:0] | 110 | |
| NEWS | 0x14[7:4] | 119 | |
| SNTP | 0x14[11:8] | 123 | |
| NETBIOS0,1,2 | 0x14[15:12] | 137~139 | |
| IMAP _0,1 | 0x15[3:0] | 143,220 | |
| SNMP_0,1 | 0x15[7:4] | 161,162 | |
| HTTPS | 0x15[11:8] | 443 | |
| USR A | 0x15[15:12] | User define A | |
| USR B | 0x16[3:0] | User define B | |
| USR C | 0x16[7:4] | User define C | |
| USR D | 0x16[11:8] | User define D | |
| USR E | 0x16[15:12] | User define E | |

### 5.3.6 Source MAC address Priority

| Related registers | 0x02[3], 0xE0 to 0xE5 | PAGE 0x01 |
|---|---|---|
| | 0x4A[7:0], 0x4B[12][8] | PAGE 0x00 |

IP1810I has an option to support the Source MAC Address based Priority. IP1810I will check the corresponding bits of the LUT entry to decide the priority of the packet if source MAC address priority function is enabled, the LUT table can be configured by programming page 0x01 register 0xE0 to 0xE5. The MAC address priority function can be enabled individually for each port by programming the corresponding bits of page 0x00 register 0x4A[7:0] and 0x4B[12][8].

### 5.3.7 VID Based VLAN priority

| Related registers | 0x4C[7:0], 0x4D[12][8] | PAGE 0x00 |
|---|---|---|
| | 0x06[1], 0xD2 to 0xD8 | PAGE 0x02 |

IP1810I has an option to support VID based priority. This function can be enabled individually for each port by programming the corresponding bits of page 0x00 register 0x4C[7:0] and 0x4D[12][8]. IP1810I will recognize the VID of incoming packets, and then check VLAN Table to find out corresponding priority setting. Each VID priority field of VLAN Table can be configured individually by programming page 0x02 register 0xD2 to 0xD8. If priority Field in VLAN entry is set to “111”, means highest queue, If priority Field in VLAN entry is set to “000”, means lowest queue.

### 5.3.8 IP address Priority

| Related registers | 0x4E[7:0], 0x4F[12][8] | PAGE 0x00 |
|---|---|---|
| | 0x2F to 0x3A | PAGE 0x01 |
| | 0x02[15] | PAGE 0x02 |

IP1810I has an option to support the source IP address priority. If the function is enabled, IP1810I will look up the IP entry from IP table to decide its priority. IP table can be configured by programming page 0x01 register 0x2F to 0x3A. For example, if the PRI field of IP entry is set to “1111”, the IP packet will be forward to highest priority queue. If the PRI field of IP entry is set to “1000”, the IP packet will be forward to lowest priority queue. IP address priority function is enabled individually for each port by programming the corresponding bits of page 0x00 register 0x4E[7:0] and 0x4F[12][8] to “1”. The Hashing algorithm for IP address can be set either CRC or direct method by programming page 0x02 register 0x02[15].

### 5.3.9 IP multicast Priority

| Related registers | 0x2F to 0x39 | PAGE 0x01 |
|---|---|---|
| | 0x50[7:0], 0x51[12][8] | PAGE 0x00 |

IP1810I has an option to support IP multicast priority. IP multicast priority function can be enable individually for each port by programming the corresponding bits of page 0x00 register 0x50[7:0] and 0x51[12][8]. The user can set the priority for the IP multicast entry by programming page 0x01 register 0x2F to 0x39, the packet will be forward to out queue by programming PRI field of IP multicast table.

### 5.3.10 ACL Priority

| Related registers | 0x52[7:0], 0x53[12][8] | PAGE 0x00 |
|---|---|---|

The ACL function also has an action for priority assignment. User can set the priority for the ACL rule. The ACL priority function can be enable individually for each port by programming the corresponding bits of page 0x00 register 0x52[7:0] and 0x53[12][8]. The packet will be forward to out queue by ACL action if it matches ACL rule.

## 5.4 Capture Ethernet protocol frame & IP packet to CPU port

### 5.4.1 In Band Management Frame

| | | |
|---|---|---|
| **Related registers** | **0x01[6], 0x5C to 0x5E, 0x21, 0x22** | **PAGE 0x00** |
| | **0x03[0]** | **PAGE 0x0C** |

IP1810I's default MAC address is 00-90-c3-00-00-03. If the DA of an incoming packet matches IP1810I's MAC address, it will be sent to the CPU port if page 0x00 register 0x01[6] is set to "1" and page 0x0C register 0x03[0] is set to "1". IP1810I's MAC address can be changed by programming page 0x00 register 0x5C to 0x5E.

IP1810I also provides in-band management restriction function. By checking the content of the related packet to decide the packet should be forwarded or dropped. For example, If page 0x00 register 0x01[6] is set to "1" and page 0x00 register 0x22[15] is set to "1", and the DA of the packet meets to the switch MAC address, IP1810I will forward or drop the packet, according to the corresponding bits of page 0x00 register 0x21[15:0] and page 0x00 register 0x22[14:0] , if the corresponding bit is set to "1" ,the packet will be forwarded, otherwise, the packet will be dropped.

### 5.4.2 Block Broadcast Frames to CPU Port

| | | |
|---|---|---|
| **Related registers** | **0x0D[11:10]** | **PAGE 0x01** |
| | **0x03[0]** | **PAGE 0x0C** |

In order to prevent broadcast packets to impact the performance of CPU port, IP1810I can drop broadcast and multicast frames to CPU port if page 0x01 register 0x0D[10] is set to "1" and page 0x0C 0x03[0] is set to "1", IP1810I also can only forward IPv4 /IPv6 broadcast and multicast frames to CPU port if page 0x01 register 0x0D[11] is set to "1" for specific application for CPU.

### 5.4.3 ARP and ICMP Storm Control

| | | |
|---|---|---|
| **Related registers** | **0x14 to 0x19[12][8]** | **PAGE 0x01** |

In order to prevent the ARP and ICMP storm to affect the performance of IP1810I, IP1810I implements ARP and ICMP storm control mechanism to drop excessive ARP/ICMP packets in specific period, ARP storm is enabled for each port by programming the corresponding bits of page 0x01 register 0x15[7:0] and 0x16[12][8] to "1", ICMP storm is enabled for each port by programming the corresponding bits of page 0x01 register 0x18[7:0] and 0x19[12][8] to "1".

A port begins to drop incoming ARP packets if the received ARP packet reaches the threshold defined in page 0x01 register 0x14[7:0], a port begins to drop the incoming ICMP packets if the received ICMP packet reaches the threshold in page 0x01 register 0x17[7:0]. Besides, IP1810I provides four kinds of periods to be selected by programming page 0x01 register 0x14[9:8] for ARP storm control and 0x17[9:8] for ICMP storm control to "00~11".

**Table 22 ARP and ICMP Storm Counter Clear Period Selection Table**

| 0x14[9:8] & 0x17[9:8] | 1000Mbps | 100Mbps | 10Mbps |
|---|---|---|---|
| 00 | 200us | 2ms | 20ms |
| 01 | 1ms | 10ms | 100ms |
| 10 | 10ms | 10ms | 10ms |
| 11 | 100ms | 100ms | 100ms |

### 5.4.4 Block ARP to CPU Port

| Related registers | 0x14[10] | PAGE 0x01 |
|---|---|---|
| | 0x03[0] | PAGE 0x0C |

In order to prevent ARP storm to reduce the performance of CPU port, IP1810I can drop ARP frames to CPU port If page 0x01 register 0x14[10] is set to “1” and page 0x0C register 0x03[0] is set to “1”. The default setting is ARP packets to be forwarded to CPU port.

### 5.4.5 Ethernet L2 protocol packet capture

| Related registers | 0x06 to 0x07 | PAGE 0x00 |
|---|---|---|

IP1810I recognizes layers 2 protocol frames (BPDU, 802.1x, LLDP, Slow Protocol, GxRP, ARP, and PPPoE) to decide packet should be forwarded, dropped or “To CPU” by programming page 0x00 register 0x06 and 0x07 and page 0x0C register 0x03[0], the detailed description is shown in the following table ,IP1810I also provides L2 protocols for user-defined by programming page 0x00 register 0x07[14] and page 0x00 register 0x23.

**Table 23 Layers Two Protocol Frames Selection Table**

| L2 Protocol | DA | Register (PAGE 0) | Definition |
|---|---|---|---|
| BPDU | (01-80-C2-00-00-00) | 0x06[1:0] | 00: forward<br>01: reserved<br>10: To CPU port only.<br>11: drop |
| Slow Protocol | (01-80-C2-00-00-02) | 0x06[3:2] | |
| 802.1x | (01-80-C2-00-00-03) | 0x06[5:4] | |
| LLDP | (01-80-C2-00-00-0E) | 0x06[7:6] | |
| Group 0 | (01-80-C2-00-00-05, 06, 09~0C, 0F) | 0x06[9:8] | |
| | 01-80-C2-00-00-04, 07, 08, 0D | 0xB5[7:0] | |
| All Bridge address | (01-80-C2-00-00-10) | 0x06[11:10] | |
| Group 1 | (01-80-C2-00-00-11~1F) | 0x06[13:12] | |
| GARP | (01-80-C2-00-00-20, 21) | 0x06[15:14] | |
| Group 2 | (01-80-C2-00-00-22~2F) | 0x07[1:0] | |
| Group 3 | (01-80-C2-00-00-30~3F) | 0x07[3:2] | |
| ARP ether type = 0806 | | 0x07[7] | 0 : Forward depend on DA<br>1 : To DA and CPU port |
| IPv6 ether type = 86DD | | 0x07[10] | 0 : Forward depend on DA<br>1 : To CPU port |

### 5.4.6 Ethernet L3 protocol packet capture

| Related registers | 0x08 to 0x09 | PAGE 0x00 |
|---|---|---|

IP1810I recognizes layer 3 protocol frame (ICMP, TCP, UDP, OSPF ,and ICMP) to decide the packet should be forwarded, dropped or “To CPU” by programming page 0x00 register 0x08 and page 0x0C register 0x03[0]. The detailed configuration is shown in the following table.IP1810I also provides two L3 protocols for user-defined through page 0x00 register 0x08[11:8] and 0x09.

**Table 24 Level Three Protocol Frames Selection Table**

| L3 Protocol | Register (PAGE 0) | Definition |
|---|---|---|
| ICMP | 0x08 [1:0] | 00: forward<br>01: reserved<br>10: To CPU port only.<br>11: drop |
| TCP | 0x08 [3:2] | |
| UDP | 0x08 [5:4] | |
| OSPF | 0x08 [7:6] | |
| User define 1 | 0x08 [9:8] | |
| User define 2 | 0x08 [11:10] | |
| Ipv4 Other protocol | 0x08 [13:12] | |

### 5.4.7 PPPoE Protocol Check

| Related registers | 0x07[9:8], 0x56 to 0x59 | PAGE 0x00 |
|---|---|---|

IP1810I will check session ID of the PPPoE packets if page 0x00 register 0x07[9:8] are set to “11”, if it matches, the header of the PPPoE packet will be removed ,if it doesn’t match, the PPPoE header won’t be removed. The PPPoE session ID can be configure by programming the corresponding bits of page 0x00 register 0x56 to 0x59.

## 5.5 Security

### 5.5.1 MAC Address Based Security

| Related registers | 0x02[6:4], 0x03[7:0], 0x04[12][8] | PAGE 0x01 |
|---|---|---|

In order to drop illegal MAC packets, IP1810I supports the MAC address based security function. When this function is enabled, IP1810I has an option to drop the packet with the SA that does not find in the MAC table. This function is valid only if the MAC learning process is disabled by programming page 0x01 register 0x03[7:0] and 0x04[12][8].

For an unknown SA packet, If page 0x01 register 0x02[5] is set to “1”, the unknown SA packet will be forwarded. If page 0x01 register 0x02[5:4] is set to “01”, the unknown SA packet will be dropped. If page 0x01 register 0x02[5:4] is set to “00”, the unknown SA packet will be forwarded to CPU port.

IP1810I also supports SA associated with source port if page 0x01 register 0x02[6] is set to “1”. For example, if a packet with specific SA is learnt by port 1 in LUT, when the same SA of the packet on port 2 want to be sent to other port, the packets will not be forwarded. The packets with specific SA can send from any port if page 0x01 register 0x02[6] is set to “0”.

### 5.5.2 802.1x Port Based Security

| Related registers | 0x01 [6], 0x06[5:4], 0x54[7:0], 0x55[12][8] | PAGE 0x00 |
|---|---|---|

IP1810I supports 802.1x port based security function. If this function is enabled, IP1810I only forwards 802.1x EAPOL packets (DA=01 80 C2 00 00 03 and packet type=88 8E or DA= switch’s MAC address, packet type=88 8E and 0x01[6]=1) to CPU port and drops other types of packets. However, the ARP packets are not affected by this function and are forwarded according to their destination address.

The User can enable the 802.1x security function through programming page 0x00 register 0x06[5:4] to “10”. If the page 0x00 register 0x06[5:4] is set to “11”, all the EAPOL packets will be dropped. This function can be enabled individually for each port by programming page 0x00 register 0x54[7:0] and 0x55[12][8].

### 5.5.3 IP Address Based Security

| Related registers | 0x01[7:0], 0x02[15:13][12][8] | PAGE 0x02 |
|---|---|---|

IP1810I supports IPv4/IPv6 address based security function by examining the SIP address of an incoming packet. User can set each port of IP1810I to check source IP address by programming page 0x02 register 0x01[7:0] and 0x02[12][8], and select one of the 2 filter modes by programming page 0x02 register 0x02[13] as shown in the following table.

To perform IP security, user has to fulfill the IP entries, if bit[183:181] in IP table are set to “111”, IP1810I will check IP address, MAC address, and source port of incoming packet whether matches the setting of IP table or not, if it matches, the packet will based on filter rule by programming page 0x02 register 0x2[13] to decide packet should be forwarded or dropped, There are two hashing methods to decide the location of an IP entry, hashing methods can support direct and CRC by programming page 0x02 register 0x02[15].

### 5.5.4 TCP/UDP Port Number Based Security

| Related registers | 0x12 to 0x20 | PAGE 0x00 |
|---|---|---|

IP1810I supports TCP/UDP port number based security function by checking the port number in an incoming packet. User can enable the function by programming register 0x1B[7:0] and 0x1C[12][8]. For instance, port 1 is corresponding to bit 0 of page 0x00 register 0x1B and port 9 is corresponding to bit 8 of page 0x00 register 0x1C.

When this function is enabled, a packet will have one of three behaviors : 1. to CPU 2. to CPU and ports 3. Drop it, if the TCP/UDP port number matches the conditions, which defined in page 0x00 register 0x12 to 0x16. For example, a packet with TCP/UDP port number (FTP: 20 or 21) will be dropped if page 0x00 register 0x12[3:0] is set to “1010”. A packet with TCP/UDP port number (SMTP: 25) will be forwarded to CPU if page 0x00 register 0x12[15:12] is set to “1001”.

IP1810I provides two user-define port number setting and three user-define port number range setting to let user define port number he wants.(reg. 0x0A to 0x11).

The well known TCP/UDP port is shown in following Table (defined in page 0x00 register. 0x12 to 0x15[11:0]), user-defined port number range setting of the TCP/UDP will override the well known TCP/UDP setting if the TCP/UDP port number is the same. For example, if FTP (20, 21) is set to drop and the port number (1~21) is set to forward to CPU port for user-defined range setting, a packet with port number (20, 21) will be forwarded to CPU port rather than dropped. There is a privilege for user-define range port number.

**Table 25 TCP/UDP Port Number Based Security Selection Table**

| Register (PAGE 0) | | Port number | Definition |
|---|---|---|---|
| FTP | 0x12 [3:0] | 20, 21 | 1001: To CPU<br>1010: drop<br>1011: To CPU and ports |
| SSH | 0x12 [7:4] | 22 | |
| TELNET | 0x12 [11:8] | 23 | |
| SMTP | 0x12 [15:12] | 25 | |
| DNS | 0x13 [3:0] | 53 | |
| BOOTP/DHCP | 0x13 [7:4] | 67, 68 | |
| TFTP | 0x13 [11:8] | 69 | |
| HTTP_0,1 | 0x13 [15:12] | 80 | |
| POP3 | 0x14 [3:0] | 110 | |
| NEWS | 0x14 [7:4] | 119 | |
| SNTP | 0x14 [11:8] | 123 | |
| NETBIOS0,1,2 | 0x14 [15:12] | 137~139 | |
| IMAP _0,1 | 0x15 [3:0] | 143, 220 | |
| SNMP_0,1 | 0x15 [7:4] | 161, 162 | |
| HTTPS | 0x15 [11:8] | 443 | |
| USR A | 0x15 [15:12] | User define A | |
| USR B | 0x16 [3:0] | User define B | |
| USR C | 0x16 [7:4] | User define C | |
| USR D | 0x16 [11:8] | User define D | |
| USR E | 0x16 [15:12] | User define E | |

### 5.5.5 Port Mirroring Security (Sniffer)

| Related registers | 0x1E[7:0] to 0x23[12][8], 0x24[7:5] | PAGE 0x01 |
|---|---|---|

In some circumstances, the network administrator requires to monitor the network traffic. The port mirroring function will help the network administrator to diagnose the content of network traffic.

IP1810I provides sniffer function to meet different actual requirement. It provide two destination port group configuration and several trigger criteria. These Sniffer function are illustrated as follows.

**First Sniffer destination port group configuration function:**

A port mirroring function is accomplished by assigning monitored ports (source ports), snooping ports (destination ports) and snooping method. IP1810I will copy the monitored packets to all snooping ports. The IP1810I supports three kinds of snooping methods: the ingress, egress and ingress + egress, which is defined in page 0x01 register 0x1F[14:13], the ports to be monitored (mirroring source),which is defined in register 0x22[7:0] and 0x23[12][8], the ports to snoop (mirroring destination), which is defined in page 0x01 register 0x1E[7:0] and 0x1F[12][8].

For example, if user wants to monitor the output traffic of port 8 and port 7 from port 1 as shown in the following figure. He has to write “01” to page 0x01 register 0x1F[14:13] to choose monitor method to be egress traffic, write 0x0080 and 0x0100 to page 0x01 register 0x22 and page 0x01 register 0x23 to select port 8 and port 9 to be monitored ports, write 0x0001 and 0x0000 to page 0x01 register 0x1E and page 0x01 register 0x1F to select port 1 as a monitoring port. IP1810I will copy the egress traffic port 8 and port 9 to port 1.

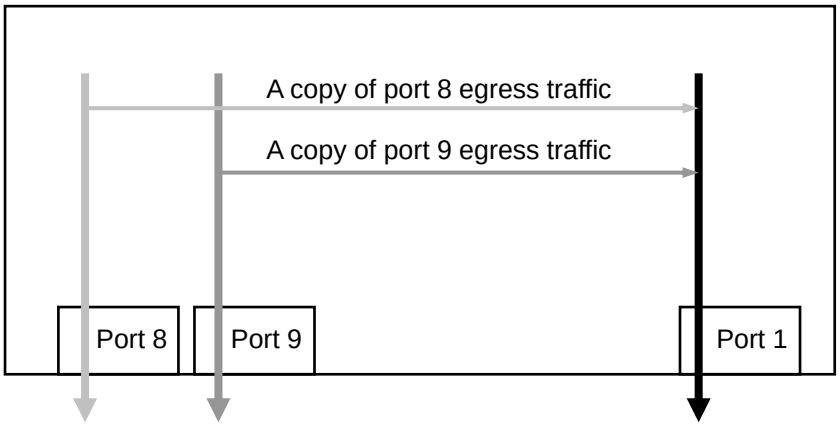



**Figure 16 Port Mirroring Security Block Diagram**

IP1810I also has options to keep the mirrored packet with the original tagging. The source of the mirrored packets can be separated into three groups. Their tagging status can be controlled through page 0x01 register 0x1F[15] and 0x24[7:5].

The above description is summarized as follows.

Sniffer Add/Rem Tag option for ACL routing

| Original packet | Action(add/remove tag) by register 0xCC~0xCF(page 0x02) | behavior | |
|---|---|---|---|
| | | Reg1f[15]=1, Reg24[6]=0(page 0x01) | Reg1f[15]=1, Reg24[6]=1(page 0x01) |
| | | Destination port | Destination port |
| Untag | Add tag | Untag | Tag |
| Tag | Remove tag | Tag | Untag |

Destination port must be assigned in the field of mode entry of ACL

Sniffer Add/Rem Tag option for CPU special tag routing

| Original packet | Action(add/remove tag) by register 0xCC~0xCF(page 0x02) | behavior | |
|---|---|---|---|
| | | Reg1f[15]=1, Reg24[5]=0(page 0x01) | Reg1f[15]=1, Reg24[5]=1(page 0x01) |
| | | Destination port (only for CPU port) | Destination port (only for CPU port) |
| Untag | Add tag | Untag | Tag |
| Tag | Remove tag | Tag | Untag |

Sniffer Add/Rem Tag option for other packet

| Original packet (monitored port) | Action (add/remove tag) by register 0xCC~0xCF (page 0x02) | behavior | |
|---|---|---|---|
| | | Reg1f[15]=1, Reg24[7]=0(page 0x01) | Reg1f[15]=1, Reg24[7]=1(page 0x01) |
| | | Destination port | Destination port |
| Untag | Add tag | Untag | Tag |
| Tag | Remove tag | Tag | Untag |

Destination port must be assigned in the field of special tag.

**Second Sniffer destination port group configuration function:**

IP1810I also provides second sniffer destination port group configuration function for MAC/IGMP/IP/ACL sniffering. IP1810I will forward packets to destination port for either first sniffer group or second sniffer group, according to MAC/IGMP/IP/ACL sniffering, Besides, DA or SA trigger can be selected for MAC address table by programming page 0x01 register 0x21[14:13], which can individually support first sniffer destination group configuration and second sniffer destination group configuration.

## 5.6 WAN/LAN Port Filtering

| | | |
|---|---|---|
| **Related registers** | **0x0A to 0x11, 0x17 to 0x1A[12][8]** | **PAGE 0x00** |
| | **0x0D, 0x10[9:8][1:0], 0x11[7:0] to 0x13, 0x16[9:8][1:0]** | **PAGE 0x06** |

IP1810I supports TCP/UDP port number based filter function by checking both the port number in an incoming packet and an outgoing packet. User can enable the function by programming page 0x00 register 0x17 to 0x1A[12][8]. IP1810I also supports positive and negative filter. User can select either positive or negative filter to use, in positive filter, if a packet with TCP/UDP port number doesn't match the corresponding bit of TCP/UDP port number setting, the packet should be dropped according to WAN/LAN port filtering by programming page 0x06 register 0x0D, 0x10[9:8][1:0], 0x11[7:0], 0x12[12][8], 0x13 and 0x16[9:8][1:0]. In negative mode, If a packet with TCP/UDP port number matches the corresponding bits of TCP/UDP port number setting, the packet should be dropped according to WAN/LAN port filtering by programming page 0x06 register 0x0D, 0x10[9:8][1:0], 0x11[7:0], 0x12[12][8], 0x13 and 0x16[9:8][1:0]. WAN/LAN port filtering can be configured as "disable filter", RX drop, TX drop, or TX/RX drop. For example. If "disable filter" is set, means that packet will be forwarded, if "RX drop" is set, means that packet will be dropped at host port.

IP1810I can choose either WAN port or LAN port by programming page 0x06 register 0x11[7:0] and 12[12][8]. For example, if the host port(port 1) is set to "forward to LAN port only" by programming page 0x06 register 0x0D[1:0] to "10" and client port(port 2) is set to LAN port by programming page 0x06 register 0x11[1] to "0" , the drop on ACT filtering only work at client port, and then the packet will be forwarded or dropped by programming page 0x06 register 0x0D[3:2] to "01" and programming page 0x06 register 0x13[3:2] to "00~1X", if client port is WAN port, the packet is always dropped. If the host port is set to "forward to WAN port only" and client port is set to WAN port, the filtering only work at client port, if client port is LAN port, the packet is always dropped.

If host port(port 1) is set to "drop on ACT filtering" by programming page 0x06 register 0x0D[1:0] to "01" and programming page 0x06 register 0x13[1:0] to "00"(RX drop) , the packet will be dropped at host port(port 1), If client port(port 2) is set to "drop on ACT filtering" by programming "page 0x06 register 0x0D[3:2] to "01" and programming page 0x06 register 0x13[3:2] to "01"(TX drop) the packet will be dropped at client port(port 2). If host port and client port are set to "Disable Forwarding WAN/LAN& Packet drop", the packet is always forwarded.

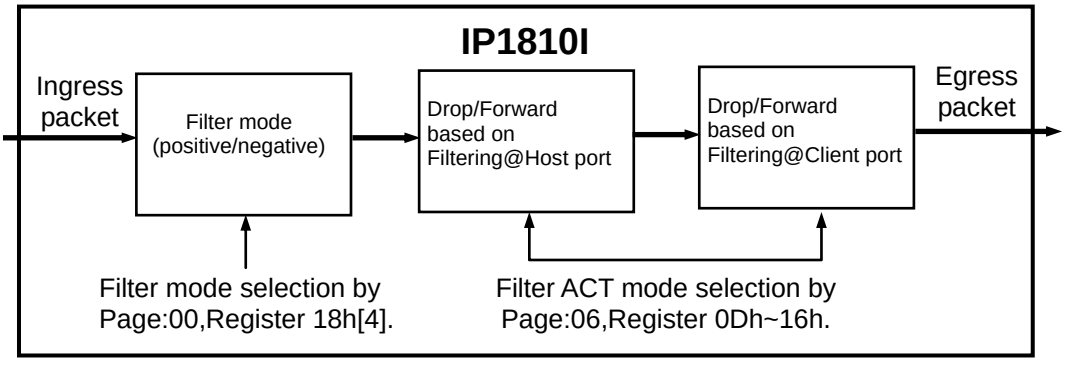



**Figure 17 WAN Port Filtering Block Diagram**

IP1810I's TCP/UDP port number filter mode (positive or negative) and the filter method conditions (disable, drop on act, Forwarding to WAN port only ,forward to LAN port only) is summarized as follows, please refer to the following table:

**Table 26 Host / Client Forwarding WAN / LAN and Packet drop setting Condition Table**

| Host / Client Forwarding WAN / LAN and Packet drop setting Condition | | | | | | | | | | | | | | (1) **V : PASS**<br>(2) **X : Drop**<br>(3) Host port Switch Port to receive packet<br>(4) Client port Switch Port to transmit packet |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Host / Client | | | Client Port setting | | | | | | | | | | | |
| | | | Client is LAN Port | | | | | | Client is WAN Port | | | | | |
| | | | 00 Disable | 01 Drop | | | 10 LAN | 11 WAN | 00 Disable | 01 Drop | | | 10 LAN | 11 WAN |
| | | | | RX | RX+TX | TX | | | | RX | RX+TX | TX | | |
| Host port setting | 00 Disable | | V | V | X | X | V | V | V | V | X | X | V | V |
| | 01 Drop | Act on RX | X | | | | | | X | | | | | |
| | | Act on TX+RX | | | | | | | | | | | | |
| | | Act on TX | V | V | X | X | V | V | V | V | X | X | V | V |
| | 10 FWD LAN | | V | V | X | X | V | V | X | X | X | X | X | X |
| | 11 FWD WAN | | X | X | X | X | X | X | V | V | X | X | V | V |

## 5.7 IEEE 1588 Precision Timing Protocol (PTP)

| Related registers | 0x00 to 0x36 | PAGE 0x09 |
|---|---|---|

IP1810I implements the precision timing protocol (including IEEE1588/IEEE802.1as) function. For the PTP application, IP1810I uses the PTP dedicated hardware to capture the timestamp of PTP frame and store them into internal buffer; the software processes the various protocol message and co-work with PTP dedicated hardware.

To synchronous the master clock, IP1810I offers the adjustable real time clock (RTC), which could increases, decreases the clock frequency and set the specific time-value into RTC. If the PTP packet is detected, the PTP dedicated hardware will capture the time-value of SFD position for PTP or event signal. After the PTP synchronization process, the CPU uses the ingress/egress time-stamp to obtain the compensation parameter. Then, the CPU will pass the relating compensation value into PTP dedicated hardware. After the long duration, the PTP dedicated RTC is synchronous with the remote master clock.

For each port, IP1810I supports 32 depth timestamp FIFO(16 for ingress, 16 for egress) to record PTP time-stamp. When all time stamps are occupied, user also set overwrite bit enable to over write time stamps cyclically. Another, IP1810I also supports 8 time stamps for incoming event trigger signal.

For the monitor system application, IP1810I support three different output synchronous signals. These output signals are pulse-per second, the programming clock duration signal and trigger-out signal. Further, please refer to PTP application note in detail.

## 5.8 Hardware Loop detection

| | | |
|---|---|---|
| **Related registers** | **0x6B[7:0] to 0x73, 0xA5[7:0], 0xA6[12][8]** | **PAGE 0x00** |
| | **0x24[3]** | **PAGE 0x01** |

IP1810I support hardware loop detection ,Each port can configure loop detection by programming page 0x00 register 0x6B[7:0] and 0x6C[12][8]. IP1810I will send loop detect packet to detect whether or not loop happen, IP1810I can check loop status of each port by reading the corresponding bits of page 0x00 register 0xA5[7:0] and 0xA6[12][8], if loop occur, IP1810I will block redundant until loop release. For blocking redundant path, page 0x01 register 0x24[3] must be set to “1”.

Hardware loop detection and EEE function can not be enabled at the same time. Because loop detect packet can not be sent at EEE mode.

## 5.9 Trunk Channel

| | | |
|---|---|---|
| **Related registers** | **0x1C** | **PAGE 0x01** |

### 5.9.1 Trunk Channel Behavior

IP1810I provides two kinds of trunk methods to configure trunk group by programming page 0x01 register 0x1C. one is the single trunk group and the other is the double trunk group, IP1810I supports 2 single trunk groups and 1 double trunk groups by programming page 0x01 register 0x1C[5]. The single trunk group A and B consist of 2 to 4 ports. The two single trunk groups can merge into one trunk group, for example, trunk A and B can be combined as a large trunk group, the members of double trunk group can be set as port 1~8(at least 2 ports), and so on. User can configure either the members of single trunk group or double trunk group individually by writing “1” to the corresponding bits of the port for each trunk in page 0x01 register 0x1C[15:8]. A trunk channel works as if a “big” port with multiple times of bandwidth. If the destination port of a packet is un-link, IP1810I forwards the packet to the other port of the trunk (auto recovery).

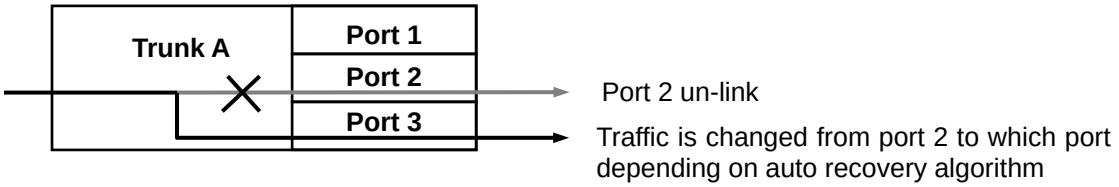



**Figure 18 Trunk Channel Behavior Block Diagram**

### 5.9.2 Load Balance

To fully utilize the bandwidth in a trunk channel, IP1810I supports load balance function. A physical port of a trunk forwards a packet only if the trunk ID of the packet matches the ID setting of the port. That is, when a packet is forwarded to a port in a trunk, its destination port is according to trunk ID.

IP1810I provides 8 kinds of hashing methods to support trunk function, user can select one of hashing methods to generate hashing ID by programming page 0x01 register 0x1C[2:0], the hashing ID0 means to select first port of the trunk group, hashing ID1 means to select second port of the trunk group, etc.

The sequence of the hashing methods can be set by programming page 0x01 register 0x1C[3], the behavior is illustrated as follows.

Case 1: disable sequence option: if incoming packet is not TCP/UDP packet and hashing method of the trunk group is set to TCP/UDP/SP, the hashing method will change to use DA/SA. If incoming packet is not TCP/UDP packet and hashing method of the trunk group is set to TCP/UDP/DP, the hashing method will change to use DA.

Case 2: enable sequence option: If incoming packet is not TCP/UDP packet and hashing method of the trunk group is set to TCP/UDP/SP, and then the hashing method will change depending on the type of packet. If the packet is IP packet, the hashing method will change to use SIP. If the packet is not IP packet, the hashing method will change to use DA/SA. If incoming packet is not TCP/UDP packet and hashing method of the trunk group is set to TCP/UDP/DP, if the packet is IP packet, the hashing method will change to use DIP. If the packet is not IP packet, the hashing method will change to use DA.

Each hashing method is described as follows

| Hashing method | Port | SA | DA | SA/DA | SIP | DIP | TCP/UDP DP | TCP/UDP DP |
|---|---|---|---|---|---|---|---|---|
| Single trunk | By port | bit[1:0] | bit[1:0] | bit[1:0] XOR bit[1:0] | bit[3 XOR 2] bit[1 XOR 0] | | | |
| Double trunk | By port | bit[2:0] | bit[2:0] | bit[2:0] XOR bit[2:0] | bit[2 XOR 1]; bit[3 XOR 2]; bit[1 XOR 0] | | | |

If the destination port of a trunk is un-link, the packet will be forward the port shifted by -1. If the port is un-link, too, the packet will be forward the port shifted by +2. For example, if port 4 is un-link, its packet will be forwarded to port 3. If port 3 is un-link, too, the packet will be forwarded to port 1.

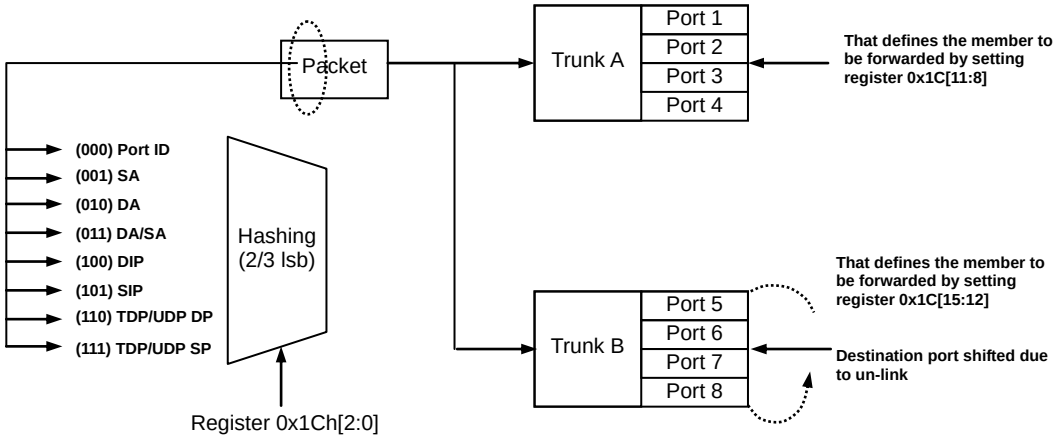



**Figure 19 Load Balance Block Diagram**

If the double trunk group is selected, the trunk hashing algorithm is changed from 2 bit to 3 bit, single trunk group uses 2 bits hashing algorithm to generate Hashing ID, ranging from 0 to 3, double trunk group use 3 bits hashing algorithm to generate Hashing ID, ranging from 0 to 7.

## 5.10 Spanning Tree

| | | |
|---|---|---|
| **Related registers** | **0x06 [1:0]** | **PAGE 0x00** |
| | **0x57 to 0x96** | **PAGE 0x01** |
| | **0x03[1:0], 0x04[15:0]** | **PAGE 0x0C** |

### 5.10.1 BPDU Packet Forwarding

IP1810I support spanning tree function with the following steps:

1. Detect BPDU frames by examining multicast address (01-80-c2-00-00-00).
2. Forward BPDU packets to CPU and add special tag with port information.
3. Special tag type is defined in page 0x0C register 04[15:0] and the special tag enable by writing “1” to page 0x0C register 0x03[1]. The BPDU packets is forwarded to CPU port by writing “01” to page 0x00 register 0x06[1:0] and CPU port is enabled by writing “1” to page 0x0C register 0x03[0].

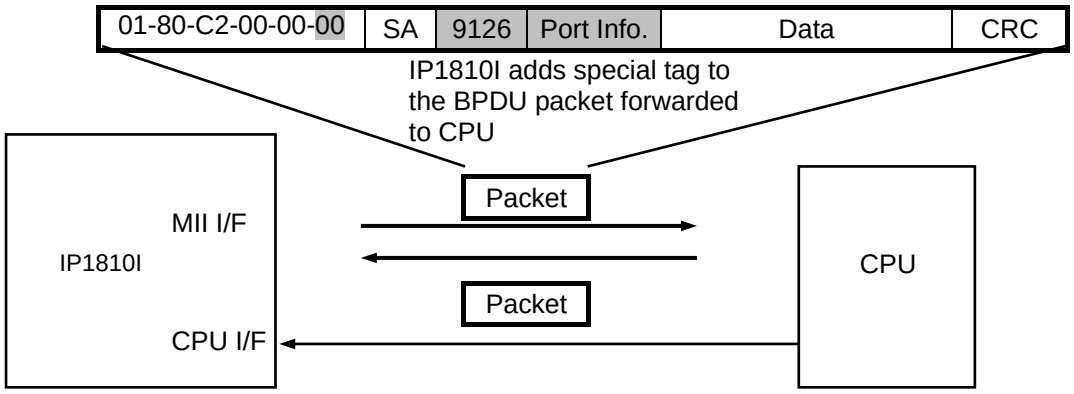



**Figure 20 BPDU Packet Forwarding Block Diagram**

### 5.10.2 Port States

| | | |
|---|---|---|
| **Related registers** | **0x57[15:0] to 0x96[9:8][1:0]** | **PAGE 0x01** |
| | **0x06[6], 0xD2 to 0xD8** | **PAGE 0x02** |

To support IEEE802.1D spanning tree protocol, each port of IP1810I supports four states (discard state, blocking state, learning state, forwarding state), is shown in the following table. Each port of IP1810I can be set in one of the four spanning tree states individually by programming page 0x01 register 0x57[15:0] and 0x5A[9:8][1:0] for STP and page 0x01 register 0x57[15:0] to 0x96[9:8][1:0] for MSTP.

IP1810I also supports multiple STP. IP1810I will use MSTP registers to set up state for each port, IP1810I use FID field of VLAN table to mapping to corresponding register. IP1810I provides 16 set of VLAN settings for MSTP.

**Table 27 Port States Selection Table**

| Status in 802.1D | Corresponding function (register) | | | |
|---|---|---|---|---|
| | BPDU packets | | Normal packets | |
| | Port_RX(to CPU) | Port_TX(from CPU) | Port_RX (to CPU) | Port_TX(from CPU) |
| Discard(00) | X | X | X | X |
| Blocking(01) | O | X | X | X |
| Learning(10) | O | O | learning | X |
| Forwarding(11) | O | O | O | O |

Note: O; means " forward", X: means " not forward"

## 5.11 IGMP Snooping

| Related registers | 0x29 to 0x56 | PAGE 0x01 |
|---|---|---|
| | 0x03[0] | PAGE 0x0C |

IP1810I supports IGMPv1, v2 snooping and IGMPv3 aware specified in RFC1112,2236 and 3376 respectively. Because IGMP is used between hosts and multicast routers, IP1810I listens the IGMP messages that communicate between router and host to establish multicast group membership. Based on the group membership information, IP1810I forwards IP multicast data to its members which is registered in group table. IGMP snooping function can be configured by page 0x01 register 0x29 to 0x56. For hardware IGMPv1, v2 snooping, IP1810I receives query, report and leave packets to build multicast table. IP1810I also has an option to support timeout mechanism, If a host port is time expired in this group, the host port will leave the multicast group.

Except for hardware IGMPv1, v2 snooping, IP1810I also supports software IGMP snooping and IGMP snooping with CPU assistance. Software IGMP snooping implies that CPU must handle IP multicast traffic which includes IGMP packet and IP multicast data packet, and then forwards them to multicast group member after processing done. For the IGMP snooping with CPU assistance, the switch directly forwards the report and leave packets into CPU port, and then CPU will build multicast table according to report and leave packets. Besides, multicast data packet will be forwarded according to multicast table by IP1810I.

IP1810I supports not only IGMP snooping for IPv4 but also MLD snooping for IPv6. MLDv1 is similar to IGMPv2 and MLDv2 is similar to IGMPv3. When the IGMPv3 or MLDv2 multicast packets are coming, they will be forwarded depending on multicast table and source list table to forward packets by CPU.

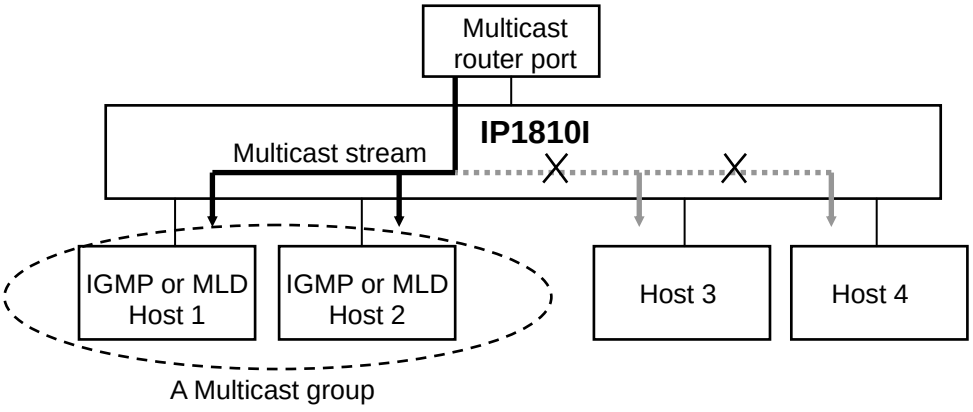



**Figure 21 IGMP Snooping Block Diagram**

## 5.12 IRMP(ICPLUS remote management protocol)

| Related registers | 0x01 to 0x0B | PAGE 0x0A |
|---|---|---|

IP1810I implements IRMP for accessing register in remote condition, IRMP is ICPLUS remote management protocol, all IP1810I registers can be easily accessed via running the IRMP protocol. All configurations of the IRMP protocol can be set by writing page 0x0A register 0x01 to 0x0B. In order to accomplish remote control, IRMP uses handshaking mechanism to complete the all IRMP actions, such as hello request /acknowledge action, login request /acknowledge, etc. The IRMPPDU frame structure is illustrated as follows (detail description please refers to application note).

**Table 28 ICPLUS remote management protocol format**

| DA (6B) = Destination MAC address | | | SA (6B) = Source MAC address | | | | Service tag (4B) (optional) **[0x88A8_xxxx]** | |
|---|---|---|---|---|---|---|---|---|
| Customer tag (4B) (optional)**[0x8100_xxxx]** | | EtherType (2B) **[0x8931]** | SubType (2B) **[0xFFFE]** | Flag (1B) | Op (1B) | Vender ID (4B) **[0x0090_C300]** | | Device ID(2B) **[0x8290]** |
| Password (4B) or Port information | | Conn. ID(2B) | Reg Add 0(2B) | Reg Data 0(2B) | | Reg Add 1(2B) | Reg Data 1(2B) | Reg Add 2(2B) |
| Reg Data 2(2B) | Reg Add 3(2B) | Reg Data 3(2B) | PAD [0x00] | | | | CRC (4B) | |

## 5.13 IPv6 related functions configuration

| Related registers | 0xA7 to 0xAC | PAGE 0x00 |
|---|---|---|

IP1810I also supports IPv6 protocol, IP1810I examines IPv6 header to support specific purpose for IPv6 protocol. IP1810I will check next header in IPv6 header, if it matches, IPv6 packet will be forwarded, dropped, “To CPU and DA”, or To CPU by programming page 0x00 register 0xA8 to 0xAC, IP1810I supports the following header table.

**Table 29 IPv6 Header**

| Order | Header type | Next header code(DEC) |
|---|---|---|
| 1 | Fragment Header | 44 |
| 2 | Encapsulation header | 50 |
| 3 | Authentication Header | 51 |
| 4 | ICMPv6 | 58 |
| 5 | User-defined header | xx |

NDP (neighbor discovery protocol) is defined in RFC 2461. It uses ICMPv6 to exchange the messages necessary for its functions.

**Table 30 NDP Header**

| Order | Type | Next header code(DEC) |
|---|---|---|
| 1 | Router solicitation | 133 |
| 2 | Router Advertisement | 134 |
| 3 | Neighbor solicitation | 135 |
| 4 | Neighbor Advertisement | 136 |
| 5 | redirect | 137 |

MLD (Multicast Listener Discovery) the protocol is embedded in ICMPv6 instead of using a separate protocol. MLDv1 is similar to IGMPv2 and MLDv2 is similar to IGMPv3. The protocol is described in RFC 3810.

**Table 31 MLD Header**

| Order | Type | Next header code(DEC) |
|---|---|---|
| 1 | MLDv1 query | 130 |
| 2 | MLDv1 listener report | 131 |
| 3 | MLDv1 listener done | 132 |
| 4 | MLDv2 listener report | 143 |

## 5.14 ACL

| | | |
|---|---|---|
| **Related registers** | **0x52[7:0], 0x53[12][8]** | **PAGE 0x00** |
| | **0x24 & 0x99** | **PAGE 0x01** |
| | **0x8A[7:0] to 0xED** | **PAGE 0x02** |
| | **0x01[7:0] to 0x6F[12][8]** | **PAGE 0x07** |

IP1810I supports 128-entries for ACL rule. The ACL function is enabled if page 0x01 register 0x24[0] is set to “1” , When a packet is received, IP1810I will examine whether the incoming packet meet in specific ACL rule entries in mode table, each mode comprises of some rules, such as SIP, DIP, TCP/UDP, SMAC, DMAC, VLAN and physical port, and so on. If it matches, the packet will act according to corresponding ACT entry.

## 5.15 OAM

| | | |
|---|---|---|
| **Related registers** | **0x01 to 0x08[12][8]** | **PAGE 0x04** |

IEEE802.3ah OAM (Operation, Administration and Maintenance) is a point to point slow protocol and is used to maintain the health of the link. The main functions are provided as follow :

- Auto Discovery
- Fault Indication
- Remote LoopBack Test

| 6 bytes | 6 bytes | 2 bytes | 1 bytes | 2 bytes | 1 bytes | | |
|---|---|---|---|---|---|---|---|
| DA(01-80-C2-00-00-02) | SA | 8809 | 03 | flags | code | Data | FCS |

**Figure 22 802.3 OAM Frame Format**

IP1810I supports OAM functions, including Auto Discovery/ Fault Indication/ Remove LoopBack Test/ Remote R/W registers.

IP1810I OAM functions and controls are set via accessing the indirect register per port. Detail information please refers to the OAM application note.

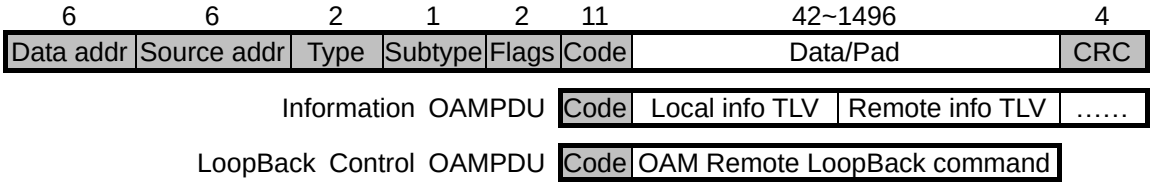



**Figure 23 OAMPDU messages Frame Format**

## 5.16 Special tag

| Related registers | 0x41[13] | PAGE 0x00 |
|---|---|---|
| | 0x03[1:0], 0x04 | PAGE 0x0C |

The purpose of special tag is:

- To allow a frame (switch to CPU) to carry ingress port number ,Queue ID ,PTP and sFlow information carriers in special tag header
- To allow a frame (CPU to switch ) to indicate the output port mask and output queue number carriers in special tag header

### 5.16.1 Special Tag for TX (From CPU to switch)

Frame direction is from CPU to switch. IP1810I enables special tag function by programming register page 0x0C register 0x03[1]. This function provides dedicated forwarding port decision and priority assignment information. This information is embedded in special tag header, which can be set by CPU. The special data will be removed by switch itself when the dedicated port had received packet, if DFP is set to 0 , the frame will be based on its DA to be forwarded.

| DA | SA | Special tag | Special data |
|---|---|---|---|
| | | 0x9126 | (1) 4 Bytes |

| Special data (1) | | | | | | |
|---|---|---|---|---|---|---|
| Special tag data (4bytes) | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26-1 | 0 |
| **Dedicated priority queue (DPQ)** | | | **DFP enable** | Dedicated forwarding port map (DFP) | | |

DPQ Enable bit : 0x41[13] page0 (0:Q0 , 1:Q1 , …. , 7: Q7)

**Figure 24 Special Tag for TX Frame Format (From CPU to switch)**

### 5.16.2 Special Tag for RX (From switch to CPU)

Frame direction is from switch to CPU. IP1810I enables special tag function by programming page 0x0C register 0x03[1]. The special tag information consists of source port number and QID, if the incoming packet with OAM information is forwarded to CPU, CPU will showed off OAM packet Flag field and code, these information carrier in special tag header 0x9126 (Page 0x0C Reg. 0x04).

| DA | SA | Special tag | Special data | | |
|---|---|---|---|---|---|
| | | 0x9126 | (1) 4 Bytes | | |
| | | 0x9127(+1) | (1) 4 Bytes | **(3) 8 bytes PTP stamp info** | |
| | | 0x9128(+2) | (1) 4 Bytes | **(2) 28 bytes sFlow info** | |
| | | 0x9129(+3) | (1) 4 Bytes | **(2) 28 bytes sFlow info** | **(3) 8 bytes PTP stamp info** |

Special data (1)

| Special tag data (4bytes) | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| **PTP/ normal (0)** | | | **Stamp** | **U/M** | **ACL redirect** | **Snif** | **U-SA** | **CCPU** | **DLF** | **PTP** | **SEQ3** | **SEQ2** | **SEQ1** | **SEQ0** | - |
| **CFM opcode (6,5,4,3,2,1)** | | | **U/M** | | | | | | | **MD_Level** | | | - | | |
| **OAM (7)** | | | **OAMPDU Flag field** | | | | | | | **OAMPDU code** | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| - | | | | | | | | Queue ID | | | Source port | | | | |

| | | |
|---|---|---|
| CFM opcode&OAM &others | 7: OAM | 6: others CFM opcode |
| | 5: LTM | 4: LTR |
| | 3: LBM | 2: LBR |
| | 1: CCM | 0: PTP or normal packet (not include CFM,OAM) |
| OAMPDU Flag & code | OAMPDU Flag | OAMPDU code |
| | bit[6]:remote stable | 6.7 : others |
| | bit[5]:remote evaluating | 5: organization specific |
| | bit[4]:local stable | 4: LoopBack control |
| | bit[3]:local evaluating | 3: variable request |
| | bit[2]:critical event | 2: variable response |
| | bit[1]:dying gasp | 1: Event notification |
| | bit[0]:link fault | 0: information |
| U/M | Uni port / Multi port output | |
| ACL redirect | ACL action redirect | |
| Snif | Sniffer packet | |
| U-SA | Unknown SA | |
| CCPU | Capture to CPU | |
| DLF | Destination lookup Fail | |
| STAMP | PTP packet is stamped | |
| PTP | PTP packet [31:29] =0 , PTP packet [21]=1 | |
| SEQ0,1,2,3 | PTP RX stamp sequence | |
| QID | Out Queue from which Queue ID | |
| SP | Source port | |

**Figure 25 Special Tag for RX Frame Format (From switch to CPU)**

If PTP special tag data is enabled by programming ”1” to page 0x07 register 0x7A[5] ,the PTP special data will be added to incoming packet ,and then the packet will be forwarded to CPU, the special tag is 0x9127. If sFlow special tag data is enabled by programming “1” to page 0x07 register 0x7A[4], the sFlow special data will be added to incoming packet ,and then the packet will be forwarded to CPU, the special tag is 0x9128. If PTP and sFlow special tag data is enabled by programming “11” to page 0x07 register 0x7A[5:4], the both PTP and sFlow special data will be add to incoming packet ,and then the packet will be forwarded to CPU, the special tag is 0x9129.

Special data (2)

| Special tag data (28bytes) | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 223 | 222 | 221 | 220 | 219 | 218 | 217 | 216 | 215 | 214 | 213 | 212 | 211 | 210 | 209 | 208 |
| Customer tag Remove/Add [57:42] | | | | | | | | | | | | | | | |
| 207 | 206 | 205 | 204 | 203 | 202 | 201 | 200 | 199 | 198 | 197 | 196 | 195 | 194 | 193 | 192 |
| Customer tag Remove/Add [41:26] | | | | | | | | | | | | | | | |
| 191 | 190 | 189 | 188 | 187 | 186 | 185 | 184 | 183 | 182 | 181 | 180 | 179 | 178 | 177 | 176 |
| Customer tag Remove/Add [25:10] | | | | | | | | | | | | | | | |
| 175 | 174 | 173 | 172 | 171 | 170 | 169 | 168 | 167 | 166 | 165 | 164 | 163 | 162 | 161 | 160 |
| Customer tag Remove/Add [9:0] | | | | | | | | | | arl_route_result [28:23] | | | | | |
| 159 | 158 | 157 | 156 | 155 | 154 | 153 | 152 | 151 | 150 | 149 | 148 | 147 | 146 | 145 | 144 |
| arl_route_result [22:7] | | | | | | | | | | | | | | | |
| 143 | 142 | 141 | 140 | 139 | 138 | 137 | 136 | 135 | 134 | 133 | 132 | 131 | 130 | 129 | 128 |
| arl_route_result [6:0] | | | | | | | PRI [2:0] | | | Mac rx count [14:9] | | | | | |
| 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 | 112 |
| Mac rx count [8:0] | | | | | | | | | VID[11:5] | | | | | | |
| 111 | 110 | 109 | 108 | 107 | 106 | 105 | 104 | 103 | 102 | 101 | 100 | 99 | 98 | 97 | 96 |
| VID [4:0] | | | | | ACL SF | Egress SF | Ingress SF | SF_Rate [15:8] | | | | | | | |
| 95 | 94 | 93 | 92 | 91 | 90 | 89 | 88 | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 |
| SF_Rate [7:0] | | | | | | | | 0 | | | | | | | |
| 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 | 64 |
| 0 | | | | | | Service tag Remove/Add [57:48] | | | | | | | | | |
| 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| Service tag Remove/Add [47:32] | | | | | | | | | | | | | | | |
| 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| Service tag Remove/Add [31:16] | | | | | | | | | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Service tag Remove/Add [15:0] | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | | | | | | | | | | | | PPPoE remove | QID | | |

Special data (3)

| Special tag data (8bytes) | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| Time stamp second [31:16] | | | | | | | | | | | | | | | |
| 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| Time stamp second [15:0] | | | | | | | | | | | | | | | |
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| full | 0 | Time stamp nano second [29:16] | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Time stamp nano second [15:0] | | | | | | | | | | | | | | | |

## 5.17 sFlow

| Related registers | 0x97[7:0] to 0xD8 | PAGE 0x01 |
|---|---|---|
| | 0x7A[3:0] | PAGE 0x07 |

IP1810I supports sFlow function, which can monitor the content of the packet, IP1810I provides various methods to sample sFlow packet to CPU port, such as inbound, outbound, VID, ACL and MAC Lookup table, which can be set by programming page 0x01 register 0x99, 0x9A[8:0], 0x9D[14:12] and 0x9E[11:9]. sFlow can be configured individually for each port, such as sampling rate, etc. IP1810I also provides global counter or independent counter, which can be set by programming page 0x01 register 0x9E[12]. For example, if global counter is set by programming page 0x01 register 0x9E[12] to "1" and inbound method is set by programming the corresponding bits of page 0x01 register 0x99, 0x9A[8:0], 0x9D[14:12] and 0x9E[11:9] to "000", packets start to accumulate for all ports during incoming packet is received, when accumulated packets reach threshold value, then one sampled packet will be forwarded to CPU port.

Besides, IP1810I also supports length cutoff mechanism for sFlow packet. According to the setting of the cutoff in page 0x07 register 0x7A[3:0]. If the sFlow packet length is less than the setting of the length cutoff, the sFlow packet length will keep unchanged. If the sFlow packet length is more than the setting of the length cutoff, the sFlow packet length will be cutoff.

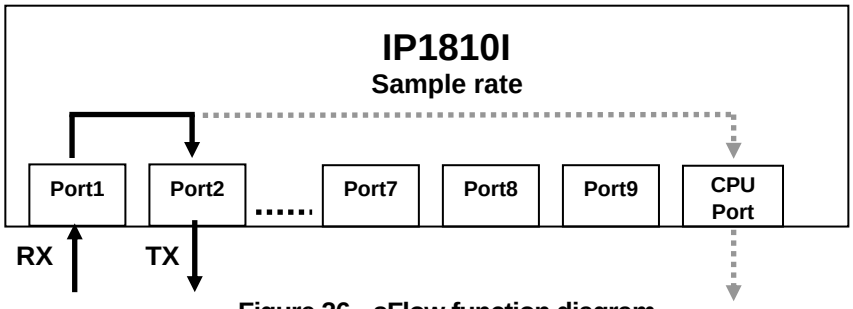



**Figure 26 sFlow function diagram**

## 5.18 WOL+

IP1810I provides WOL+ function for WOL+ port that can speed down to 10Mbps in the idle status and recover to the previous speed status by detecting WOL+ events. For example, if WOL+ port have no activity exceeds pre-defined idle time, the WOL+ ports will link to 10Mbps, If IP1810I detects WOL+ events in down speed period, IP1810I will change the speed status of ports to 100Mbps from 10Mbps for WOL+ port 1~8 and change the speed status of the port to 1000Mbps/100Mbps from 10Mbps for WOL+ port 9.

The WOL+ function is enabled and disabled for port 1~8 by writing both page 0 register 0xAD[7:0] and PHY's page 4 register 0x16[13]. WOL+ function for port 9 are enabled and disabled by writing both page 0 rigister 0xAE[8] and page 3 register 0x10[0], the WOL+ event for port9 also be configured as detecting magic packet and any packet by writing page 3 register 0x10[5:4].

IP1810I also supports 4 kinds of time units, the unit of idle time can be set to 00: " disable", 01: "10s", 10: "1min", and 11:"10 min"by writing page 0 register 0xAF[6:5], if the idle time of the WOL+ port reaches threshold , the WOL+ port will link to 10Mbps.

IP1810I also provides WOL+ master and slave modes, only the difference between master mode and slave mode. In slave mode, WOL+ port is enabled by writing page 0 register 0xB0[7:0] and 0xB1[12][8], and reading page 0 register 0xB2[7:0] and 0xB3[12][8] for WOL+ port status to complete WOL+ action by CPU.In master mode, HW can automatically finish the WOL+ action.

# 6 Register Description

## 6.1 PHY Register Map

Read / Write PHY register by page 3 register 0x13, 0x14

**Table 32 PHY Register Map**

| Page | PHY address | Register | Description | Default |
|---|---|---|---|---|
| 0 | PHY 8~15 | 0 | Control Register | |
| 0 | PHY 8~15 | 1 | Status Register | |
| 0 | SHARE X 1 | 2 | PHY Identifier 1 Register | |
| 0 | SHARE X 1 | 3 | PHY Identifier 2 Register | |
| 0 | PHY 8~15 | 4 | Auto-Negotiation Advertisement Register | |
| 0 | PHY 8~15 | 5 | Auto-Negotiation Link Partner Ability Register | |
| 0 | PHY 8~15 | 6 | Auto-Negotiation Expansion Register | |
| 0 | PHY 8~15 | 7 | Auto-Negotiation Next Page Transmit Register | |
| 0 | PHY 8~15 | 8 | Auto-Negotiation Link Partner Next Page Register | |
| 0 | PHY 8~15 | 13 | MMD Access Control Register | |
| 0 | PHY 8~15 | 14 | MMD Access Address Data Register | |
| 0 | PHY 8~15 | 3.0 | PCS control 1 register | |
| 0 | PHY 8~15 | 3.1 | PCS status 1 register | |
| 0 | PHY 8~15 | 3.20 | EEE capability | |
| 0 | PHY 8~15 | 3.22 | EEE wake error count | |
| 0 | PHY 8~15 | 7.60 | EEE advertisement register | |
| 0 | PHY 8~15 | 7.61 | EEE link partner ability | |
| 0 | SHARE X 1 | 16 | Special Control Register (APS) | |
| 0 | PHY 8~15 | 18 | Special Status Register | |
| X | SHARE X 1 | 20 | Page Control Register | |
| 4 | SHARE X 1 | 16 | WOL+ Control Register | |
| 0 | PHY 8~15 | 23 | MDI/MDIX Control Register | |
| 1 | PHY 8~15 | 23 | 10BaseT Control Register | |

**Share x 1**: 8 ports share the register
X8 Each port has its individual register
X: indicate do not care.

**Table 33 Register Symbol Abbreviations**

| Type | Description |
|---|---|
| R | Read |
| W | Write |
| R/W | Read/Write |
| SC | Self-Clearing |
| RO | Read Only |
| LL | Latching Low |
| LH | Latching High |

### 6.1.1 MII Register

**MII register 0 of PHY8~15 (Each PHY has its own MII register 0 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Control Register** | | | | |
| 8~15 | 0.15 | RW/ SC | Reset<br>The PHY is reset if user writes “1” to this bit. The reset period is around 2ms. User has to wait for at least 2ms to access IP1810I. | 0 |
| 8~15 | 0.14 | R/W | Loop back<br>1 = Loop back mode<br>0 = normal operation<br>When this bit set, IP1810I will be isolated from the network media, that is, the assertion of TXEN at the MII will not transmit data on the network. All MII transmission data will be returned to MII receive data path in response to the assertion of TXEN. | 0 |
| 8~15 | 0.13 | RW | Speed Selection<br>1 = 100Mbps<br>0 = 10Mbps<br>It is valid only if bit 0.12 is set to be 0. | 1 |
| 8~15 | 0.12 | RW | Auto-Negotiation(AN) Enable<br>1 = Auto-Negotiation Enable<br>0 = Auto-Negotiation Disable | 1 |
| 8~15 | 0.11 | R/W | Power Down<br>1: power down mode<br>0: normal operation | 0 |
| 8~15 | 0.10 | - | Isolate<br>IP1810I doesn’t support this function. | 0 |
| 8~15 | 0.9 | RW/ SC | Restart Auto-Negotiation<br>1 = re-starting Auto-Negotiation<br>0 = Auto-Negotiation re-start complete | 0 |
| 8~15 | 0.8 | R/W | Duplex mode<br>1 = full duplex<br>0 = half duplex<br>It is valid only if bit 0.12 is set to be 0. | 1 |
| 8~15 | 0.7 | R/W | Collision test | 0 |
| 8~15 | 0.6 | RO | Reserved | 0 |
| 8~15 | 0.5 | R/W | Unidirectional enable<br>When bit 0.12 is one or bit 0.8 is zero or bit 0.13 is zero, this bit is ignored.<br>When bit 0.12 is zero and bit 0.8 is one and bit 0.13 is one：<br>1 = Enable transmit from MII regardless of whether the PHY has determined that a valid link has been established.<br>0 = Enable transmit from MII only when the PHY has determined that a valid link has been established. | 0 |
| 8~15 | 0[4:0] | RO | Reserved | 0 |

**MII register 1 of PH8~15 (Each PHY has its own MII register 1 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Status Register** | | | | |
| 8~15 | 1.15 | RO | 100Base-T4 capable<br>1 = 100Base-T4 capable<br>0 = not 100Base-T4 capable<br>IP1810I does not support 100Base-T4. This bit is fixed to be 0. | 0 |
| 8~15 | 1.14 | RO | 100Base-X full duplex Capable<br>1 = 100Base-X full duplex capable<br>0 = not 100Base-X full duplex capable | 1 |
| 8~15 | 1.13 | RO | 100Base-X half duplex Capable<br>1 = 100Base-X half duplex capable<br>0 = not 100Base-X half duplex capable | 1 |
| 8~15 | 1.12 | RO | 10Base-T full duplex Capable<br>1 = 10Base-T full duplex capable<br>0 = not 10Base-T full duplex capable | 1 |
| 8~15 | 1.11 | RO | 10Base-T half duplex Capable<br>1 = 10Base-T half duplex capable<br>0 = not 10Base-T half duplex capable | 1 |
| 8~15 | 1.10 | RO | 100Base-T2 full duplex Capable<br>1 = 100Base-T2 full duplex capable<br>0 = not 100Base-T2 full duplex capable | 0 |
| 8~15 | 1.9 | RO | 100Base-T2 half duplex Capable<br>1 = 100Base-T2 half duplex capable<br>0 = not 100Base-T2 half duplex capable | 0 |
| 8~15 | 1.8 | RO | Extended Status | 0 |
| 8~15 | 1.7 | RO | Unidirectional ability<br>1 = PHY able to transmit from MII regardless of whether the PHY has determined that a valid link has been established.<br>0 = PHY able to transmit from MII only when the PHY has determined that a valid link has been established. | 1 |
| 8~15 | 1.6 | RO | MF preamble Suppression<br>1 = preamble may be suppressed<br>0 = preamble always required | 1 |
| 8~15 | 1.5 | RO | Auto-Negotiation Complete<br>1 = Auto-Negotiation complete<br>0 = Auto-Negotiation in progress<br>When read as logic 1, indicates that the Auto-Negotiation process has been completed, and the contents of register 4 and 5 are valid. When read as logic 0, indicates that the Auto-Negotiation process has not been completed, and the contents of register 4 and 5 are meaningless. If Auto-Negotiation is disabled (bit 0.12 set to logic 0), then this bit will always read as logic 0. | 0 |

**MII register 1 of PH8~15 (Each PHY has its own MII register 1 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Status Register** | | | | |
| 8~15 | 1.4 | RO LH | Remote fault<br>1 = remote fault detected<br>0 = not remote fault detected<br>When read as logic 1, indicates that IP1810I has detected a remote fault condition. This bit is set until remote fault condition gone and before reading the contents of the register. This bit is cleared after IP1810I reset. | 0 |
| 8~15 | 1.3 | RO | Auto-Negotiation Ability<br>1 = Auto-Negotiation capable<br>0 = not Auto-Negotiation capable<br>When read as logic 1, indicates that IP1810I has the ability to perform Auto-Negotiation. | 1 |
| 8~15 | 1.2 | RO LL | Link Status<br>1 = Link Pass<br>0 = Link Fail<br>When read as logic 1, indicates that IP1810I has determined a valid link has been established. When read as logic 0, indicates the link is not valid. This bit is cleared until a valid link has been established and before reading the contents of this registers. | 0 |
| 8~15 | 1.1 | RO LH | Jabber Detect<br>1 = jabber condition detected<br>0 = no jabber condition detected<br>When read as logic 1, indicates that IP1810I has detected a jabber condition. This bit is always 0 for 100Mbps operation and is cleared after IP1810I reset. When the duration of TXEN exceeds the jabber timer (21ms), the transmission and loop back functions will be disabled and the COL is active. After TXEN goes low for more than 500 ms, the transmitter will be re-enabled. | 0 |
| 8~15 | 1.0 | RO | Extended capability<br>1 = Extended register capabilities<br>0 = No extended register capabilities<br>IP1810I has extended register capabilities. | 1 |

**MII register 2 of PHY8~15 (8 PHYs share the MII register)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY Identifier 1 Register** | | | | |
| 8~15 | 2 | RO | IP1810I OUI (Organizationally Unique Identifier) ID, the MSB is 3rd bit of IP1810I OUI ID, and the LSB is 18th bit of IP1810I OUI ID. IP1810I OUI is 0090C3. | 16'h0243 |

**MII register 3 of PHY8~15 (8 PHYs share the MII register)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY Identifier 2 Register** | | | | |
| 8~15 | 3[15:10] | RO | PHY identifier<br>IP1810I OUI ID, the MSB is 19th bit of IP1810I OUI ID, and LSB is 24th bit of IP1810I OUI ID. | 6'h03 |
| 8~15 | 3[9:4] | RO | Manufacture's Model Number<br>IP1810I model number | 6'h01 |
| 8~15 | 3[3:0] | RO | Revision Number<br>IP1810I revision number | 0 |

**MII register 4 of PHY8~15 (Each PHY has its own MII register 4 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Auto-Negotiation Advertisement Register** | | | | |
| 8~15 | 4.15 | R/W | 1 = Next pages are supported<br>0 = Next pages are not supported | 0 |
| 8~15 | 4.14 | RO | Reserved by IEEE, write as 0, ignore on read | 0 |
| 8~15 | 4.13 | R/W | Remote Fault<br>1 = Advertises that this port has detected a remote fault.<br>0 = There is no remote fault. | 0 |
| 8~15 | 4.12 | RO | Reserved for future IEEE use, write as 0, ignore on read | 0 |
| 8~15 | 4.11 | RW | Asymmetric PAUSE<br>1 = Asymmetric flow control is supported<br>0 = Asymmetric flow control is not supported | 1 |
| 8~15 | 4.10 | RW | PAUSE<br>1 = Symmetric flow control is supported<br>0 = Symmetric flow control is not supported | 1 |
| 8~15 | 4.9 | RO | 100BASE-T4 Not supported | 0 |
| 8~15 | 4.8 | R/W | 100BASE-TX full duplex<br>1 = 100BASE-TX full duplex is supported<br>0 = 100BASE-TX full duplex is not supported | 1 |
| 8~15 | 4.7 | R/W | 100BASE-TX<br>1 = 100BASE-TX is supported<br>0 = 100BASE-TX is not supported | 1 |
| 8~15 | 4.6 | R/W | 10BASE-T full duplex<br>1 = 10BASE-T full duplex is supported<br>0 = 10BASE-T full duplex is not supported | 1 |
| 8~15 | 4.5 | R/W | 10BASE-T<br>1 = 10BASE-T is supported<br>0 = 10BASE-T is not supported | 1 |
| 8~15 | 4[4:0] | RO | Selector Field<br>Use to identify the type of message being sent by Auto-Negotiation. | 5'b00001 |

**MII register 5 of PHY8~15 (Each PHY has its own MII register 5 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Auto-Negotiation Link Partner Ability Register** | | | | |
| 8~15 | 5.15 | RO | Next Page<br>1 = Next Page ability is supported by link partner<br>0 = Next Page ability does not supported by link partner | 0 |
| 8~15 | 5.14 | RO | Acknowledge<br>1 = Link partner has received the ability data word<br>0 = Not acknowledge | 0 |
| 8~15 | 5.13 | RO | Remote Fault<br>1 = Link partner indicates a remote fault<br>0 = No remote fault indicate by link partner<br>If this bit is set to logic 1, then bit 1.4 (Remote fault) will set to logic 1. | 0 |
| 8~15 | 5.12 | RO | Reserved by IEEE for future use, write as 0, and read as 0. | 0 |
| 8~15 | 5.11 | RO | Asymmetric PAUSE<br>1 = Link partner support Asymmetric PAUSE<br>0 = Link partner does not support Asymmetric PAUSE<br>When local or link partner is Auto-negotiation disabled, this bit is read as 1. The pause resolution is determined by MII Reg4.[11:10]. | 0 |
| 8~15 | 5.10 | RO | PAUSE<br>1 = Link partner support Symmetric PAUSE<br>0 = Link partner does not support Symmetric PAUSE<br>When local or link partner is Auto-negotiation disabled, this bit is read as 1. The pause resolution is determined by MII Reg4.[11:10]. | 0 |
| 8~15 | 5.9 | RO | 100BASE-T4<br>1 = Link partner support 100BASE-T4<br>0 = Link partner does not support 100BASE-T4 | 0 |
| 8~15 | 5.8 | RO | 100BASE-TX full duplex<br>1 = Link partner support 100BASE-TX full duplex<br>0 = Link partner does not support 100BASE-TX full duplex | 0 |
| 8~15 | 5.7 | RO | 100BASE-TX<br>1 = Link partner support 100BASE-TX<br>0 = Link partner does not support 100BASE-TX | 0 |
| 8~15 | 5.6 | RO | 10BASE-T full duplex<br>1 = Link partner support 10BASE-T full duplex<br>0 = Link partner does not support 10BASE-T full duplex | 0 |
| 8~15 | 5.5 | RO | 10BASE-T<br>1 = Link partner support 10BASE-T<br>0 = Link partner does not support 10BASE-T | 0 |
| 8~15 | 5[4:0] | RO | Selector Field<br>Protocol selector of the link partner | 5'b00000 |

**MII register 6 of PHY8~15 (Each PHY has its own MII register 6 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Auto-Negotiation Expansion Register** | | | | |
| 8~15 | 6[15:5] | RO | Reserved | 0 |
| 8~15 | 6.4 | RO/ LH | Parallel Detection Fault<br>1 = a fault has been detected via parallel detection function.<br>0 = a fault has not been detected via parallel detection function. | 0 |
| 8~15 | 6.3 | RO | Link Partner Next Page Able<br>1 = Link partner is next page able.<br>0 = Link partner is not next page able.<br>In 100FX or AN disabled, then this bit is always equal to 0. | 0 |
| 8~15 | 6.2 | RO | Next Page Able<br>1 = IP1810I next page able.<br>0 = IP1810I is not next page able. | 1 |
| 8~15 | 6.1 | RO/ LH | Page Received<br>1 = A new page has been received.<br>0 = A new page has not been received. | 0 |
| 8~15 | 6.0 | RO | If AN is enabled, this bit means:<br>1 = Link partner is Auto-Negotiation able.<br>0 = Link partner is not Auto-Negotiation able.<br>In 100FX or AN disabled, then this bit is always equal to 0. | 0 |

**MII register 7 of PHY8~15 (Each PHY has its own MII register 7 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Auto-Negotiation Next Page Transmit Register** | | | | |
| 8~15 | 7.15 | RW | Next Page<br>Transmit Code Word bit 15 | 0 |
| 8~15 | 7.14 | RO | Reserved<br>Transmit Code Word bit 14 | 0 |
| 8~15 | 7.13 | RW | Message Page<br>Transmit Code Word bit 13 | 1 |
| 8~15 | 7.12 | RW | Acknowledge 2<br>Transmit Code Word bit 12 | 0 |
| 8~15 | 7.11 | RO | Toggle<br>Transmit Code Word bit 11 | 0 |
| 8~15 | 7[10:0] | RW | Message/Unformatted Field<br>Transmit Code Word bit 10:0 | 1 |

**MII register 8 of PHY8~15 (Each PHY has its own MII register 8 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Auto-Negotiation Link Partner Next Page Register** | | | | |
| 8~15 | 8.15 | RO | Next Page<br>Received Code Word bit 15 | 0 |
| 8~15 | 8.14 | RO | Acknowledge<br>Received Code Word bit 14 | 0 |
| 8~15 | 8.13 | RO | Message Page<br>Received Code Word bit 13 | 0 |
| 8~15 | 8.12 | RO | Acknowledge 2<br>Received Code Word bit 12 | 0 |
| 8~15 | 8.11 | RO | Toggle<br>Received Code Word bit 11 | 0 |
| 8~15 | 8[10:0] | RO | Message/Unformatted Field<br>Received Code Word bit 10:0 | 0 |

**MII register 16 of PHY8~15 (Every 8 PHYs share the MII register)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Special Control Register** | | | | |
| 8~15 | 16.7 | RW | Advance power saving mode<br>1 = Enable APS mode (default)<br>0 = Disable APS mode<br><br>Please refer to the Power Saving application note for more detail description. | 1 |
| 8~15 | 16.4 | RW | Far end fault function<br>1 = Far end fault function disable<br>0 = Far end fault function enable (default)<br>This bit is only used for fiber mode. | 0 |

**MII register 18 of PHY8~15 (Each PHY has its own MII register 18 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Special Status Register** | | | | |
| 8~15 | 18.14 | RO | Linkup<br>1 = linkup<br>0 = unlink | 0 |
| 8~15 | 18.11 | RO | Speed Mode<br>1 = 100 Mbps<br>0 = 10 Mbps | 0 |
| 8~15 | 18.10 | RO | Duplex Mode<br>1 = Full Duplex<br>0 = Half Duplex | 0 |

The other Registers are reserved registers. User is inhibited to access to these registers. It may introduce abnormal function to write these registers.

### 6.1.2 MMD Control Register

**MII register 13 of PHY8~15 (Each PHY has its own MII register 13 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **MMD Access Control Register** | | | | |
| 8~15 | 13[15:14] | R/W | Function<br>00 = address<br>01 = data, no post increment<br>10 = data, post increment on reads and writes<br>11 = data, post increment on writes only | 0 |
| 8~15 | 13[13:5] | R/W | Reserved<br>Write as 0, ignore on read | 0 |
| 8~15 | 13[4:0] | R/W | DEVAD<br>Device Address | 0 |

**MII register 14 of PHY8~15 (Each PHY has its own MII register 14 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **MMD Access Address Data Register** | | | | |
| 8~15 | 14[15:0] | R/W | Address Data<br>If 13.15:14 = 00, MMD DEVAD's address register.<br>Otherwise, MMD DEVAD's data register as indicated by the contents of its address register | 0 |

Example 1, Read 0.3.20 (Read Data from MMD register 3.20 of PHY address 0):

1. Write 0.13 = 0x0003 //MMD DEVAD 3
2. Write 0.14 = 0x0014 //MMD Address 20
3. Write 0.13 = 0x4003 //MMD Data command for MMD DEVAD 3
4. Read 0.14 //Read MMD Data from 0.3.20

Example 2, Write 1.7.60 = 0x3210 (Write 0x3210 Data to MMD register 7.60 of PHY address 1):

1. Write 1.13 = 0x0007 //MMD DEVAD 7
2. Write 1.14 = 0x003C //MMD Address 60
3. Write 1.13 = 0x4007 //MMD Data command for MMD DEVAD 7
4. Write 1.14 = 0x3210 //Write MMD Data 0x3210 to 1.7.60

### 6.1.3 MMD Data Register

**MMD register 3.0 of PHY8~15 (Each PHY has its own MMD register 3.0 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PCS control 1 Register** | | | | |
| 8~15 | 3.0[15:11] | RO | Reserved<br>Ignore when read | 0 |
| 8~15 | 3.0.10 | R/W | Clock stop enable<br>1 = PHY may stop xMII Rx clock during LPI<br>0 = Clock not stoppable | 0 |
| 8~15 | 3.0[9:0] | RO | Reserved<br>Ignore when read | 0 |

**MMD register 3.1 of PHY8~15 (Each PHY has its own MMD register 3.1 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PCS status 1 Register** | | | | |
| 8~15 | 3.1[15:12] | RO | Reserved<br>Ignore when read | 0 |
| 8~15 | 3.1.11 | RO/LH | Tx LPI received<br>1 = Tx PCS has received LPI<br>0 = LPI not received | 0 |
| 8~15 | 3.1.10 | RO/LH | Rx LPI received<br>1 = Rx PCS has received LPI<br>0 = LPI not received | 0 |
| 8~15 | 3.1.9 | RO | Tx LPI indication<br>1 = Tx PCS is currently receiving LPI<br>0 = PCS is not currently receiving LPI | 0 |
| 8~15 | 3.1.8 | RO | Rx LPI indication<br>1 = Rx PCS is currently receiving LPI<br>0 = PCS is not currently receiving LPI | 0 |
| 8~15 | 3.1.7 | RO | Reserved<br>Ignore on read | 0 |
| 8~15 | 3.1.6 | RO | Clock stop capable<br>1 = The MAC may stop the xMII Tx clock during LPI<br>0 = Clock not stoppable | 0 |
| 8~15 | 3.1[5:0] | RO | Reserved<br>Ignore when read | 0 |

**MMD register 3.20 of PHY8~15 (Each PHY has its own MMD register 3.20 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **EEE capability Register** | | | | |
| 8~15 | 3.20[15:7] | RO | Reserved<br>Ignore when read | 0 |
| 8~15 | 3.20.6 | RO | 10GBASE-KR EEE<br>1 = EEE is supported for 10GBASE-KR<br>0 = EEE is not supported for 10GBASE-KR | 0 |
| 8~15 | 3.20.5 | RO | 10GBASE-KX4 EEE<br>1 = EEE is supported for 10GBASE-KX4<br>0 = EEE is not supported for 10GBASE-KX4 | 0 |
| 8~15 | 3.20.4 | RO | 1000BASE-KX EEE<br>1 = EEE is supported for 1000BASE-KX<br>0 = EEE is not supported for 1000BASE-KX | 0 |
| 8~15 | 3.20.3 | RO | 10GBASE-T EEE<br>1 = EEE is supported for 10GBASE-T<br>0 = EEE is not supported for 10GBASE-T | 0 |
| 8~15 | 3.20.2 | RO | 1000BASE-T EEE<br>1 = EEE is supported for 1000BASE-T<br>0 = EEE is not supported for 1000BASE-T | 0 |
| 8~15 | 3.20.1 | RO | 100BASE-TX EEE<br>1 = EEE is supported for 100BASE-TX<br>0 = EEE is not supported for 100BASE-TX | 1 |
| 8~15 | 3.20.0 | RO | Reserved<br>Ignore when read | 0 |

**MMD register 3.22 of PHY8~15 (Each PHY has its own MMD register 3.22 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **EEE wake error count** | | | | |
| 8~15 | 3.22[15:0] | RO | EEE wake error count<br>Count wake time faults where IP1810I fails to complete its normal wake sequence within the time required for the specific PHY type. This register keeps the value before reading the contents of the register. | 0x0000 |

**MMD register 7.60 of PHY8~15 (Each PHY has its own MMD register 7.60 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **EEE advertisement Register** | | | | |
| 8~15 | 7.60[15:7] | RO | Reserved<br>Ignore when read | 0 |
| 8~15 | 7.60.6 | RO | 10GBASE-KR EEE<br>1 = Advertise that the 10GBASE-KR has EEE capability<br>0 = Do not advertise that the 10GBASE-KR has EEE capability | 0 |
| 8~15 | 7.60.5 | RO | 10GBASE-KX4 EEE<br>1 = Advertise that the 10GBASE-KX4 has EEE capability<br>0 = Do not advertise that the 10GBASE-KX4 has EEE capability | 0 |
| 8~15 | 7.60.4 | RO | 1000BASE-KX EEE<br>1 = Advertise that the 1000BASE-KX has EEE capability<br>0 = Do not advertise that the 1000BASE-KX has EEE capability | 0 |
| 8~15 | 7.60.3 | RO | 10GBASE-T EEE<br>1 = Advertise that the 10GBASE-T has EEE capability<br>0 = Do not advertise that the 10GBASE-T has EEE capability | 0 |
| 8~15 | 7.60.2 | RO | 1000BASE-T EEE<br>1 = Advertise that the 1000BASE-T has EEE capability<br>0 = Do not advertise that the 1000BASE-T has EEE capability | 0 |
| 8~15 | 7.60.1 | R/W | 100BASE-TX EEE<br>1 = Advertise that the 100BASE-TX has EEE capability<br>0 = Do not advertise that the 100BASE-TX has EEE capability | 1 |
| 8~15 | 7.60.0 | RO | Reserved<br>Ignore when read | 0 |

**MMD register 7.61 of PHY8~15 (Each PHY has its own MMD register 7.61 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **EEE link partner ability** | | | | |
| 8~15 | 7.61[15:7] | RO | Reserved<br>Ignore when read | 0 |
| 8~15 | 7.61.6 | RO | 10GBASE-KR EEE<br>1 = Link partner is advertising EEE capability for 10GBASE-KR<br>0 = Link partner is not advertising EEE capability for 10GBASE-KR | 0 |
| 8~15 | 7.61.5 | RO | 10GBASE-KX4 EEE<br>1 = Link partner is advertising EEE capability for 10GBASE-KX4<br>0 = Link partner is not advertising EEE capability for | 0 |
| 8~15 | 7.61.4 | RO | 1000BASE-KX EEE<br>1 = Link partner is advertising EEE capability for 1000BASE-KX<br>0 = Link partner is not advertising EEE capability for 1000BASE-KX | 0 |
| 8~15 | 7.61.3 | RO | 10GBASE-T EEE<br>1 = Link partner is advertising EEE capability for 10GBASE-T<br>0 = Link partner is not advertising EEE capability for 10GBASE-T | 0 |
| 8~15 | 7.61.2 | RO | 1000BASE-T EEE<br>1 = Link partner is advertising EEE capability for 1000BASE-T<br>0 = Link partner is not advertising EEE capability for 1000BASE-T | 0 |
| 8~15 | 7.61.1 | RO | 100BASE-TX EEE<br>1 = Link partner is advertising EEE capability for 100BASE-TX<br>0 = Link partner is not advertising EEE capability for 100BASE-TX | 0 |
| 8~15 | 7.61.0 | RO | Reserved<br>Ignore when read | 0 |

The other Registers are reserved registers. User is inhibited to access to these registers. It may introduce abnormal function to write these registers.

### 6.1.4 Register Page mode Control Register

**MII register 20 of PHY8~15 (Every 8 PHYs share the MII register)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| **Page Control Register** | | | | |
| 8~15 | 20[4:0] | RW | Reg16~31_Page_Sel[4:0] | 00000 |

The other Registers are reserved registers. User is inhibited to access to these registers. It may introduce unexpected function to write these registers.

### 6.1.5 WOL+ Control Register

**MII page4 register16 of PHY8~15 (Every 8 PHYs share the MII register)**

| page | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY WOL+ Control Register** | | | | |
| 4 | 16.[15] | RW | WOL+ Interrupt Enable<br>Set high to enable WOL+ interrupt<br>1=Enable<br>0=Disable<br>Each PHY address can access the register of the corresponding port. | 0 |
| | 16.[14] | RW | WOL+ Level Trigger<br>This bit is used to select the output mode of WOL+ interrupt.<br>1=Level trigger (Low goes high or high goes low when WOL+ interrupt)<br>0=Edge trigger (Positive pulse or negative pulse when WOL+ interrupt) | 1 |
| | 16.[13] | RW | WOL+ Positive Trigger<br>This bit is used to select the polarity of WOL+ interrupt.<br>1=Low goes high or positive pulse<br>0=High goes low or negative pulse | 0 |
| | 16.[12] | RW | Sense Link Change<br>Set high to enable WOL+ interrupt when link change is sensing.<br>1=Enable<br>0=Disable | 1 |
| | 16.[11] | RW | Sense Magic Packet<br>Set high to enable WOL+ interrupt when magic packet is receiving.<br>1=Enable<br>0=Disable | 1 |
| | 16.[10] | RW | Sense Any Packet<br>Set high to enable WOL+ interrupt when any packet is receiving.<br>1=Enable<br>0=Disable | 1 |
| | 16.[9] | RW | Sense DUT<br>Set high to enable WOL+ interrupt when DUT is sensing WOL+ event.<br>1=Enable<br>0=Disable<br>Each PHY address can access the register of the corresponding port. | 1 |
| | 16.[8] | RW | WOL+ Down Speed Enable<br>Set high to enable WOL+ down speed function<br>1=Enable<br>0=Disable | 1 |
| | 16.[7:1] | RO | Reserved | 0x00 |

**MII page4 register16 of PHY8~15 (Every 8 PHYs share the MII register)**

| page | MII | R/W | Description | Default |
|---|---|---|---|---|
| | 16.[0] | RO | PHY WOL+ Interrupt Status<br>The status of PHY WOL+ interrupt is based on the setting of Reg16 Page4 bit14 and bit13.<br>Each PHY address can access the register of the corresponding port. | 1 |

**MII page5 register16 of PHY8~15 (Every 8 PHYs share the MII register)**

| page | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY WOL+ MAC Address Register 0** | | | | |
| 5 | 16[15:0] | R/W | WOL+ MAC Address 0 (the most significant word)<br>WOL+ MAC Address = {WOL+_MAC_Address_0, WOL+_MAC_Address_1, WOL+_MAC_Address_2} | 0x0000 |

**MII page5 register16 of PHY8~15 (Every 8 PHYs share the MII register)**

| page | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY WOL+ MAC Address Register 1** | | | | |
| 5 | 16[15:0] | R/W | WOL+ MAC Address 1<br>WOL+ MAC Address = {WOL+_MAC_Address_0, WOL+_MAC_Address_1, WOL+_MAC_Address_2} | 0x0000 |

**MII page5 register16 of PHY8~15 (Every 8 PHYs share the MII register)**

| page | MII | R/W | Description | Default |
|---|---|---|---|---|
| **PHY WOL+ MAC Address Register 2** | | | | |
| 5 | 16[15:0] | R/W | WOL+ MAC Address 2 (the least significant word)<br>WOL+ MAC Address = {WOL+_MAC_Address_0, WOL+_MAC_Address_1, WOL+_MAC_Address_2} | 0x0000 |

Above three definitions are in the same register. Continuously write/read this register 3 times to set/obtain WOL Magic packet MAC address.

The first write/read：WOL+ MAC Address 0.
The second write/read：WOL+ MAC Address 1.
The third write/read：WOL+ MAC Address 2.

The other Registers are reserved registers. User is inhibited to access to these registers. It may introduce abnormal function to write these registers.

Example 1, Read page3 register16 (Read Data from page3 register16 of PHY address 0):

1. Write 0.20 = 0x0003 //page3
2. Read 0.16 //Read Data from page3 register16
3. Write 0.20 = 0x0000 //restore to page0

Example 2, Write page3 register16 = 0x3400 (Write Data 0x3400 to page3 register16 of PHY address 0):

1. Write 0.20 = 0x0003 //page3
2. Write 0.16 = 0x3400 //Write Data 0x3400 to page3 register16
3. Write 0.20 = 0x0000 //restore to page0

### 6.1.6 MDI/MDIX Control Register

**MII register 23 of PHY8~15 (Each PHY has its own MII register 23 with different PHY address)**

| PHY | MII | R/W | Description | Default |
|---|---|---|---|---|
| 8~15 | 23[1] | RW | Auto_MDIX<br>1 : Enable auto crossover function<br>0 : Disable auto crossover function | 1 |
| 8~15 | 23[0] | RW | MDI/MDIX channel selection<br>1 : Select MDIX channel<br>0 : Select MDI channel | 0 |

The other Registers are reserved registers. User is inhibited to access to these registers. It may introduce abnormal function to write these registers.

### 6.1.7 10BaseT Control Register

**MII register 23 of PHY8~15 (Each PHY has its own MII register 23 with different PHY address)**

| Page | MII | R/W | Description | Default |
|---|---|---|---|---|
| **10BaseT Control Register** | | | | |
| 1 | 23[15:4] | RW | Reserved | 0x8000 |
| | 23[3] | RW | 10BaseT selection<br>1 : Select 10BaseT<br>0 : Select 10BaseTe | |
| | 23[2:0] | RW | Reserved | |

## 6.2 Switch Register

### 6.2.1 Switch Page Map

| Page | Description |
|---|---|
| 0x00 | MAC control registers |
| 0x01 | ARL registers 1 |
| 0x02 | ARL registers 2 |
| 0x03 | SMI registers |
| 0x04 | OAM control registers |
| 0x05 | Reserved |
| 0x06 | RxDMA registers |
| 0x07 | TxDMA registers |
| 0x08 | Otpt Qee registers |
| 0x09 | PTP registers |
| 0x0A | IRMP registers |
| 0x0B | Advanced EEPROM code registers |
| 0x0C | Misc/System registers |
| 0x0D | Reserved |

### 6.2.2 Switch Register Map

| Type | Description |
|---|---|
| R | Read |
| W | Write |
| R/W | Read/Write |
| SC | Self-Clearing |
| RO | Read Only |
| LL | Latching Low |
| LH | Latching High |

## 6.3 MAC Control Register

**Table 34 MAC Control Register**

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x00 | Switch ID/version register<br>bit[3:0] : IP1810I IC version<br>bit[15:4] : IP1810I ID | RO | 0x8290 |
| 0x01 | General MAC operation behavior register 1<br>bit[1:0] : Enable IPG compensation<br>00 : Disable<br>01 : compensation 160 ppm<br>10 : compensation 80 ppm<br>11 : compensation 40 ppm (default)<br>bit[2] : Back pressure method<br>0 : Carrier Sense base<br>1 : Collision base<br>bit[3] : Collision 16 times drop enable<br>bit[4] : Collision back off enable<br>bit[5] : Auto turn off flow control function if priority queue enable<br>bit[6] : In-band management<br>( Destination MAC address = switch MAC address)<br>0: Drop<br>1: to CPU port<br>(switch MAC address define in offset address x5C, 0x5D, 0x5E)<br>bit[7] : Disable TX packets CRC recalculation<br>bit[8] : Bypass RX packets CRC error<br>bit[9] : Enable for MIBs counters<br>bit[10] : Port 9 interface select<br>0 : Reserved<br>1 : RGMII<br>bit[13:11] : Reserved<br>bit[14] : Port 10 interface select<br>0 : MII<br>1 : RGMII (default) | R/W | 0x7C17 |
| 0x02 | Jumbo Packets enable for port 1 ~ port 8, 1 bit per port<br>bit[7:0] : enable jumbo<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x03 | Jumbo Packets enable for port 9 ~ port 10, 1 bit per port<br>bit[12][8] : enable jumbo<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x04 | MAC LoopBack enable (MDI TX to RX) for port 1 ~ port 8<br>bit[7:0] : LoopBack enable<br>0 : Disable 1 : Enable | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x05 | MAC LoopBack enable (RGMII/MII TX to RX) for port 9 ~ port 10<br>bit[12][8]: LoopBack enable<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x06 | Ethernet L2 protocol frames capture (if 0x33 bit[8] is 1'b1, the BPDU capture should follow the 0x30 ~ 0x33 setting)<br>bit[1:0] : for BPDU (01-80-C2-00-00-00)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[3:2] : for Slow Protocol (01-80-C2-00-00-02)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[5:4] : for 802.1X (01-80-C2-00-00-03)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[7:6] : for LLDP (01-80-C2-00-00-0E)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[9:8] : for Group 0 (01-80-C2-00-00-05, 06, 09 ~ 0C, 0F)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[11:10] : for All Bridge address (01-80-C2-00-00-10)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[13:12] : for Group 1 (01-80-C2-00-00-11 ~ 1F)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[15:14] : for GARP (01-80-C2-00-00-20, 21)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop | R/W | 0x030C |
| 0x07 | Ethernet L2 protocol frames capture<br>bit[1:0] : for Group 2 (01-80-C2-00-00-22 ~ 2F)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[3:2] : for Group 3 (01-80-C2-00-00-30 ~ 3F)<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[4] : CPU does not act for drop setting for reserved group address as defined above.<br>0 : enable drop setting<br>1 : Disable drop setting<br>bit[6:5] : Reserved<br>bit[7] : ARP<br>0: Forward depend on DA<br>1: To DA and CPU<br>bit[8] : PPPoE protocol check enable<br>0 : Don't check PPPoE type<br>1 : Check PPPoE type | R/W | 0x0040 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[9] : PPPoE header remove<br>0 : Don't remove 1 : Remove<br>bit[10] : IPv6 0 : Forward depend on DA 1 : To CPU port<br><br>bit[11] : Pass Pause frame with DA not belong to Switch Address and 01-80-C2-00-00-01<br>0 : Drop 1 : Pass<br>bit[12] : set (BPDU, 802.1x, Slow protocol, LLDP, GxRP) packets to highest priority if the corresponding protocol enable.<br>0 : Disable 1 : Enable<br>bit[13] : Capture CFM packet to CPU<br>0 : Disable 1 : Enable<br>bit[14] : Capture User define Ether Type (define in 0x23)<br>0 : Disable 1 : to DA and CPU<br>bit[15] : Check if OAM frame DA equals to 0x01_80_C2_00_00_02<br>0 : can be unicast<br>1 : DA must be 0x01_80_C2_00_00_02 | | |
| 0x08 | Ethernet L3 protocol packet capture<br>bit[1:0] : ICMP<br>bit[3:2] : TCP<br>bit[5:4] : UDP<br>bit[7:6] : OSPF<br>bit[9:8] : USR1 (0x09 low byte)<br>bit[11:10] : USR2 (0x09 high byte)<br>bit[13:12] : Ipv4 Other protocol,<br>00 : Send to ports only<br>01 : Send to CPU and ports<br>10 : Send to CPU only<br>11 : Drop | R/W | 0x0000 |
| 0x09 | User define IP protocol field<br>bit[7:0] : IP protocol USR1<br>bit[15:8] : IP protocol USR2 | R/W | 0x0000 |
| 0x0A | User define TCP/UDP port number A | R/W | 0x0000 |
| 0x0B | User define TCP/UDP port number B | R/W | 0x0000 |
| 0x0C | User define TCP/UDP port number C upper range setting | R/W | 0x0000 |
| 0x0D | User define TCP/UDP port number C lower range setting<br>IP1810I will check if the port number C is in the following range<br>lower range <= port number C <= upper range | R/W | 0x0000 |
| 0x0E | User define TCP/UDP port number D upper range setting | R/W | 0x0000 |
| 0x0F | User define TCP/UDP port number D lower range setting<br>IP1810I will check if the port number D is in the following range<br>lower range <= port number D <= upper range | R/W | 0x0000 |
| 0x10 | User define TCP/UDP port number E upper range setting | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x11 | User define TCP/UDP port number E lower range setting<br>IP1810I will check if the port number E is in the following range<br>lower range <= port number E <= upper range | R/W | 0x0000 |
| 0x12 | TCP/UDP port number based priority setting:<br>bit[3:0]: FTP (20,21)<br>bit[7:4]: SSH (22)<br>bit[11:8]: TELNET (23)<br>bit[15:12]: SMTP (25)<br>0000 to queue 0(Disable)<br>0001 ~ 0111: to queue 1 ~ queue 7<br>1001: to CPU only<br>1011 : to CPU and ports<br>1010: drop | R/W | 0x0000 |
| 0x13 | TCP/UDP port number based priority setting:<br>bit[3:0]: DNS (53)<br>bit[7:4]: BOOTP/DHCP (67, 68)<br>bit[11:8]: TFTP (69)<br>bit[15:12]: HTTP_0,1 (80)<br>0000 to queue 0(Disable)<br>0001 ~ 0111: to queue 1 ~ queue 7<br>1001: to CPU only<br>1011 : to CPU and ports<br>1010: drop | R/W | 0x0000 |
| 0x14 | TCP/UDP port number based priority setting:<br>bit[3:0]: POP3 (110)<br>bit[7:4]: NEWS (119)<br>bit[11:8]: SNTP 123)<br>bit[15:12]: NETBIOS0,1,2 (137 ~ 139)<br>0000 to queue 0(Disable)<br>0001 ~ 0111: to queue 1 ~ queue 7<br>1001: to CPU only<br>1011 : to CPU and ports<br>1010: drop | R/W | 0x0000 |
| 0x15 | TCP/UDP port number based priority setting:<br>bit[3:0]: IMAP_0,1 (143,220)<br>bit[7:4]: SNMP_0,1 (161,162)<br>bit[11:8]: HTTPS (443)<br>bit[15:12]: USR define A<br>0000 to queue 0(Disable)<br>0001 ~ 0111: to queue 1 ~ queue 7<br>1001: to CPU only<br>1011 : to CPU and ports<br>1010: drop | R/W | 0x0000 |
| 0x16 | TCP/UDP port number based priority setting:<br>bit[3:0]: USR define B<br>bit[7:4]: USR define C<br>bit[11:8]: USR define D<br>bit[15:12]: USR define E<br>0000 to queue 0(Disable)<br>0001 ~ 0111: to queue 1 ~ queue 7<br>1001: to CPU only<br>1011 : to CPU and ports<br>1010: drop | R/W | 0x0000 |
| 0x17 | TCP/UDP port number based filter to WAN port select:<br>bit[0]: FTP 0: Don't care 1: select<br>bit[1]: SSH 0: Don't care 1: select<br>bit[2]: TELNET 0: Don't care 1: select<br>bit[3]: SMTP 0: Don't care 1: select<br>bit[4]: DNS 0: Don't care 1: select<br>bit[5]: BOOP/DHCP 0: Don't care 1: select<br>bit[6]: TFTP 0: Don't care 1: select<br>bit[7]: HTTP_0,1 0: Don't care 1: select<br>bit[8]: POP3 0: Don't care 1: select<br>bit[9]: NEWS 0: Don't care 1: select<br>bit[10]: SNTP 0: Don't care 1: select<br>bit[11]: NETBIOS0,1,2 0: Don't care 1: select<br>bit[12]: IMAP_0,1 0: Don't care 1: select | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[13]: SNMP_0,1, 0: Don't care 1: select<br>bit[14]: HTTPS 0: Don't care 1: select<br>bit[15]: USR define A 0: Don't care 1: select<br>(corresponding to offset address 0x12 ~ 0x16) | | |
| 0x18 | TCP/UDP port number based filter to WAN port select:<br>bit[0]: USR define B 0: Don't care 1: select<br>bit[1]: USR define C 0: Don't care 1: select<br>bit[2]: USR define D 0: Don't care 1: select<br>bit[3]: USR define E 0: Don't care 1: select<br>(corresponding to offset address 0x12 ~ 0x16)<br><br>bit[4]: filter_md (0: negative list 1: positive list)<br>0 : negative list, the setting "1"s above which routed to wan will be filtered<br>1 : positive list, the setting "1"s above which routed to wan port can be passed, otherwise be filtered<br><br>( bit[4] : Detail ref to offset address 0x0D~0x10 (page6)) | R/W | 0x00 |
| 0x19 | TCP/UDP Port Number function enable for port 1 ~ port 8, 1 bit per port<br>bit[7:0] : function enable for Port 1 ~ Port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x1A | TCP/UDP Port Number function enable for port 9 ~ port 10, 1 bit per port<br>bit[12][8] : function enable for Port 9 ~ Port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x1B | TCP/UDP Port Number priority enable for port 1 ~ port 8,1 bit per port<br>bit[7:0] : priority enable for Port 1 ~ Port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x1C | TCP/UDP Port Number priority enable for port 9 ~ port 10, 1-bit /port<br>bit[12][8] : priority enable for Port 9 ~ Port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x1D | TCP/UDP Port Number check for "TCP protocol" enable<br>bit[15:0] : enable the TCP/UDP Port Number Function for "TCP protocol"<br>0 : Disable 1 : Enable<br>The Port Number Mapping is the same as define in 0x17 | R/W | 0xFFFF |
| 0x1E | TCP/UDP Port Number check for "TCP protocol" enable<br>bit[3:0] : enable the TCP/UDP Port Number Function for "TCP protocol"<br>0 : Disable 1 : Enable<br>The Port Number Mapping is the same as define in 0x18 | R/W | 0xF |
| 0x1F | TCP/UDP Port Number check for "UDP protocol" enable<br>bit[15:0] : enable the TCP/UDP Port Number Function for "UDP protocol"<br>0 : Disable 1 : Enable<br>The Port Number Mapping is the same as define in 0x17 | R/W | 0xFFFF |
| 0x20 | TCP/UDP Port Number check for "UDP protocol" enable<br>bit[3:0] : enable the TCP/UDP Port Number Function for "UDP protocol"<br>0 : Disable 1 : Enable<br>The Port Number Mapping is the same as define in 0x18 | R/W | 0xF |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x21 | In-band management restriction configuration 1<br>When the register 0x01 bit[6] and 0x22 bit[15] are both 1'b1, the in-band Management packets will forward as the register 0x21 and 0x22 settings.<br>bit[0] : ARP<br>bit[1] : IPv4<br>bit[2] : IPv6<br>bit[3] : PPPoE<br>bit[4] : ICMP<br>bit[5] : TCP<br>bit[6] : UDP<br>bit[7] : User define Ether Type<br>bit[8] : User define IP Protocol 1<br>bit[9] : User define IP Protocol 2<br>bit[10] : ICMPv6<br>bit[11] : FTP<br>bit[12] : SSH<br>bit[13] : Telnet<br>bit[14] : SMTP<br>bit[15] : DNS<br>0 : drop 1 : forward to CPU | R/W | 0x0000 |
| 0x22 | In-band management restriction configuration 2<br>bit[0] : BOOTP/DHCP<br>bit[1] : TFTP<br>bit[2] : HTTP<br>bit[3] : POP3<br>bit[4] : NEWS<br>bit[5] : SNTP<br>bit[6] : NETBIOS<br>bit[7] : IMAP<br>bit[8] : SNMP<br>bit[9] : HTTPS<br>bit[10] : user define TCP/UDP Port Number A<br>bit[11] : user define TCP/UDP Port Number B<br>bit[12] : user define TCP/UDP Port Number C<br>bit[13] : user define TCP/UDP Port Number D<br>bit[14] : user define TCP/UDP Port Number E<br>0 : drop 1 : forward to CPU<br>bit[15] : in-band management restriction enable<br>(0x01 bit[6] need to be 1'b1)<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x23 | User define Ether Type<br>bit[15:0] : user define Ether Type Value | R/W | 0x0000 |
| 0x24 ~ 0x2F | Reserved | | |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x30 | BPDU per Port Capture configuration for port 1 ~ port 8<br>bit[1:0] : Port 1 BPDU setting<br>bit[3:2] : Port 2 BPDU setting<br>bit[5:4] : Port 3 BPDU setting<br>bit[7:6] : Port 4 BPDU setting<br>bit[9:8] : Port 5 BPDU setting<br>bit[11:10] : Port 6 BPDU setting<br>bit[13:12] : Port 7 BPDU setting<br>bit[15:14] : Port 8 BPDU setting<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop | R/W | 0x0000 |
| 0x31<br>0x32 | Reserved | | |
| 0x33 | BPDU per Port Capture configuration for port 9<br>bit[5:4] : BPDU setting for Port 9<br>bit[8] : BPDU per Port setting enable | R/W | 0x000 |
| 0x34<br>~<br>0x39 | Reserved | | |
| 0x3A | Port based priority enable for port 1 ~ port5, 3 bit per port<br>bit[2:0] : port based priority setting for port 1<br>000 – to queue 0 (lowest priority)<br>001 – to queue 1<br>…<br>111 – to queue 7 (highest priority)<br>bit[5:3] : port 2<br>bit[8:6] : port 3<br>bit[11:9] : port 4<br>bit[14:12] : port 5 | R/W | 0x0000 |
| 0x3B | Port based priority enable for port 6 ~ port 8, 3 bit per port, bit[8:0] | R/W | 0x0000 |
| 0x3C<br>0x3D | Reserved | | |
| 0x3E | Port based priority enable for port 9, 3 bit per port, bit[14:12] | R/W | 0x0000 |
| 0x3F | Port based priority enable for port 10, 3 bit per port, bit[11:9] | R/W | 0x0000 |
| 0x40 | Tag based priority settings<br>bit[7:0] : enable 802.1Q tag priority for port 1 ~ port 8, 1 bit per port | R/W | 0x0000 |
| 0x41 | Tag based priority settings<br>bit[12][8] : enable 802.1Q tag priority for port 9 ~ port 10, 1 bit per port<br>bit[13] : enable special tag priority for CPU Port (port 10)<br>bit[14] : 802.1Q tag priority Edition select<br>0 : 2005 Edition 1 : earlier Edition<br>bit[15] : exchange the priority of 3'b000 and 3'b001 for 2005 Edition | R/W | 0x0000 |
| 0x42 | IP CoS based priority enable<br>bit[7:0]: IP CoS based priority enable for port 1 ~ port 8, 1 bit per port | R/W | 0x0000 |
| 0x43 | IP CoS based priority enable<br>bit[12][8]: IP CoS based priority enable for port 9 ~ port 10. 1 bit per port | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x44 | Differentiated Services (DS) priority setting<br>bit[2:0]: priority setting for DSCP 0<br>bit[5:3]: priority setting for DSCP 1<br>bit[8:6]: priority setting for DSCP 2<br>bit[11:9]: priority setting for DSCP 3<br>bit[14:12]: priority setting for DSCP 4 | R/W | 0x7FFF |
| 0x45 | Differentiated Services (DS) priority setting<br>bit[2:0]: priority setting for DSCP 5<br>bit[5:3]: priority setting for DSCP 6<br>bit[8:6]: priority setting for DSCP 7<br>bit[9] : priority for DSCP not match<br>0 : regard as low priority (priority 0)<br>1 : ignore IP priority (priority will according to tag/port) | R/W | 0x3FF |
| 0x46 | DSCP configuration for DSCP_0 and DSCP_1<br>bit[5:0]: DSCP 0 configuration<br>bit[11:6]: DSCP 1 configuration | R/W | 0x280 |
| 0x47 | DSCP configuration for DSCP_2 and DSCP_3<br>bit[5:0]: DSCP 2 configuration<br>bit[11:6]: DSCP 3 configuration | R/W | 0x692 |
| 0x48 | DSCP configuration for DSCP_4 and DSCP_5<br>bit[5:0]: DSCP 4 configuration<br>bit[11:6]: DSCP 5 configuration | R/W | 0xBA2 |
| 0x49 | DSCP configuration for DSCP_6 and DSCP_7<br>bit[5:0]: DSCP 6 configuration<br>bit[11:6]: DSCP 7 configuration | R/W | 0xE30 |
| 0x4A | Source MAC address based priority setting<br>bit[7:0] : SMAC priority enable for port 1~ port 8 | R/W | 0x0000 |
| 0x4B | Source MAC address based priority setting<br>bit[12][8] : SMAC priority enable for port 9 ~ port 10 | R/W | 0x0000 |
| 0x4C | VID based priority enable setting<br>bit[7:0] : VID priority enable for port 1 ~ port 8 | R/W | 0x0000 |
| 0x4D | VID based priority enable setting<br>bit[12][8]: VID priority enable for port 9 ~ port 10 | R/W | 0x0000 |
| 0x4E | IP based priority setting<br>bit[7:0] : IP priority enable for port 1 ~ port 8<br>(ref Page 0x02, reg 0x01, 0x02) | R/W | 0x0000 |
| 0x4F | IP based priority setting<br>bit[12][8] : IP priority enable for port 9 ~ port 10<br>(ref Page 0x02, reg 0x01, 0x02) | R/W | 0x0000 |
| 0x50 | IGMP Multicast Group based priority enable setting<br>bit[7:0] : IGMP Multicast Group priority enable for port 1 ~ port 8 | R/W | 0x0000 |
| 0x51 | IGMP Multicast Group based priority enable setting<br>bit[12][8] : IGMP Multicast Group priority enable for port 9 ~ port 10 | R/W | 0x0000 |
| 0x52 | ACL based priority enable setting<br>bit[7:0] : ACL priority enable for port 1 ~ port 8 | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x53 | ACL based priority enable setting<br>bit[12][8] : ACL priority enable for port 9 ~ port 10 | R/W | 0x0000 |
| 0x54 | 802.1x Port Lock Enable (port 1 ~ port 8)<br>bit[7:0]: 0 : normal operation<br>1 : drop all frames except ARP & 802.1x EAPOL packet | R/W | 0x0000 |
| 0x55 | 802.1x Port Lock Enable (port 9 ~ port 10)<br>bit[12][8] 0 : normal operation<br>1 : drop all frames except ARP & 802.1x EAPOL packet | R/W | 0x0000 |
| 0x56 | PPPoE session ID A | R/W | 0x0000 |
| 0x57 | PPPoE session ID B | R/W | 0x0000 |
| 0x58 | PPPoE session ID C | R/W | 0x0000 |
| 0x59 | PPPoE session ID D | R/W | 0x0000 |
| 0x5A | MAC reset for port 1 ~ port 8, 1 bit per port<br>bit[7:0] : 0: reset 1: normal | R/W | 0xFFFF |
| 0x5B | MAC reset for port 9 ~ port 10, 1 bit per port<br>bit[12][8] : 0: reset 1: normal | R/W | 0x1FFF |
| 0x5C | Switch's MAC address[15:0] | R/W | 0x0003 |
| 0x5D | Switch's MAC address[31:16] | R/W | 0xC300 |
| 0x5E | Switch's MAC address[47:32] | R/W | 0x0090 |
| 0x5F | Configuration for Low Power Idle<br>bit[4:0] : threshold of IDLE period for entering LPI<br>bit[7:5] : the interval selection for IDLE counter<br>0 ~ 7 – 1us/10us/100us/ 1ms/ 10ms/ 100ms/ 1s/ 10s<br>bit[8] : Switch enter PSS when IDLE period hit the threshold<br>0 : Disable 1 : Enable<br>bit[9] : use default Tw as the resolved Tw<br>0 : Disable 1 : Enable<br>bit[10] : use LinkPartner's Tw as the resolved Tw (note : bit[9] has higher priority than this bit )<br>0 : Disable 1 : Enable<br>bit[11] : Reserved<br>bit[12] : select the large one between default Tw and LinkPartner's Tw as the resolved Tw<br>0 : Disable 1 : Enable<br>bit[13] : force to enable PSS support<br>0 : Disable 1 : Enable<br>bit[14] : Reserved. Should be 1'b0.<br>bit[15] : Reserved. Should be 1'b1. | R/W | 0x820F |
| 0x60 | Configuration of the LPI State for port 1 ~ port 8, per port 1 bit<br>bit[7:0] : LPI state for each port<br>0 : normal state 1 : in LPI State | R/W | 0x0000 |
| 0x61 | Configuration of the LPI State for port 9 ~ port 10, per port 1 bit<br>bit[12][8]: LPI state for each port<br>0 : normal state 1 : in LPI State | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x62 | Configuration for LPI default Tw for 10/100 Mbps<br>bit[15:0] : default Tw setting in us | R/W | 0x0023 |
| 0x63 | Configuration LPI default Tw for giga speed<br>bit[15:0] : default Tw setting in us | R/W | 0x0011 |
| 0x64 | CPU Read statistic counter command<br>bit[4:0]: statistic counter read address ( R : ~0x17, T : ~0x13)<br>bit[9:5]: statistic counter Port select ( port 1 ~ port 10 : 0~7, 24, 28)<br>bit[10] : statistic counter RX/TX selection<br>0 : RX 1 : TX<br>bit[11] : statistic counter sequential read enable<br>0 : Disable 1 : Enable<br>bit[12] : Reserved<br>bit[14] : statistic counter read clear enable<br>0 : Disable 1 : Enable<br>bit[15] : statistic counter command trig/Ack | R/W | 0x0000 |
| 0x65 | CPU Read low 16 bits statistic counter data | R/W | 0x0000 |
| 0x66 | CPU Read high 16 bits statistic counter data | R/W | 0x0000 |
| 0x67 | MAC self test packets control setting<br>bit[7:0] : send self test packet for port 1 ~ port 8, per port 1 bit<br>0 : do not send 1 : send | R/W | 0x0000 |
| 0x68 | MAC self test packets control setting<br>bit[12][8] : send self test packet for port 9 ~ port 10, per port 1 bit<br>0 : do not send 1 : send<br>bit[14:13] : setting for self test packets number<br>00 : 32768 01 : 4096<br>10 : 256 11 : 16 | R/W | 0x0000 |
| 0x69 | MAC self test result<br>bit[7:0] : self test result for port 1 ~ port 8, per port 1 bit<br>0 : fail 1 : pass | RO | 0x0000 |
| 0x6A | MAC self test result<br>bit[12][8] : self test result for port 9 ~ port 10, per port 1 bit<br>0 : fail 1 : pass | RO | 0x0000 |
| 0x6B | LoopDetect config register<br>bit[7:0] : loop detect enable for port 1 ~ port 8, per port 1 bit<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x6C | LoopDetect config register<br>bit[12][8] : loop detect enable for port 9 ~ port 10, per port 1 bit<br>0 : Disable 1 : Enable<br>bit[13] : loop detect time unit select<br>0 : 500ms 1 : 10s<br>bit[14] : setting LoopDetect SA[40] value<br>bit[15] : setting for LoopDetect re-random<br>0 : Disable 1 : Enable<br>PS. To block loop port, Page 0x01, register 0x24 bit[3] need to be 1'b1 | R/W | 0xC000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x6D | LoopDetect timer<br>bit[7:0] : LPD packet sending timer, the time interval is timer x time unit<br>bit[15:8] : Block release timer, the time interval is timer x time unit | R/W | 0x0903 |
| 0x6E | LoopDetect frame DA[15:0] setting<br>bit[15:0] : setting for LPD frame DA[15:0] | R/W | 0x0000 |
| 0x6F | LoopDetect frame DA[31:16] setting<br>bit[15:0] : setting for LPD frame DA[31:16] | R/W | 0xC300 |
| 0x70 | LoopDetect frame DA[47:32] setting<br>bit[15:0] : setting for LPD frame DA[47:32] | R/W | 0x0190 |
| 0x71 | LoopDetect frame EtherType setting<br>bit[15:0] : setting for LPD frame EtherType | R/W | 0x8931 |
| 0x72 | LoopDetect frame sub type setting<br>bit[15:0] : setting for LPD frame sub type | R/W | 0xFFFF |
| 0x73 | LoopDetect frame Device ID setting<br>bit[7:0] : setting for LPD frame device ID | R/W | 0x05 |
| 0x74 | TCP flag global configuration<br>bit[0] : Drop null flag<br>0 : Disable 1 : Enable<br>bit[1] : Drop ALL-Set flag<br>0 : Disable 1 : Enable<br>bit[3:2] : Period selection for Flag storm control counter<br>00 : 1000 / 100 / 10 Mbps -> 200us / 2ms / 20ms<br>01 : 1000 / 100 / 10 Mbps -> 1ms / 10ms / 100ms<br>10 : 1000 / 100 / 10 Mbps -> 10ms / 10ms / 10ms<br>11 : 1000 / 100 / 10 Mbps -> 100ms / 100ms / 100ms | R/W | 0x0 |
| 0x75 | TCP flag 0 configuration<br>bit[7:0] : The TCP flag 0 configuration<br>bit[10:8] : Priority setting for TCP flag 0<br>bit[11] : Priority setting enable for TCP flag 0<br>0 : Disable 1 Enable<br>bit[13:12] : action for TCP flag 0<br>00 : none 01 : storm control<br>10 : to CPU only 11 : drop<br>bit[14] : reserved, should be 1'b0 | R/W | 0x0000 |
| 0x76 | TCP flag 1 configuration<br>bit[7:0] : The TCP flag 1 configuration<br>bit[10:8] : Priority setting for TCP flag 1<br>bit[11] : Priority setting enable for TCP flag 1<br>0 : Disable 1 Enable<br>bit[13:12] : action for TCP flag 1<br>00 : none 01 : storm control<br>10 : to CPU only 11 : drop<br>bit[14] : reserved, should be 1'b0 | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x77 | TCP flag 2 configuration<br>bit[7:0] : The TCP flag 2 configuration<br>bit[10:8] : Priority setting for TCP flag 2<br>bit[11] : Priority setting enable for TCP flag 2<br>0 : Disable 1 Enable<br>bit[13:12] : action for TCP flag 2<br>00 : none 01 : storm control<br>10 : to CPU only 11 : drop<br>bit[14] : reserved, should be 1'b0 | R/W | 0x0000 |
| 0x78 | TCP flag 3 configuration<br>bit[7:0] : The TCP flag 3 configuration<br>bit[10:8] : Priority setting for TCP flag 3<br>bit[11] : Priority setting enable for TCP flag 3<br>0 : Disable 1 Enable<br>bit[13:12] : action for TCP flag 3<br>00 : none 01 : storm control<br>10 : to CPU only 11 : drop<br>bit[14] : reserved, should be 1'b0 | R/W | 0x0000 |
| 0x79 | TCP flag0 and flag1 storm control threshold setting<br>bit[7:0] : threshold setting for TCP flag 0<br>bit[15:8] : threshold setting for TCP flag 1 | R/W | 0x0000 |
| 0x7A | TCP flag2 and flag3 storm control threshold setting<br>bit[7:0] : threshold setting for TCP flag 2<br>bit[15:8] : threshold setting for TCP flag 3 | R/W | 0x0000 |
| 0x7B | TCP flag 0 related function enable for port 1 ~ port 8<br>bit[7:0] : enable for TCP flag 0 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x7C | TCP flag 0 related function enable for port 9 ~ port 10<br>bit[12][8] : enable for TCP flag 0 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x7D | TCP flag 1 related function enable for port 1 ~ port 8<br>bit[7:0] : enable for TCP flag 1 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x7E | TCP flag 1 related function enable for port 9 ~ port 10<br>bit[12][8]: enable for TCP flag 1 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x7F | TCP flag 2 related function enable for port 1 ~ port 8<br>bit[7:0] : enable for TCP flag 2 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x80 | TCP flag 2 related function enable for port 9 ~ port 10<br>bit[12][8] : enable for TCP flag 2 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x81 | TCP flag 3 related function enable for port 1 ~ port 8<br>bit[7:0] : enable for TCP flag 3 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x82 | TCP flag 3 related function enable for port 9 ~ port 10<br>bit[12][8]: enable for TCP flag 3 related function<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x83 | EoC configuration 1<br>bit[7:0] : EoC function enable for port 1 ~ port 8 | R/W | 0x0000 |
| 0x84 | EoC configuration 2<br>bit[12][8] : EoC function enable for port 9 ~ port 10<br>bit[13] : Clear EoC block when receive a good packet<br>bit[14]: EoC release time selection<br>0 : 1 Minute 1 : 10 Minutes<br>bit[15] : Force to clear all EoC block (this bit will be self clear)<br>PS. To block loop port, Page 0x01, register 0x24 bit[3] need to be 1'b1 | R/W | 0x0000 |
| 0x85 | EoC block status for port 1 ~ port 8<br>bit[7:0] : block status check for port 1 ~ port 8<br>0 : normal 1 : loop detected | RO | 0x0000 |
| 0x86 | EoC block status for port 9 ~ port 10<br>bit[12][8]: block status check for port 9 ~ port 10<br>0 : normal 1 : loop detected | RO | 0x0000 |
| 0x87 | Ingress Rate control for port 1<br>bit[13:0] : Input Rate control<br>The bandwidth = Rate control configure value x 64k bps | R/W | 0x0000 |
| 0x88 ~ 0x8E | Ingress Rate control for port 2 ~ port 8<br>bit[13:0] : Input Rate control | R/W | 0x0000 |
| 0x8F ~ 0x9E | Reserved | | |
| 0x9F | Ingress Rate control for port 9<br>bit[13:0] : Input Rate control | R/W | 0x0000 |
| 0xA0 ~ 0xA2 | Reserved | | |
| 0xA3 | Ingress Rate control for port 10<br>bit[13:0] : Input Rate control | R/W | 0x0000 |
| 0xA4 | Configuration for PTP<br>bit[0] : PTP protocol enable (packets will to CPU only)<br>bit[1] : PTP config for DA = 01-1B-19-00-00-00<br>0 : Enable 1 : not regard as PTP<br>bit[2] : PTP config for DA = 01-80-C2-00-00-0E<br>0 : Enable 1 : not regard as PTP<br>bit[3] : PTP config for UDP Destination Port<br>0 : Enable 1 : not check for PTP<br>bit[4] : PTP config for UDP Source Port<br>0 : Enable 1 : not check as PTP | R/W | 0x000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[5] : Disable PTP packet send to CPU only<br>0 : to CPU only 1 : forward as DA specific<br>bit[6] : Also check NTP packet<br>0 : Disable 1 : Enable<br>bit[7] : PTP packets regard as high priority<br>0 : Disable 1 : Enable<br>bit[8] : Allow PTP packets with unicast DA<br>0 : Disable 1 : Enable<br>bit[9] : Check if Destination IP equal to 224.0.1.129 or 224.0.0.107<br>(if enable, the UDP port number must also match)<br>0 : Disable 1 : Enable | | |
| 0xA5 | LoopDetect port 1 ~ port 8 status<br>bit[7:0] : port status for port 1 ~ port 8<br>0 : normal 1 : loop detected | R | 0x0000 |
| 0xA6 | LoopDetect port 9 ~ port 10 status<br>bit[12][8] : port status for port 9 ~ port 10<br>0 : normal 1 : loop detected | R | 0x0000 |
| 0xA7 | IPv6 related functions configuration<br>bit[0] : Enable TCP/UDP Port Number related functions for IPv6<br>0 : Disable 1 : Enable<br>bit[1] : Enable TCP/UDP flag related functions for IPv6 (bit[0] need 1'b1)<br>0 : Disable 1 : Enable<br>bit[2] : Reserved, should be 1'b0.<br>bit[3] : stop finding next header at Auth header<br>bit[4] : stop finding next header at Encapsulation header<br>bit[5] : stop finding next header at Fragment header<br>bit[6] : stop finding next header at user define next header 1<br>bit[7] : stop finding next header at user define next header 2<br>0 : Enable 1 : Disable<br>bit[8] : forwarding action "Send to CPU and ports" has higher priority than "drop"<br>0 : Disable 1 : Enable<br>bit[9] : forwarding action "Send to CPU only" has higher priority than "drop"<br>0 : Disable 1 : Enable | R/W | 0x003 |
| 0xA8 | IPv6 related header forwarding<br>bit[1:0] : Fragment header<br>bit[3:2] : Encapsulation header<br>bit[5:4] : Authentication header<br>bit[7:6] : ICMPv6 header<br>bit[9:8] : ICMPv6 MLD<br>bit[11:10] : ICMPv6 NDP<br>bit[13:12] : user define next header 1<br>bit[15:14] : user define next header 2<br>00 : Send to ports only<br>01 : Send to CPU and ports<br>10 : Send to CPU only<br>11 : Drop | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xA9 | IPv6 related header forwarding<br>bit[1:0] : ICMPv6 user define type 1<br>bit[3:2] : ICMPv6 user define type 2<br>00 : Send to ports only<br>01 : Send to CPU and ports<br>10 : Send to CPU only<br>11 : Drop | R/W | 0x0 |
| 0xAA | IPv6 user define next header configuration<br>bit[7:0] : user define next header 1<br>bit[15:8] : user define next header 2 | R/W | 0x0000 |
| 0xAB | ICMPv6 user define type 1 setting<br>bit[7:0] : lower range setting<br>bit[15:8] : upper range setting | R/W | 0x0000 |
| 0xAC | ICMPv6 user define type 2 setting<br>bit[7:0] : lower range setting<br>bit[15:8] : upper range setting | R/W | 0x0000 |
| 0xAD | WOL+ Enable for TX enter saving mode<br>bit[7:0] : WOL+ Enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0xAE | WOL+ enable for TX enter saving mode<br>bit[12][8]: WOL+ enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0xAF | WOL+ configuration<br>bit[4:0] : threshold of idle period for entering WOL+ saving mode<br>(if giga port is enable for this function, the threshold should be greater than 30s)<br>bit[6:5] : the interval selection for WOL+ idle counter<br>00 : Disable 01 : 10 s<br>10 : 1min 11 : 10 min<br>bit[7] : if TX any packet, leave WOL+ mode or not<br>0 : leave 1 : Don't care<br>bit[8] : if RX any packet, leave WOL+ mode or not<br>0 : leave 1 : Don't care<br>bit[9] : WOL+ mode select<br>0 : slave 1 : master<br>bit[10] : leave WOL+ mode if match ACL action : also to CPU<br>0 : Disable 1 : Enable<br>bit[11] : Enable Interrupt to CPU<br>0 : Disable 1 : Enable | R/W | 0x000 |
| 0xB0 | WOL+ mode control for port 1 ~ port 8<br>bit[7:0] : WOL+ active for port 1 ~ port 8<br>0 : normal 1 : enter WOL+ mode | R/W | 0x0000 |
| 0xB1 | WOL+ mode control for port 9 ~ port 10<br>bit[12][8] : WOL+ active for port 9 ~ port 10<br>0 : normal 1 : enter WOL+ mode | R/W | 0x0000 |

| MAC Control Registers : Page 0x00 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xB2 | WOL+ event for port 1 ~ port 8<br>bit[7:0] : Indicate the port had entered WOL+ mode, read clear<br>0 : normal 1 : had entered WOL+ mode | RO | 0x0000 |
| 0xB3 | WOL+ event for port 9 ~ port 10<br>bit[12][8] : Indicate the port had entered WOL+ mode, read clear<br>0 : normal 1 : had entered WOL+ mode | RO | 0x0000 |
| 0xB4 | Reserved | | |
| 0xB5 | Ethernet L2 protocol frames capture<br>bit[1:0] : for 01-80-C2-00-00-04<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[3:2] : for 01-80-C2-00-00-07<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[5:4] : for 01-80-C2-00-00-08<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop<br>bit[7:6] : for 01-80-C2-00-00-0D<br>00 : Forward 01 : Reserved<br>10 : to CPU 11 : Drop | R/W | 0xFF |
| 0xB6 | Reserved | | |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x0000 |

## 6.4 ARL Control Register Page 1

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | LUT Aging timer setting<br>bit[14:0] : Aging timer, the aging time is about :<br>(mg_aging_time + 1) x 55.3 sec. ±3.8%<br>bit[15] : Aging timer disable<br>0 : Enable 1 : Disable | R/W | 0x0005 |
| 0x02 | Source MAC address learning configure<br>bit[1:0] : LUT learning mode<br>00 : overwrite L1/L2 according to the aging time<br>01, 10 : never overwrite before aging out<br>11 : only L2 will overwrite before aging out<br>bit[2] : learning for null SA.<br>0 : Don't learn 1: learn.<br>bit[3] : Hashing algorithm selection<br>0 : Direct hashing<br>1 : CRC hashing<br>bit[4] : Disable forward unknown SA frame to CPU port<br>0 : forward 1 : not forward<br>bit[5] : forward unknown SA frame if source address learning function is disable<br>0 : drop the unknown SA frame<br>1 : forward the unknown SA frame<br>bit[6] : indicates the SA of input frame should match the original learn port or not<br>0 : SA will not bind the port<br>1 : SA will bind to the original learned port<br>bit[7] : Learning SA or not when Packet drop by VLAN ingress check<br>0 : Learning 1 : not Learning | R/W | 0x2C |
| 0x03 | Source MAC address learning enable for port 1 ~ port 8<br>bit[7:0] : Learning enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0xFFFF |
| 0x04 | Source MAC address learning Enable for port 9 ~ port 10<br>bit[12][8]: Learning Enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x1FFF |
| 0x05 | Source MAC address learning count control enable for port 1 ~ port 8<br>bit[7:0] : learning count control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x06 | Source MAC address learning count control enable for port 9 ~ port 10<br>bit[12][8]: learning count control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x07 | Source MAC address learning count control threshold setting<br>bit[8:0] : learning count control threshold | R/W | 0x000 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x08 ~ 0x0C | Reserved | | |
| 0x0D | Broadcast storm control configuration<br>bit[7:0] : Broadcast storm control threshold<br>bit[9:8] : Broadcast storm counter clear period selection<br>00 : 200 us / 2ms / 20ms for (Giga/100/10)<br>01 : 1 ms / 10ms / 100ms for (Giga/100/10)<br>10 : 10 ms / 10ms / 10ms for (Giga/100/10)<br>11 : 100 ms / 100ms / 100ms for (Giga/100/10)<br>bit[10] : Block Broadcast/Multicast to CPU Enable<br>0 : Disable 1 : Enable<br>bit[11] : Don't block bcst/Mcst Ipv4/IPv6 packets when bit[10] is enable<br>0 : block 1 : Don't block<br>bit[12] : ignore 01:00:5E:xx:xx:xx for Mcst storm<br>0: Don't ignore 1 : Ignore | R/W | 0x01FF |
| 0x0E | Broadcast storm control enable for port 1 ~ port 8<br>bit[7:0] : Broadcast storm control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x0F | Broadcast storm control enable for port 9 ~ port 10<br>bit[12][8]: Broadcast storm control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x10 | Multicast storm control enable for port 1 ~ port 8<br>(share the same threshold with broadcast storm control)<br>bit[7:0] : Multicast storm control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x11 | Multicast storm control enable for port 9 ~ port 10<br>(share the same threshold with broadcast storm control)<br>bit[12][8] : Multicast storm control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x12 | DLF storm control enable for port 1 ~ port 8<br>(share the same threshold with broadcast storm control)<br>bit[7:0] : DLF storm control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x13 | DLF storm control enable for port 9 ~ port 10<br>(share the same threshold with broadcast storm control)<br>bit[12][8]: DLF storm control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x14 | ARP storm control configuration<br>bit[7:0] : ARP storm control threshold<br>bit[9:8] : ARP storm counter clear period selection<br>00 : 200 us / 2ms / 20ms for (Giga/100/10)<br>01 : 1 ms / 10ms / 100ms for (Giga/100/10)<br>10 : 10 ms / 10ms / 10ms for (Giga/100/10)<br>11 : 100 ms / 100ms / 100ms for (Giga/100/10) | R/W | 0x000 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[10] : Blocked ARP frame forward to CPU<br>0 : Disable 1 : Enable<br>bit[11] : ARP storm drop interrupt enable | | |
| 0x15 | ARP storm control enable for port 1 ~ port 8<br>bit[7:0] : ARP storm control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x16 | ARP storm control enable for port 9 ~ port 10<br>bit[12][8] : ARP storm control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x17 | ICMP storm control configuration<br>bit[7:0] : ICMP storm control threshold<br>bit[9:8] : ICMP storm counter clear period selection<br>00 : 200 us / 2ms / 20ms for (Giga/100/10)<br>01 : 1 ms / 10ms / 100ms for (Giga/100/10)<br>10 : 10 ms / 10ms / 10ms for (Giga/100/10)<br>11 : 100 ms / 100ms / 100ms for (Giga/100/10)<br>bit[10] : ICMP storm drop interrupt enable | R/W | 0x000 |
| 0x18 | ICMP storm control enable for port 1 ~ port 8<br>bit[7:0] : ICMP storm control enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x19 | ICMP storm control enable for port 9 ~ port 10<br>bit[12][8] : ICMP storm control enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x1A | Port based address flush configuration<br>bit[7:0] : indicates that these addresses learnt by selected port should be flushed for port 1 ~ port 8<br>0 : Don't flush 1 : flush | R/W | 0x0000 |
| 0x1B | Port based address flush configuration<br>bit[12][8] : indicates that these addresses learnt by selected port should be flushed for port 9 ~ port 10<br>0 : Don't flush 1 : flush<br>bit[13] : flush command trig | R/W | 0x0000 |
| 0x1C | Trunk operation / Group configuration<br>bit[2:0] : Trunk load balance hashing method selection<br>000 : Port ID 100 : DIP<br>001 : SA 101 : SIP<br>010 : DA 110 : TCP/UDP DP<br>011 : DA/SA 111 : TCP/UDP SP<br>bit[3] : Trunk hashing method sequence enable<br>0 : Disable<br>1 : Enable<br>bit[4] : CPU route function Don't care Trunk and VLAN result<br>bit[5] : combine Trunk Group A and Group B into one Trunk<br>bit[7:6] : Reserved<br>bit[11:8] : Trunk Group A configuration (port 1 ~ port 4)<br>0 : not in trunk 1 : in trunk | R/W | 0x0000 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[15:12] : Trunk Group B configuration (port 5 ~ port 8)<br>0 : not in trunk 1 : in trunk | | |
| 0x1D | Reserved | | |
| 0x1E | First Sniffer destination port group configuration<br>bit[7:0] : select destination configuration for port 1 ~ port 8<br>0 : none 1 : selected | R/W | 0x0000 |
| 0x1F | First Sniffer destination port group configuration<br>bit[12][8] : select destination configuration for port 9 ~ port 10<br>0 : none 1 : selected<br>bit[14:13] : sniffer method<br>00: Disable 01 : egress<br>10: ingress 11 : ingress+egress<br>bit[15] : Tag/Untag setting for sniffered packet<br>0 : according to other criteria 1 : keep the original packet | R/W | 0x0000 |
| 0x20 | Second Sniffer destination port group configuration<br>(only work for LUT/ACL/IGMP)<br>bit[7:0] : select destination configuration for port 1 ~ port 8<br>0 : none 1 : selected | R/W | 0x0000 |
| 0x21 | Second Sniffer destination port group configuration<br>(only work for LUT/ACL/IGMP)<br>bit[12][8]: select destination configuration for port 9 ~ port 10<br>0 : none 1 : selected<br>bit[13] : LUT trigger target for sniffer destination port group 1<br>0 : DA 1 : SA<br>bit[14] : LUT trigger target for sniffer destination port group 2<br>0 : DA 1 : SA<br>bit[15] : check if the destination port linkup or not<br>0 : check 1 : don't check | R/W | 0x0000 |
| 0x22 | Sniffer source port configuration<br>bit[7:0] : sniffer source config for port 1 ~ port 8<br>0 : no selected 1 : source | R/W | 0x0000 |
| 0x23 | Sniffer source port configuration<br>bit[12][8] : sniffer source config for port 9 ~ port 10<br>0 : no selected 1 : source | R/W | 0x0000 |
| 0x24 | MISC configuration<br>bit[0] : ACL function enable<br>0 : Disable 1 : Enable<br>bit[1] : ACL ether type location for tag frame<br>0 : the location is after SA<br>1 : the location is after tag (ctag and/or stag)<br>bit[2] : ACL ether type location for RFC 1042 frame<br>0 : the location is after SA<br>1 : the location is after RFC 1042 header<br>bit[3] : LoopDetect block enable<br>0 : Disable 1 : Enable | R/W | 0x00 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[5] : Sniffer Add/Rem Tag option for CPU special tag routing<br>(reg 0x1C, bit[4] must be 1'b1)<br>0 : keep the original data 1 : according to other criteria<br>bit[6] : Sniffer Add/Rem Tag option for ACL redirect/to_cpu action<br>0 : keep the original data 1 : according to other criteria<br>bit[7] : Sniffer Add/Rem Tag option for other packets<br>(include CPU special tag counting with Reg 0x1C, bit[4] is 1'b0)<br>0 : keep the original data 1 : according to other criteria<br>Note : bit[7:5] only work when 0x1F bit[15] is 1'b1 | | |
| 0x25 | 802.3 OAM LoopBack configuration<br>bit[7:0] : LoopBack enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x26 | 802.3 OAM LoopBack configuration<br>bit[12][8]: LoopBack enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x27 | 802.3 OAM LoopBack configuration<br>bit[7:0] : LoopBack mode select for port 1 ~ port 8<br>0 : passive mode 1 : active mode | R/W | 0x0000 |
| 0x28 | 802.3 OAM LoopBack configuration<br>bit[12][8] : LoopBack mode select for port 9 ~ port 10<br>0 : passive mode 1 : active mode | R/W | 0x0000 |
| 0x29 | IGMP snooping function configuration<br>bit[0] : enable IGMP snooping function<br>bit[1] : multicast table make by CPU<br>bit[2] : Disable IP address 224.0.0.X broadcast<br>(except IGMP packet )<br>bit[3] : Disable **"Leave"** function<br>( work in hardware-based IGMP-snooping)<br>bit[4] : Reserved<br>bit[5] : enable IGMP group member aging<br>0 : Disable 1 : Enable<br>bit[12:6] : group member aging time setting<br>(mg_mcst_thr + 1) x 70.4 sec. ±9.1%<br>bit[13] : router list make by CPU only<br>0 : Disable 1 : make by CPU<br>bit[15:14] : router port list aging setting for geometry change detect<br>00 : aging Disable 01 : about 320s<br>10 : about 500s 11 : about 640s | R/W | 0x4000 |
| 0x2A | IGMP packets forwarding configuration<br>bit[3:0] : Query setting<br>bit[7:4] : Leave setting<br>bit[11:8] : unregister mcst data setting<br>bit[15:12] : undefined IGMP type<br>the 4-bit setting for each kind of packet is {drop, router, CPU, bcst}<br>0 : Disable 1 : Enable | R/W | 0x1141 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x2B | IGMP packets forwarding configuration<br>bit[4:0] : Report setting<br>bit[9:5] : Group Specific Query setting<br>bit[14:10] : registered mcst data setting<br>the 5-bit setting is {group member, drop, router, CPU, bcst}<br>0 : Disable 1 : Enable<br>bit[15] : the Hashing method for IGMP<br>0 : CRC 1 : Direct | R/W | 0x5294 |
| 0x2C | IGMP Router List configuration<br>bit[7:0] : Router List setting for port 1 ~ port 8<br>0 : normal port 1 : router port | R/W | 0x0000 |
| 0x2D | IGMP Router List configuration<br>bit[12][8] : Router List setting for port 9 ~ port 10<br>0 : normal port 1 : router port<br>bit[13] : use for Router List clear process<br>0 : no action 1 : clear internal router list setting bits | R/W | 0x0000 |
| 0x2E | Memory access command for Mcst table and Source List table<br>Source List command (bit[8] is 1'b0)<br>bit[5:0] : memory address<br>bit[7:6] : Reserved<br>bit[9:8] : should be 2'b00<br>bit[13:10] : Reserved<br>bit[14] : read/write control<br>0 : read 1 : write<br>bit[15] : command trig/ack<br><br>Mcst table command<br>bit[7:0] : memory address<br>bit[9:8] : should be 2'b01<br>bit[13:10] : Reserved<br>bit[14] : read/write control<br>0 : read 1 : write<br>bit[15] : command trig/ack<br>IP table command<br>bit[6:0] : memory address<br>bit[7] : Reserved<br>bit[9:8] : should be 2'b10<br>bit[13:10] : Reserved<br>bit[14] : read/write control<br>0 : read 1 : write<br>bit[15] : command trig/ack | R/W | 0x0000 |
| 0x2F | Read/Write data bit[15:0] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x30 | Read/Write data bit[31:16] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x31 | Read/Write data bit[47:32] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x32 | Read/Write data bit[63:48] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x33 | Read/Write data bit[79:64] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x34 | Read/Write data bit[95:80] for Mcst table/Source List table/IP table | R/W | 0x0000 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x35 | Read/Write data bit[111:96] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x36 | Read/Write data bit[127:112] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x37 | Read/Write data bit[143:128] Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x38 | Read/Write data bit[159:144] for Mcst table/Source List table/IP table | R/W | 0x0000 |
| 0x39 | Read/Write data<br>bit[4:0] : for Mcst table bit[164:160]<br>bit[15:0] : for Source List table/IP table bit[175:160] | R/W | 0x0000 |
| 0x3A | Read/Write data<br>bit[11:0] : for IP table bit[187:176]<br>bit[15:0] : for Source List table bit[191:176] | R/W | 0x0000 |
| 0x3B<br>~<br>0x56 | Read/Write data bit[207:192] for Source List table<br>~<br>Read/Write data bit[627:624] for Source List table | R/W | 0x0000 |
| 0x57 | Spanning Tree Protocol port state register 1_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x58<br>0x59 | Reserved | | |
| 0x5A | Spanning Tree Protocol port state register 1_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x3FF |
| 0x5B | Spanning Tree Protocol port state register 2_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x5C<br>0x5D | Reserved | | |
| 0x5E | Spanning Tree Protocol port state register 2_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x5F | Spanning Tree Protocol port state register 3_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x60<br>0x61 | Reserved | | |
| 0x62 | Spanning Tree Protocol port state register 3_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x63 | Spanning Tree Protocol port state register 4_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x64<br>0x65 | Reserved | | |
| 0x66 | Spanning Tree Protocol port state register 4_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x67 | Spanning Tree Protocol port state register 5_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x68<br>0x69 | Reserved | | |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x6A | Spanning Tree Protocol port state register 5_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x6B | Spanning Tree Protocol port state register 6_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x6C<br>0x6D | Reserved | | |
| 0x6E | Spanning Tree Protocol port state register 6_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x6F | Spanning Tree Protocol port state register 7_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x70<br>0x71 | Reserved | | |
| 0x72 | Spanning Tree Protocol port state register 7_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x73 | Spanning Tree Protocol port state register 8_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x74<br>0x75 | Reserved | | |
| 0x76 | Spanning Tree Protocol port state register 8_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x77 | Spanning Tree Protocol port state register 9_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x78<br>0x79 | Reserved | | |
| 0x7A | Spanning Tree Protocol port state register 9_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x7B | Spanning Tree Protocol port state register 10_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x7C<br>0x7D | Reserved | | |
| 0x7E | Spanning Tree Protocol port state register 10_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x7F | Spanning Tree Protocol port state register 11_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x80<br>0x81 | Reserved | | |
| 0x82 | Spanning Tree Protocol port state register 11_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x83 | Spanning Tree Protocol port state register 12_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x84<br>0x85 | Reserved | | |
| 0x86 | Spanning Tree Protocol port state register 12_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x87 | Spanning Tree Protocol port state register 13_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x88<br>0x89 | Reserved | | |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x8A | Spanning Tree Protocol port state register 13_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x8B | Spanning Tree Protocol port state register 14_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x8C<br>0x8D | Reserved | | |
| 0x8E | Spanning Tree Protocol port state register 14_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x8F | Spanning Tree Protocol port state register 15_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |
| 0x90<br>0x91 | Reserved | | |
| 0x92 | Spanning Tree Protocol port state register 15_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x93 | Spanning Tree Protocol port state register 16_0<br>bit[15:0] : port state setting for port 1 ~ port 8, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0xFFFF |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x94<br>0x95 | Reserved | | |
| 0x96 | Spanning Tree Protocol port state register 16_1<br>bit[9:8][1:0] : port state setting for port 9 ~ port 10, two bit per port.<br>00 : discard state<br>01 : blocking state<br>10 : learning state<br>11 : forwarding state<br>note : this register for both STP and MSTP | R/W | 0x03FF |
| 0x97 | sFlow enable configuration for port 1 ~ port 8<br>bit[7:0] : enable for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x98 | sFlow enable configuration for port 9 ~ port 10<br>bit[12][8] : enable for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x99 | sFlow sample method, 3 bits per port<br>bit[14:0] : sample method for port 1 ~ port 5<br>000 : inbound 001 : VID<br>010 : ACL 011 : outbound<br>100 : Reserved 101 : LUT_MAC | R/W | 0x0000 |
| 0x9A | sFlow sample method, 3 bits per port<br>bit[8:0] : sample method for port 6 ~ port 8 | R/W | 0x0000 |
| 0x9B | Reserved | | |
| 0x9C | Reserved | | |
| 0x9D | sFlow sample method, 3 bits per port<br>bit[14:12] : sample method for port 9 | R/W | 0x0000 |
| 0x9E | sFlow sample method, 3 bits per port<br>bit[11:9] : sample method for port 10<br>bit[12] : each port use the same sample counter | R/W | 0x0000 |
| 0x9F | sFlow sample rate configure for port 1<br>bit[15:0] : rate configure. (rate = total packets / sampled packet ) | R/W | 0x0064 |
| 0xA0<br>~<br>0xA6 | sFlow sample rate configure for port 2 ~ port 8<br>bit[15:0] : rate configure. (rate = total packets / sampled packet ) | R/W | 0x0064 |
| 0xA7<br>~<br>0xB6 | Reserved | | |
| 0xB7 | sFlow sample rate configure for port 9<br>bit[15:0] : rate configure. (rate = total packets / sampled packet ) | R/W | 0x0064 |
| 0xB8<br>~<br>0xBA | Reserved | | |
| 0xBB | sFlow sample rate configure for port 10<br>bit[15:0] : rate configure. (rate = total packets / sampled packet ) | R/W | 0x0064 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xBC | sFlow sample VID configure for port 1<br>bit[11:0] : VID that want to sampled for sFlow | R/W | 0x000 |
| 0xBD ~ 0xC3 | sFlow sample VID configure for port 2 ~ port 8<br>bit[11:0] : VID that want to sampled for sFlow | R/W | 0x000 |
| 0xC4 ~ 0xD3 | Reserved | | |
| 0xD4 | sFlow sample VID configure for port 9<br>bit[11:0] : VID that want to sampled for sFlow | R/W | 0x000 |
| 0xD5 ~ 0xD7 | Reserved | | |
| 0xD8 | sFlow sample VID configure for port 10<br>bit[11:0] : VID that want to sampled for sFlow | R/W | 0x000 |
| 0xD9 | VLAN mask function configuration<br>bit[11:0] : 12 bits VID mask<br>0 : mask 1 : not mask | R/W | 0xFFF |
| 0xDA | VLAN mask function enable for port 1 ~ port 8<br>bit[7:0] : function enable setting, per port 1 bit<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0xDB | VLAN mask function enable for port 9 ~ port 10<br>bit[12][8]: function enable setting, per port 1 bit<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0xDC ~ 0xDF | Reserved | | |
| 0xE0 | CPU read/write address table configuration<br>bit[12:0] : LUT address<br>bit[13] : LUT block select<br>0 : first block 1 : second block<br>bit[14] : Read/Write configuration<br>0 : Read 1 : Write<br>bit[15] : Command trig/ack | R/W | 0x0000 |
| 0xE1 | CPU read/write address table data<br>bit[15:0] : LUT data[15:0]<br>if the command is read, store the read back data<br>if the command is write, write this data to address table | R/W | 0x0000 |
| 0xE2 | CPU read/write address table data<br>bit[15:0] : LUT data[31:16]<br>if the command is read, store the read back data<br>if the command is write, write this data to address table | R/W | 0x0000 |

| ARL Registers : Page 0x01 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xE3 | CPU read/write address table data<br>bit[15:0] : LUT data[47:32]<br><br>if the command is read, store the read back data<br>if the command is write, write this data to address table | R/W | 0x0000 |
| 0xE4 | CPU read/write address table data<br>bit[15:0] : LUT data[63:48]<br><br>if the command is read, store the read back data<br>if the command is write, write this data to address table | R/W | 0x0000 |
| 0xE5 | CPU read/write address table data<br>bit[4:0] : LUT data[68:64]<br>bit[5] : Table address access method<br>0 : address calculated by CPU<br>1 : address calculated by HW for normal MAC entry<br><br>if the command is read, store the read back data<br>if the command is write, write this data to address table | R/W | 0x00 |
| 0xE6 | IGMP join / leave enable for each port<br>bit[7:0] : enable join / leave for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0xFFFF |
| 0xE7 | IGMP join / leave enable for each port<br>bit[12][8] : enable join / leave for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x1FFF |
| 0xE8 | ARP storm drop event for port 1 ~ port 8<br>bit[7:0] : indicate the ARP storm drop event for port 1 ~ port 8 | R/C | 0x0000 |
| 0xE9 | ARP storm drop event for port 9 ~ port 10<br>bit[12][8] : indicate the ARP storm drop event for port 9 ~ port 10 | R/C | 0x0000 |
| 0xEA | ICMP storm drop event for port 1 ~ port 8<br>bit[15:0] : indicate the ICMP storm drop event for port 1 ~ port 8 | R/C | 0x0000 |
| 0xEB | ICMP storm drop event for port 9 ~ port 10<br>bit[12][8]: indicate the ICMP storm drop event for port 9 ~ port 10 | R/C | 0x0000 |
| 0xEC ~ 0xEF | Reserved | | |
| 0xF0 | Multicast VLAN Group 1 setting<br>bit[7:0] : VLAN member setting for port 1 ~ port 8 | R/W | 0x0000 |
| 0xF1 | Multicast VLAN Group 1 setting<br>bit[12][8]: VLAN member setting for port 9 ~ port 10 | R/W | 0x0000 |
| 0xF2 ~ 0xFD | Multicast VLAN Group 2 ~ 7 setting | R/W | 0x0000 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.5 ARL Control Register Page 2

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | Configuration for IP filter function<br>bit[7:0] : IP filter function enable for port 1 ~ port 8 | R/W | 0x0000 |
| 0x02 | Configuration for IP filter function<br>bit[12][8] : IP filter function enable for port 9 ~ port 10<br>bit[13] : filter mode selection<br>0 : drop if IP mis-match 1 : drop if IP match<br>bit[14] : enable pass for IP equal 0.0.0.0<br>0 : Disable 1 : Enable<br>bit[15] : Hashing method selection<br>0 : CRC hashing 1 : direct hashing | R/W | 0x0000 |
| 0x03 ~ 0x05 | Reserved | | |
| 0x06 | VLAN configuration<br>bit[0] : protocol based VLAN enable<br>0 : Disable 1 : Enable<br>bit[1] : tag based VLAN enable<br>0 : Disable 1 : Enable<br>bit[2] : SVL/IVL select<br>0 : SVL 1 : IVL<br>bit[3] : select the tag/untag method<br>0 : based on ports 1 : based on VIDs<br>bit[4] : Modify tag in VID tag mode for CPU port<br>0 : Don't modify<br>1 : modify tag according to settings<br>bit[5] : CTAG Remove/Modify enable<br>0 : Don't change CTAG<br>1 : remove/modify CTAG enable<br>bit[6] : MSTP enable<br>0 : Disable 1 : Enable<br>bit[7] : always use CTAG VID<br>0 : use CTAG / STAG ( when STAG exist)<br>1 : use CTAG (even with STAG)<br>bit[8] : Multicast Group specify VLAN Group<br>0 : Disable 1 : Enable<br>bit[9] : VID priority works with the priority bits remark function<br>0 : Disable 1 : Enable<br>bit[14:10] : inactive VID redirect<br>0 ~ 7,24,28 : redirect to port 1 ~ port 10<br>30 : broadcast<br>others : drop<br>bit[15] : inactive VID redirect enable<br>0 : Disable 1 : Enable | R/W | 0x0000 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x07 | Insert PVID to the output port<br>bit[7:0] : insert PVID to the output frame for port 1 ~ port 8<br>0 : Disable<br>1 : insert PVID to the current port<br>will replace the VID of tag frame to PVID | R/W | 0x0000 |
| 0x08 | Insert PVID to the output port<br>bit[12][8] : insert PVID to the output frame for port 9 ~ port 10<br>0 : Disable<br>1 : insert PVID to the current port<br>will replace the VID of tag frame to PVID | R/W | 0x0000 |
| 0x09 | VLAN ingress check configuration (802.1Q 8.6.2)<br>bit[15:0] : check if the port is the VLAN member for port 1 ~ port 8<br>0 : Don't check if the ingress port is in the VLAN group<br>1 : check if the ingress port is in the VLAN group | R/W | 0x0000 |
| 0x0A | VLAN ingress check configuration (802.1Q 8.6.2)<br>bit[12][8]: check if the port is the VLAN member for port 9 ~ port 10<br>0 : Don't check if the ingress port is in the VLAN group<br>1 : check if the ingress port is in the VLAN group | R/W | 0x0000 |
| 0x0B | VLAN ingress check for Frame Type<br>bit[15:0] : the ingress frame type check for port 1 ~ port 8, 2-bit/port<br>00 : admit all<br>01 : admit tag frame only<br>10 : admit priority tag and non-tag frame<br>11 : admit priority tag and tag frame | R/W | 0x0000 |
| 0x0C<br>0x0D | Reserved | | |
| 0x0E | VLAN ingress check for Frame Type<br>bit[9:8][1:0] : the ingress frame type check for port 9 ~ port 10, 2-bit/port<br>00 : admit all<br>01 : admit tag frame only<br>10 : admit priority tag and non-tag frame<br>11 : admit priority tag and tag frame | R/W | 0x000 |
| 0x0F | VLAN egress configuration. Set the criteria for the output to use default group setting.<br>bit[7:0] : by port. One bit per port, for port 1 ~ port 8<br>0 : Don't use default group setting<br>1 : use default group setting | R/W | 0x0000 |
| 0x10 | VLAN egress configuration. Set the criteria for the output to use default group setting.<br>bit[12][8] : by port. One bit per port, for port 9 ~ port 10<br>0 : Don't use default group setting<br>1 : use default group setting<br>bit[13] : unicast frame<br>0 : Don't use default group setting<br>1 : use default group setting | R/W | 0x0000 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[14] : multicast frame (DA equal to 01:00:5E:xx:xx:xx and IGMP need to be enabled)<br>0 : Don't use default group setting<br>1 : use default group setting<br>bit[15] : ARP frame<br>0 : Don't use default group setting<br>1 : use default group setting | | |
| 0x11 | Allow for local traffic<br>bit[7:0] : enable local traffic forwarding for port 1 ~ port 8<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x12 | Allow for local traffic<br>bit[12][8] : enable local traffic forwarding for port 9 ~ port 10<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0x13 | port 1 PVID configuration<br>bit[11:0] : PVID setting<br>bit[14:12] : PVID priority bits setting<br>bit[15] : Force use PVID as VID for VLAN checking<br>0 : Disable 1 : Enable | R/W | 0x0001 |
| 0x14 ~ 0x1A | port 2 ~ port 8 PVID configuration<br>bit[11:0] : PVID setting<br>bit[14:12] : PVID priority bits setting<br>bit[15] : Force use PVID as VID for VLAN checking<br>0 : Disable 1 : Enable | R/W | 0x0001 |
| 0x1B ~ 0x2A | Reserved | | |
| 0x2B | port 9 PVID configuration<br>bit[11:0] : PVID setting<br>bit[14:12] : PVID priority bits setting<br>bit[15] : Force use PVID as VID for VLAN checking<br>0 : Disable 1 : Enable | R/W | 0x0001 |
| 0x2C ~ 0x2E | Reserved | | |
| 0x2F | port 10 PVID configuration<br>bit[11:0] : PVID setting<br>bit[14:12] : PVID priority bits setting<br>bit[15] : Force use PVID as VID for VLAN checking<br>0 : Disable 1 : Enable | R/W | 0x0001 |
| 0x30 | port 1 Port based VLAN configuration<br>bit[7:0] : Port based VLAN group (port 1 ~ port 8) setting | R/W | 0xFFFF |
| 0x31 | port 1 Port based VLAN configuration<br>bit[12][8]: Port based VLAN group port 9 ~ port 10) setting<br>bit[13] : Force Port 1 use Port based VLAN | R/W | 0x1FFF |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x32 ~ 0x3F | port 2 ~ port 8 Port based VLAN configuration | R/W | 0xFFFF 0x1FFF |
| 0x40 ~ 0x5F | Reserved | | |
| 0x60 0x61 | port 9 Port based VLAN configuration | R/W | 0xFFFF 0x1FFF |
| 0x68 0x69 | port 10 Port based VLAN configuration | R/W | 0xFFFF 0x1FFF |
| 0x6A | Protocol based VLAN entry 1 configuration<br>bit[15:0] : TYPE for protocol define in entry 1 | R/W | 0x0000 |
| 0x6B | Protocol based VLAN entry 1 configuration<br>bit[11:0] : VID for protocol define in entry 1<br>bit[13:12] : Entry A protocol selection<br>00 : invalid<br>01 : EtherType<br>10 : LLC<br>11 : RFC 1042 | R/W | 0x0000 |
| 0x6C ~ 0x89 | 0x6C ~ 0x71 for Protocol based VLAN entry 2 ~ 4 configuration<br>Note : 0x6A ~ 0x89 also used for ACL VID remarking. | R/W | 0x0000 0x0000 |
| 0x8A | ACL Assigned VLAN Group 1<br>bit[7:0] : VLAN Group 01 member setting for port 1 ~ port 8<br>0 : not member 1 : member | R/W | 0x0000 |
| 0x8B | ACL Assigned VLAN Group 1<br>bit[12][8] : VLAN Group 01 member setting for port 9 ~ port 10<br>0 : not member 1 : member | R/W | 0x0000 |
| 0x8C ~ 0xC9 | ACL Assigned VLAN Group 2 ~ 32 configuration | R/W | 0x0000 0x0000 |
| 0xCA | VLAN exclusive setting for port 1 ~ port 8<br>bit[7:0] : 0: this port is normal port<br>1: this port is protected port | R/W | 0x0000 |
| 0xCB | VLAN exclusive setting for port 9 ~ port 10<br>bit[12][8]: 0: this port is normal port<br>1: this port is protected port | R/W | 0x0000 |
| 0xCC | VLAN output port add tag setting for port 1 ~ port 8<br>bit[7:0] : output port add tag setting<br>0 : Don't add 1 : add | R/W | 0x0000 |
| 0xCD | VLAN output port add tag setting for port 9 ~ port 10<br>bit[12][8] : output port add tag setting<br>0 : Don't add 1 : add | R/W | 0x0000 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xCE | VLAN output port remove tag setting for port 1 ~ port 8<br>bit[7:0] : output port remove tag setting<br>0 : Don't remove 1 : remove | R/W | 0x0000 |
| 0xCF | VLAN output port remove tag setting for port 9 ~ port 10<br>bit[12][8] : output port remove tag setting<br>0 : Don't remove 1 : remove | R/W | 0x0000 |
| 0xD0 | VLAN uplink function configuration<br>bit[7:0] : uplink port configuration for port 1 ~ port 8<br>0 : normal port 1 : uplink port | R/W | 0x0000 |
| 0xD1 | VLAN uplink function configuration<br>bit[12][8]: uplink port configuration for port 9 ~ port 10<br>0 : normal port 1 : uplink port<br>bit[13] : uplink port function enable<br>0 : Disable 1 : Enable | R/W | 0x0000 |
| 0xD2 | VLAN Table Access command register<br>bit[11:0] : VID<br>bit[12] : sequential Read<br>bit[13] : Reserved<br>bit[14] : Read/Write 0 : Read 1 : Write<br>bit[15] : command trig/ack | R/W | 0x0000 |
| 0xD3 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[15:0] | R/W | 0x0000 |
| 0xD4 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[31:16] | R/W | 0x0000 |
| 0xD5 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[47:32] | R/W | 0x0000 |
| 0xD6 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[63:48] | R/W | 0x0000 |
| 0xD7 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[79:64] | R/W | 0x0000 |
| 0xD8 | VLAN Table Access data register<br>bit[15:0] : Read/Write VLAN table data[95:80] | R/W | 0x0000 |
| 0xD9 | ACL Bandwidth threshold setting 0 ~ 1<br>bit[6:0] : ACL bw0 threshold<br>bit[7] : unit selection 0 : 128Kbps, 1 : 2Mbps<br>bit[14:8] : ACL bw1 threshold<br>bit[15] : unit selection 0 : 128Kbps, 1 : 2Mbps | R/W | 0x0000 |
| 0xDA ~ 0xDF | ACL Bandwidth threshold setting 2 ~ 13 | R/W | 0x0000 |
| 0xE0 | ACL Bandwidth threshold setting 14<br>bit[6:0] : ACL bw14 threshold<br>bit[7] : unit selection<br>0 : 128Kbps, 1 : 2Mbps | RW | 0x00 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xE1 | ACL user define pattern location setting<br>bit[4:0] : pattern 1 location setting<br>bit[9:5] : pattern 2 location setting | R/W | 0x000 |
| 0xE2 | ACL user define pattern location setting<br>bit[4:0] : pattern 3 location setting<br>bit[9:5] : pattern 4 location setting | R/W | 0x000 |
| 0xE3 | ACL Table Access command register<br>bit[6:0] : entries select<br>bit[7] : sequential read<br>bit[8] : table select 0 : rule 1 : action<br>bit[14] : Read/ Write 0 : Read 1 : Write<br>bit[15] : command trig | R/W | 0x0000 |
| 0xE4 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[15:0] | R/W | 0x0000 |
| 0xE5 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[31:16] | R/W | 0x0000 |
| 0xE6 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[47:32] | R/W | 0x0000 |
| 0xE7 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[63:48] | R/W | 0x0000 |
| 0xE8 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[79:64] | R/W | 0x0000 |
| 0xE9 | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[95:80] | R/W | 0x0000 |
| 0xEA | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[111:96] | R/W | 0x0000 |
| 0xEB | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[127:112] | R/W | 0x0000 |
| 0xEC | ACL entries access data register<br>bit[15:0] : read/write ACL entry data[143:128] | R/W | 0x0000 |
| 0xED | ACL entries access data register<br>bit[6:0] : read/write ACL entry data[150:144] | R/W | 0x00 |
| 0xEE | Configuration for VID/PVID match behavior setting bit0 for port 1 ~ port 8<br>(not affect non_tag/priority_tag)<br>bit[7:0] : bit 0 setting for port 1 ~ port 8 | R/W | 0x0000 |
| 0xEF | Configuration for VID/PVID match behavior setting bit0 for port 9 ~ port 10<br>(not affect non_tag/priority_tag)<br>bit[12][8] : bit 0 setting for port 9 ~ port 10 | R/W | 0x0000 |
| 0xF0 | Configuration for VID/PVID match behavior setting bit1 for port 1 ~ port 8<br>bit[7:0] : bit 1 setting for port 1 ~ port 8 | R/W | 0x0000 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xF1 | Configuration for VID/PVID match behavior setting bit1 for port 9 ~ port 10<br>bit[12][8]: bit 1 setting for port 9 ~ port 10<br>The behavior for {bit1, bit0} is :<br>00 : Disable<br>01 : drop if VID equals to PVID<br>10 : drop if VID does not equal to PVID<br>11 : act with Multicast SRC function, to pass VID equal to PVID | R/W | 0x0000 |
| 0xF2 | Configuration Source Ports for Multicast SRC function<br>bit[7:0] : source port setting for port 1 ~ port 8 | R/W | 0x0000 |
| 0xF3 | Configuration Source Ports for Multicast SRC function<br>bit[12][8] : source port setting for port 9 ~ port 10 | R/W | 0x0000 |
| 0xF4 | LUT memory repair control<br>bit[0] : disable the repair entry 0<br>0 : enable, 1 : disable<br>bit[1] : disable the repair entry 1<br>bit[2] : disable the repair entry 2<br>bit[3] : disable the repair entry 3<br>bit[5:4] : repair entry read back selection | R/W | 0x00 |
| 0xF5 | LUT memory repair status<br>bit[12:0] : the failed memory address<br>bit[13] : the failed memory block<br>0 : low bits block, 1 : high bits block<br>bit[14] : Valid bit<br>0 :invalid 1 : valid | RO | 0x0000 |
| 0xF6 | MAC Based VLAN configuration<br>bit[0] : MAC based vlan enable<br>0 : disable 1 : enable<br>bit[1] : the related LUT entry need to be static<br>0 : disable 1 : enable<br>bit[2] : CPU enable for MAC based vlan<br>0 : disable 1 : enable<br>bit[3] : assign VLAN for the unknown SA packets<br>0 : disable 1 : enable | R/W | 0x0 |
| 0xF7 | CPU read/write MAC based VLAN Table configuration<br>bit[12:0] : Table address<br>bit[13] : block selection (the same as LUT)<br>bit[14] : read/write select<br>0 : read 1 : write<br>bit[15] : command trigger | R/W | 0x0000 |
| 0xF8 | CPU write MAC based VLAN Table data<br>bit[12:0] : Table data | R/W | 0x0000 |
| 0xF9 | CPU read MAC based VLAN Table data (first block)<br>bit[12:0] : Table data | R/W | 0x0000 |

| ARL Registers : Page 0x02 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xFA | CPU read MAC based VLAN Table data (second block)<br>bit[12:0] : Table data | R/W | 0x0000 |
| 0xFB | VID configuration for unknown SA<br>bit[11:0] : VID<br>bit[12] : enable | R/W | 0x0000 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.6 SMI Control Register

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | Auto negotiation configuration for port 1 ~ port 8<br>bit[7:0] : 0 : Disable auto negotiation<br>1 : Enable auto negotiation | R/W | 0xFFFF |
| 0x02 | Auto negotiation configuration for port 9 ~ port 10<br>bit[12][8] : 0 : Disable auto negotiation<br>1 : Enable auto negotiation | R/W | 0x1FFF |
| 0x03 | Speed setting for 1000 Mbps for port 9 ~ port 10<br>bit[4][0] : 0 : 1000 Mbps ability disable<br>1 : 1000 Mbps ability enable | R/W | 0x1F |
| 0x04 | Speed setting for 10/100 Mbps for port 1 ~ port 8<br>bit[7:0] : 0 : 10 Mbps<br>1 : 100 Mbps | R/W | 0xFFFF |
| 0x05 | Speed setting for 10/100 Mbps for port 9 ~ port 10<br>bit[12][8] : 0 : 10 Mbps<br>1 : 100 Mbps | R/W | 0x1FFF |
| 0x06 | Duplex setting for port 1 ~ port 8<br>bit[7:0] : 0 : half duplex<br>1 : full duplex | R/W | 0xFFFF |
| 0x07 | Duplex setting for port 9 ~ port 10<br>bit[12][8] : 0 : half duplex<br>1 : full duplex | R/W | 0x1FFF |
| 0x08 | Pause setting for port 1 ~ port 8<br>bit[15:0] : 0 : Disable pause<br>1 : Enable pause | R/W | 0xFFFF |
| 0x09 | Pause setting for port 9 ~ port 10<br>bit[12][8] : 0 : Disable pause<br>1 : Enable pause | R/W | 0x1FFF |
| 0x0A | Asymmetric pause setting for port 1 ~ port 8<br>bit[7:0] : 0 : Disable asymmetric pause<br>1 : Enable asymmetric pause | R/W | 0xFFFF |
| 0x0B | Asymmetric pause setting for port 9 ~ port 10<br>bit[12][8] : 0 : Disable asymmetric pause<br>1 : Enable asymmetric pause | R/W | 0x1FFF |
| 0x0C | Backpressure setting for port 1 ~ port 8<br>bit[7:0] : 0 : backpressure function disable<br>1 : backpressure function enable | R/W | 0xFFFF |
| 0x0D | Backpressure setting for port 9 ~ port 10<br>bit[12][8] : 0 : backpressure function disable<br>1 : backpressure function enable | R/W | 0x1FFF |
| 0x0E | Power down setting for port 1 ~ port 8<br>bit[7:0] : 0 : power down disable<br>1 : power down enable | R/W | 0x0000 |
| 0x0F | Power down setting for port 9 ~ port 10<br>bit[12][8] : 0 : power down disable<br>1 : power down enable | R/W | 0x0000 |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x10 | WOL function enable for port 9<br>bit[0] : 0 : WOL disable<br>1 : WOL enable<br>WOL function will be disabled when both IP1810I WOL and PHY WOL settings are disabled.<br>bit[3:1] : Reserved<br>bit[4] : port 9 WOL sense ANY packet to wake up<br>bit[5] : port 9 WOL sense MAGIC packet to wake up<br>bit[11:6] : Reserved | R/W | 0x0FF0 |
| 0x11 | PHY address setting for port 9 ~ 10<br>bit[2:0] : port 9 PHY address, default 3'd2<br>bit[11:3] : Reserved<br>bit[14:12] :port 10 PHY address, default 3'd1 | R/W | 0x1B1A |
| 0x12 | SMI MISC setting<br>bit[0] : port 10 force link set<br>0 : force link Disable<br>1 : force link Enable<br>bit[1] : port 10 force duplex pin setting (RO)<br>0 : half duplex<br>1 : full duplex<br>bit[3:2] : port 10 force speed pin setting (RO)<br>00 : 10 Mbps<br>01 : 100 Mbps<br>10 : 1000 Mbps<br>11 : Reserved<br>bit[7:4] : Reserved<br>bit[8] : port 10 force status (according to 0x03/0x05/0x07/0x09/0x0B)<br>0 : Disable<br>1 : Enable<br>bit[9] : port 9 force PAUSE<br>0 : Disable<br>1 : Enable<br>bit[15:10] : Reserved | R/W | 0x003E |
| 0x13 | CPU read/write PHY register command<br>bit[4:0] : PHY address<br>bit[9:5] : MII register address<br>bit[10] : write all TP (port 1 ~ port 8)<br>bit[13:11] : Reserved<br>bit[14] : 0 : read operation<br>1 : write operation<br>bit[15] : the read/write PHY register command trigger (RC)<br>0 : idle<br>1 : start command/ command complete | R/W<br>RC | 0x0000 |
| 0x14 | CPU read/write PHY register command data<br>bit[15:0] : in read command – the read back data<br>in write command – data want to write | R/W | 0x0000 |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x15 | SCA startup status for port 1 ~ port 8 (fiber port not support)<br>bit[7:0] : 0 : cable test fail<br>1 : cable test good | RO | 0x0000 |
| 0x16 | SCA startup status for port 9 ~ port 10 (fiber port not support)<br>bit[12][8] : 0 : cable test fail<br>1 : cable test good<br>bit[13] : SCA period setting<br>0 : run 5 seconds show 3 seconds<br>1 : run 10 seconds show 5 seconds<br>bit[14] : SCA enable<br>0 : Disable<br>1 : Enable<br>bit[15] : SCA trig (only work when bit[14] is asserted)<br>0 : no action<br>1 : trig SCA and do it once | R/W | 0x0000 |
| 0x17 | LED Setting<br>bit[2:0] : LED mode ( B : bi-color; M : mono-color)<br>000 : 3-bit M (Giga-Spd, L/A, Dupx/Col) @ GE port<br>(100-Spd, L/A, Dupx/Col) @ FE port<br>001 : 3-bit M (Giga-L/A, 10/100-L/A, Dupx/Col) @ GE port<br>(100-L/A, 10-L/A, Dupx/Col) @ FE port<br>010 : 3-bit M (Giga-Spd, 10/100-Spd, /A) @ GE port<br>(100-Spd, 10-Spd, /A) @ FE port<br>011 : 2-bit M (Giga-Spd, L/A) @ GE port<br>(100-Spd, L/A) @ FE port<br>100 : 2-bit B, 1-bit M (**Spd & L/A**, PSE) @ GE port<br>(**Spd1 & L/A**, PSE) @ FE port<br>101 : 2-bit M (L, /A)<br>110 : 2-bit B (**Spd & L/A)** @ GE port<br>(**Spd1 & L/A**) @ FE port<br>111 : 2-bit B (**Spd & L/A**) @ GE port<br>and<br>1-bit M (L/A) @ FE port<br>(Reference 0x1E, 0x1F for PSE setting)<br>bit[3] : Reserved<br>bit[5:4] : LED blink rate<br>00 : 40 ms<br>01 : 80 ms<br>10 : 120 ms<br>11 : 160 ms<br>bit[7:6] : LED clock rate<br>00 : 781 KHz<br>01 : 2.5 MHz<br>10 : 5.2 MHz<br>11 : 10.4 MHz<br>bit[8] : LED col blink disable<br>0 : blink when collision<br>1 : no blink when collision | R/W<br>RC | 0x00C3 |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[9] : LED act (tx/rx) blink disable<br>0 : blink when act<br>1 : no blink when act<br>bit[10] : LED loop-detect indication<br>0 : disable<br>1 : enable<br>bit[11] : LED loop-detect indication time<br>0 : short pattern<br>1 : long pattern<br>bit[12] : LED power-saving set.<br>0 : disable<br>1 : enable<br>bit[13] : LED power-saving trig, and this bit will work when bit[12] is asserted. (RC)<br>0 : non action<br>1 : trig<br>bit[14] : LED power-saving time<br>0 : LED trig show time is 10s<br>1 : LED trig show time is 30s<br>bit[15] : CPU control serial-LED function<br>0 : disable<br>1 : enable | | |
| 0x18 | LED port selection for port 1 ~ port 8<br>bit[7:0] : LED port selection | R/W | 0xFFFF |
| 0x19 | LED port selection for port 9 ~ port 10<br>bit[12][8] : LED port selection<br>bit[13] : Reserved<br>bit[14] : port 10 LED CPU control enable<br>0 : Disable<br>1 : Enable<br>(The corresponding signals for led_0/led_1/led_2, reference to register 0x1B[12]/0x1D[12]/0x1F[12])<br>bit[15] : Reserved | R/W | 0x1FFF |
| 0x1A | CPU control serial-LED setting for port 1 ~ port 8<br>bit[7:0] : serial-LED(led_0) setting | R/W | 0xFFFF |
| 0x1B | CPU control serial-LED setting for port 9 ~ port 10<br>bit[12][8] : serial-LED(led_0) setting | R/W | 0x1FFF |
| 0x1C | CPU control serial-LED setting for port 1 ~ port 8<br>bit[7:0] : serial-LED(led_1) setting | R/W | 0xFFFF |
| 0x1D | CPU control serial-LED setting for port 9 ~ port 10<br>bit[12][8] : serial-LED(led_1) setting | R/W | 0x1FFF |
| 0x1E | CPU control serial-LED setting for port 1 ~ port 8<br>bit[7:0] : serial-LED(led_2) setting | R/W | 0xFFFF |
| 0x1F | CPU control serial-LED setting for port 9 ~ port 10<br>bit[12][8] : serial-LED(led_2) setting<br>(reference to 0x17 LED mode = 3'b100, when enable PSE, the led_2 will be controlled via 0x1E & 0x1F) | R/W | 0x1FFF |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x20 | SMI port status for port 1 ~ port 2 ( 7bit / port )<br>bit[6:0] : port 1 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link}<br>bit[7] : Reserved<br>bit[14:8] : port 2 status<br>flow_ctrl: include backpressure in half duplex<br>speed definition:<br>1x: 1000 Mbps<br>01: 100 Mbps<br>00: 10 Mbps | RO | 0x7A7A |
| 0x21 | SMI port status for port 3 ~ port 4 ( 7bit / port )<br>bit[6:0] : port 3 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link}<br>bit[7] : Reserved<br>bit[14:8] : port 4 status | RO | 0x7A7A |
| 0x22 | SMI port status for port 5 ~ port 6 ( 7bit / port )<br>bit[6:0] : port 5 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link}<br>bit[7] : Reserved<br>bit[14:8] : port 6 status | RO | 0x7A7A |
| 0x23 | SMI port status for port 7 ~ port 8 ( 7bit / port )<br>bit[6:0] : port 7 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link}<br>bit[7] : Reserved<br>bit[14:8] : port 8 status | RO | 0x7A7A |
| 0x24<br>~<br>0x2B | Reserved | | |
| 0x2C | SMI port status for port 9 ( 7bit / port )<br>bit[6:0] : port 9 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link}<br>bit[15:7] : Reserve | RO | 0x3E3E |
| 0x2D | Reserved | | |
| 0x2E | SMI port status for port 10 ( 7bit / port )<br>bit[6:0] : port 10 status<br>{AN enable, asymmetric pause, flow_ctrl, duplex,speed[1:0], link} | RO | 0x3E |
| 0x2F | EEE polling enable for port 1 ~ port 8<br>bit[7:0] : EEE port enable | R/W | 0xFFFF |
| 0x30 | EEE polling enable for port 9 ~ port 10<br>bit[12][8] : EEE port enable<br>0 : Disable<br>1 : Enable<br>bit[13] : MMD register read/write enable<br>0 : Disable<br>1 : Enable<br>bit[14] : Reserved | R/W | 0x1FFF |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[15] : EEE check duplex enable<br>0 : Disable<br>1 : Enable | | |
| 0x31 | MMD read/write command<br>bit[4:0] : PHY address<br>bit[9:5] : DEVICE address<br>bit[13:10] : Reserved<br>bit[14] : 0 : read operation<br>1 : write operation<br>bit[15] : the read/write PHY register command trigger (RC)<br>0 : idle<br>1 : start command/ command complete | R/W<br>RC | 0x0000 |
| 0x32 | MMD read/write ADDRESS<br>bit[15:0] : ADDRESS | R/W | 0x0000 |
| 0x33 | MMD write DATA<br>bit[15:0] : write data | R/W | 0x0000 |
| 0x34 | MMD read DATA<br>bit[15:0] : read data | R/W | 0x0000 |
| 0x35 | EEE auto-negotiation ability for port 1 ~ port 8<br>bit[7:0] : 0 : AN EEE disable<br>1 : AN EEE enable | RO | 0x0000 |
| 0x36 | EEE auto-negotiation ability for port 9 ~ port 10<br>bit[12][8] : 0 : AN EEE disable<br>1 : AN EEE enable | RO | 0x0000 |
| 0x37 | EEE Tw for port 1<br>bit[8:0] : port 1 EEE Tw | RO | 0x014 |
| 0x38 | EEE Tw for port 2<br>bit[8:0] : port 2 EEE Tw | RO | 0x014 |
| 0x39 | EEE Tw for port 3<br>bit[8:0] : port 3 EEE Tw | RO | 0x014 |
| 0x3A | EEE Tw for port 4<br>bit[8:0] : port 4 EEE Tw | RO | 0x014 |
| 0x3B | EEE Tw for port 5<br>bit[8:0] : port 5 EEE Tw | RO | 0x014 |
| 0x3C | EEE Tw for port 6<br>bit[8:0] : port 6 EEE Tw | RO | 0x014 |
| 0x3D | EEE Tw for port 7<br>bit[8:0] : port 7 EEE Tw | RO | 0x014 |
| 0x3E | EEE Tw for port 8<br>bit[8:0] : port 8 EEE Tw | RO | 0x014 |
| 0x3F ~ 0x4E | Reserved | | |
| 0x4F | EEE Tw for port 9<br>bit[8:0] : port 9 EEE Tw | RO | 0x014 |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x50 ~ 0x52 | Reserved | | |
| 0x53 | EEE Tw for port 10<br>bit[8:0] : port 10 EEE Tw | RO | 0x014 |
| 0x54 | For PHY status register, register address and page setting<br>bit[4:0] : PHY status register address (default 5'd17)<br>bit[9:5] : PHY status page selection (default 5'd15) | R/W | 0x01F1 |
| 0x55 | For PHY status register, bit selection (select 2 bits)<br>bit[15:0] : bit selection for PHY status register (default 16'h14) | R/W | 0x0014 |
| 0x56 | For PHY status register, read enable for port 1 ~ port 8<br>bit[7:0] : PHY status register address | R/W | 0x0000 |
| 0x57 | For PHY status register, read enable for port 9 ~port 10<br>bit[12][8] : PHY status register address | R/W | 0x0000 |
| 0x58 | For PHY status register, 100 fiber status for port 1 ~port 8<br>bit[7:0] : running on 100 fiber | RO | 0x0000 |
| 0x59 | For PHY status register, 100 fiber status for port 9 ~ port 10<br>bit[12][8] : running on 100 fiber | RO | 0x0000 |
| 0x5A | For PHY status register, 1000 fiber status for port 1 ~ port 8<br>bit[7:0] : running on 1000 fiber | RO | 0x0000 |
| 0x5B | For PHY status register, 1000 fiber status for port 9 ~ port 10<br>bit[12][8] : running on 1000 fiber | RO | 0x0000 |
| 0x5C | Force link setting for port 1 ~ port 8<br>bit[7:0] : force link enable | R/W | 0x0000 |
| 0x5D | Force link setting for port 9<br>bit[8] : force link enable<br>bit[13:12] : Reserved<br>bit[14] : Test mode polling PHY enable<br>0 : Disable (default)<br>1 : Enable<br>bit[15] : Bypass PHY enable<br>0 : Disable (default)<br>1 : Enable | R/W | 0x0000 |
| 0x5E | Reserved | | |
| 0x5F | SCA setting<br>bit[4][0] : SCA enable for port 9 ~ port 10<br>0 : Disable (default)<br>1 : Enable<br>bit[9][5]: SCA mode for port 9 ~ port 10<br>0 : running as GE PHY (default)<br>1 : running as FE PHY<br>bit[15:10] : Reserved | R/W | 0x0000 |
| 0x60 | EEE local Tw register address of MMD at 1000Mbps<br>bit[15:0] : MMD register address (0x60, 0x61 and 0x62 a set) | R/W | 0x0004 |
| 0x61 | EEE local Tw device address of MMD at 1000Mbps<br>bit[4:0] : MMD device address | R/W | 0x1E |

| SMI Registers : Page 0x03 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x62 | EEE local Tw bit number of MMD at 1000Mbps<br>bit[15:0] : MMD bit number of the selected register<br>(select 9-bit of 16-bit, ex 0x01FF select the lowest 9 bits) | R/W | 0x01FF |
| 0x63 | EEE local Tw register address of MMD at 100Mbps<br>bit[15:0] : MMD register address (0x63, 0x64 and 0x65 a set) | R/W | 0x0003 |
| 0x64 | EEE local Tw device address of MMD at 100Mbps<br>bit[4:0] : MMD device address | R/W | 0x1E |
| 0x65 | EEE local Tw bit number of MMD at 100Mbps<br>bit[15:0] : MMD bit number of the selected register<br>(select 9-bit of 16-bit, ex 0x01FF select the lowest 9 bits) | R/W | 0x01FF |
| 0x66 | Uni-direction enable for port 1 ~ port 8<br>bit[7:0] : 0 : uni-direction disable 1 : uni-direction enable | R/W | 0x0000 |
| 0x67 | Uni-direction enable for port 9 ~ port 10<br>bit[12][8] : 0 : uni-direction disable 1 : uni-direction enable | R/W | 0x0000 |
| 0x68 ~ 0xFE | Reserved | | |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.7 OAM Control Register

| OAM Registers 1 : Page 0x04 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | OAM Information OAMPDU OUI setting<br>bit[15:0] : Information OUI[15:0] | R/W | 0x90C3 |
| 0x02 | OAM Information OAMPDU OUI setting<br>bit[7:0] : Information OUI[23:16] | R/W | 0x00 |
| 0x03 | OAM Information OAMPDU VENDOR setting<br>bit[15:0] : Information VENDOR[15:0] | R/W | 0x0000 |
| 0x04 | OAM Information OAMPDU VENDOR setting<br>bit[15:0] : Information VENDOR[31:16] | R/W | 0x0000 |
| 0x05 | OAM indirect register command<br>bit[4:0] : port selection<br>bit[9:5] : register selection<br>bit[13:10] : Reserved<br>bit[14] : read/write<br>0 : read command<br>1 : write command<br>bit[15] : the read/write OAM indirect register command trigger (RC)<br>0 : idle<br>1 : start command(for write)/ command complete(for read)<br>(reference the OAM indirect registers) | R/W<br>RC | 0x0000 |
| 0x06 | OAM indirect register data<br>bit[15:0] : in read command – the read back data<br>in write command – data want to write<br>(reference the OAM indirect registers) | R/W | 0x0000 |
| 0x07 | OAM Dying Gasp Independent enable for port 1 ~ port 8<br>bit[7:0] : OAM Dying Gasp Independent enable<br>(Using this function, do not enable OAM by asserting indirect register 0x00 bit[0] OAM enable) | R/W | 0x0000 |
| 0x08 | OAM Dying Gasp Independent enable for port 9 ~ port 10<br>bit[12][8] : OAM Dying Gasp Independent enable<br>(Using this function, do not enable OAM by asserting indirect register 0x00 bit[0] OAM enable) | R/W | 0x0000 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.8 OAM Indirect Registers for Each Port

| OAM Indirect Registers | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x00 | OAM configuration<br>bit[0] : OAM enable<br>0 : Disable<br>1 : Enable<br>bit[1] : OAM configuration trig<br>bit[2] : OAM remote LoopBack trig<br>bit[3] : OAM reset<br>bit[4] : Information OAMPDU configuration Mode<br>(When setting Mode, bit[1] should be set as “1.”)<br>0 : passive mode<br>1 : active mode<br>bit[5] : Information OAMPDU configuration Remote LoopBack support<br>(When setting Remote LoopBack ability, bit[1] should be set as “1.”)<br>0 : Disable<br>1 : Enable<br>bit[6] : OAM remote read/write enable<br>0 : Disable<br>1 : Enable<br>bit[7] : OAM remote read/write check OUI<br>0 : Disable<br>1 : Enable<br>bit[8] : OAM LoopBack timeout enable (timeout time is 5 seconds)<br>0 : Disable<br>1 : Enable<br>bit[9] : OAM LoopBack test result Read Clear enable, please refer to indirect register 0x08 bit[6] and bit[7].<br>0 : Disable<br>1 : Enable<br>bit[10] : OAM critical event register setting (Should check setting with bit[11] )<br>0 : no critical event<br>1 : critical event occur<br>bit[11] : OAM internal critical event disable (OAM critical event according to register setting; reference to bit[10])<br>0 : Enable<br>1 : Disable<br>bit[12] : OAM external critical event disable (OAM critical event according to PIN)<br>0 : Enable<br>1 : Disable<br>bit[13] : OAM critical event indication once per second enable<br>0 : Disable<br>1 : Enable | R/W | 0x0000 |

| OAM Indirect Registers | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[14] : OAM Discovery check Unidirectional field<br>0 : Disable<br>1 : Enable<br>bit[15] : OAM Discovery check Remote LoopBack field<br>0 : Disable<br>1 : Enable | | |
| 0x01 | OAM remote read/write command | R/W | 0x0000 |
| 0x02 | OAM remote read/write data | R/W | 0x0000 |
| 0x03 ~ 0x07 | Reserved | | |
| 0x08 | OAM Status<br>bit[0] : link status<br>0 : unlink<br>1 : link<br>bit[1] : OAM mode<br>0 : passive mode<br>1 : active mode<br>bit[2] : local fault (refer to 0x17 for detail)<br>0 : local OAM no faults<br>1 : local OAM fault occurred<br>bit[3] : remote fault (refer to 0x17 for detail)<br>0 : remote OAM no faults<br>1 : remote OAM fault occurred<br>bit[5:4] : Reserved<br>bit[6] : active LoopBack test packets send out<br>0 : test packets not send out<br>1 : test packets send out<br>bit[7] : active LoopBack test result<br>0 : fail<br>1 : ok<br>bit[9:8] : Reserved<br>bit[10] : OAM remote read/write OUI check result (if 0x00 bit[6] and bit[7] are enable)<br>0 : mismatch<br>1 : match<br>bit[13:11] : OAM Discovery state<br>bit[15:14] : OAM Transmit state | RO | 0x0000 |
| 0x09 | OAM local flag<br>bit[15:0] : OAM local flag | RO | 0x0000 |
| 0x0A | OAM local Information OAMPDU revision<br>bit[15:0] : OAM local Information OAMPDU revision | RO | 0x0000 |
| 0x0B | OAM local Information OAMPDU state and configuration<br>bit[7:0] : OAM local Information OAMPDU configuration<br>bit[15:8] : OAM local Information OAMPDU state | RO | 0x0000 |

| OAM Indirect Registers | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x0C | OAM local Information OAMPDU OUI[15:0]<br>bit[15:0] : OAM local Information OAMPDU OUI[15:0] | RO | 0x0000 |
| 0x0D | OAM local Information OAMPDU OUI[23:16]<br>bit[7:0] : OAM local Information OAMPDU OUI[23:16] | RO | 0x00 |
| 0x0E | OAM local Information OAMPDU VENDOR<br>bit[15:0] : OAM local Information OAMPDU VENDOR[15:0] | RO | 0x0000 |
| 0x0F | OAM local Information OAMPDU VENDOR<br>bit[15:0] : OAM local Information OAMPDU VENDOR[31:16] | RO | 0x0000 |
| 0x10 | OAM otifi flag<br>bit[15:0] : OAM remote flag | RO | 0x0000 |
| 0x11 | OAM remote Information OAMPDU revision<br>bit[15:0] : OAM remote Information OAMPDU revision | RO | 0x0000 |
| 0x12 | OAM remote Information OAMPDU state and configuration<br>bit[7:0] : OAM remote Information OAMPDU configuration<br>bit[15:8] : OAM remote Information OAMPDU state | RO | 0x0000 |
| 0x13 | OAM remote Information OAMPDU OUI[15:0]<br>bit[15:0] : OAM remote Information OAMPDU OUI[15:0] | RO | 0x0000 |
| 0x14 | OAM remote Information OAMPDU OUI[23:16]<br>bit[7:0] : OAM remote Information OAMPDU OUI[23:16] | RO | 0x00 |
| 0x15 | OAM remote Information OAMPDU VENDOR<br>bit[15:0] : OAM remote Information OAMPDU VENDOR[15:0] | RO | 0x0000 |
| 0x16 | OAM remote Information OAMPDU VENDOR<br>bit[15:0] : OAM remote Information OAMPDU VENDOR[31:16] | RO | 0x0000 |
| 0x17 | OAM local/remote fault record(clear by indirect register 0x00 bit[3] reset)<br>bit[0] : OAM local link fault<br>bit[1] : OAM local dying gasp<br>bit[2] : OAM local critical event<br>bit[7:3] : Reserved<br>bit[8] : OAM remote link fault<br>bit[9] : OAM remote dying gasp<br>bit[10] : OAM remote critical event | RO | 0x0000 |
| 0x18<br>~<br>0x1F | Reserved | | |

### 6.9 RxDMA Control Register

| RxDMA Registers : Page 0x06 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 ~ 0x0C | Reserved | | |
| 0x0D | TCP/UDP destination port filter method<br>bit[1:0] : port 1 “Forwarding WAN/LAN / Packet drop” setting<br>00 : Disable Forwarding WAN/LAN& Packet drop<br>01 : Packet Drop enable (TX/RX ref 0x13 ~ 0x16)<br>10 : Forwarding to LAN port only (ref 0x11~0x12)<br>11 : Forwarding to WAN port only (ref 0x11 ~0x12)<br><br>bit[3:2] : port 2 “Forwarding WAN/LAN / Packet drop” setting<br>bit[5:4] : port 3 “Forwarding WAN/LAN / Packet drop” setting<br>bit[7:6] : port 4 “Forwarding WAN/LAN / Packet drop” setting<br>bit[9:8] : port 5 “Forwarding WAN/LAN / Packet drop” setting<br>bit[11:10]: port 6 “Forwarding WAN/LAN / Packet drop” setting<br>bit[13:12]: port 7 “Forwarding WAN/LAN / Packet drop” setting<br>bit[15:14]: port 8 “Forwarding WAN/LAN / Packet drop” setting | R/W | 0x0000 |
| 0x0E ~ 0x0F | Reserved | | |
| 0x10 | TCP/UDP destination port filter method<br>bit[1:0] : port 9 “Forwarding WAN/LAN / Packet drop” setting<br>bit[7:2] : Reserved<br>bit[9:8] : port 10“Forwarding WAN/LAN / Packet drop” setting<br><br>**Drop/ Filter rule**: ( ref below table )<br>(1) “Forwarding WANLAN / Packet drop setting = 10 or 11<br>this port only reference forwarding to WAN/LAN port only setting.<br>(2) “Forwarding WAN/LAN / Packet drop setting = 01<br>(i) this port is Drop enable and look at “drop act on” setting<br>(a) if act on RX , then all packet RX at this port is dropped<br>(b) if act on TX , then all packet from other port<br>TX out this port will be dropped<br>(3) “Forwarding WAN/LAN / Packet drop setting = 00<br>Filter & Drop function at this port is disabled | R/W | 0x000 |
| 0x11 | WAN Port setting for TCP/UDP destination port<br>bit[7:0] : WAN port setting (port 1 ~ port 8)<br>0: set as LAN 1: set as WAN | R/W | 0x0000 |

| RxDMA Registers : Page 0x06 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x12 | WAN Port setting for TCP/UDP destination port<br>bit[12][8] : WAN port setting (port 9 ~ port 10)<br>0: set as LAN 1: set as WAN | R/W | 0x0000 |
| 0x13 | TCP/UDP destination port filter – Drop act on TX/RX setting<br>bit[1:0] : port 1 drop act<br>00: Rx (all matched port 1 Rx packet drop)<br>01: Tx (all matched packet Tx to port 1 drop)<br>1x: Rx+Tx (packet RX/TX to port 1 drop)<br>bit[3:2] : port 2 drop act<br>bit[5:4] : port 3 drop act<br>bit[7:6] : port 4 drop act<br>bit[9:8] : port 5 drop act<br>bit[11:10] : port 6 drop act<br>bit[13:12] : port 7 drop act<br>bit[15:14] : port 8 drop act | R/W | 0x0000 |
| 0x14<br>0x15 | Reserved | | |
| 0x16 | TCP/UDP destination port filter – Drop act on TX/RX setting<br>bit[1:0] : port 9 drop act<br>bit[7:2] : Reserved<br>bit[9:8] : port 10 drop act | R/W | 0x000 |

Host / Client Forwarding WAN / LAN and Packet drop setting Condition

| Host / Client | | | Client Port setting | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Client is LAN Port | | | | | | Client is WAN Port | | | | | |
| | | | 00 | 01 Drop | | | 10 | 11 | 00 | 01 Drop | | | 10 | 11 |
| | | | Disable | RX | RX+TX | TX | LAN | WAN | Disable | RX | RX+TX | TX | LAN | WAN |
| Host port setting | 00 Disable | | V | V | X | X | V | V | V | V | X | X | V | V |
| | 01 Drop | Act on RX | X | | | | | | X | | | | | |
| | | Act on TX+RX | | | | | | | | | | | | |
| | | Act on TX | V | V | X | X | V | V | V | V | X | X | V | V |
| | 10 FWD LAN | | V | V | X | X | V | V | X | X | X | X | X | X |
| | 11 FWD WAN | | X | X | X | X | X | X | V | V | X | X | V | V |

(1) V : PASS

(2) X : Drop

(3) Host port
Switch Port to receive packet

(4) Client port
Switch Port to transmit packet

| RxDMA Registers : Page 0x06 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x17 | RX Priority queuing decision setting (1 bit/ port) port 1 ~ port 8<br>bit[7:0] : priority to Q based on ACL DSCP selected remark QID<br>0: Disable 1 : Enable<br><br>Priority remapping :<br>TX priority remapping (PG08.0xB2~0xB3 > PG08.0x14 ~ 0x4D) ><br>RX priority remapping (PG06.0x24~0x5D ) ><br>ACL DSCP remark addr / VLAN Table Remark pri (PG06.0x17 ~ 0x1A ) | R/W | 0x0000 |
| 0x18 | RX Priority queuing decision setting (1 bit/ port) port 9 ~ port 10<br>bit[12][8] : priority to Q based on ACL DSCP selected remark QID<br>0: Disable 1 : Enable | R/W | 0x0000 |
| 0x19 | RX Priority queuing decision setting (1 bit/ port) port 1 ~ port 8<br>bit[7:0] : priority to Q based on VLAN-table remarked tag priority<br>0: Disable 1: Enable | R/W | 0x0000 |
| 0x1A | RX Priority queuing decision setting (1 bit/ port) port 9 ~ port 10<br>bit[12][8] : priority to Q based on VLAN-table remarked tag priority<br>0: Disable 1: Enable<br>bit[14] : 0 : disable 1: enable (priority to Q based on ACL/VLAN)<br>bit[15] : ACL DSCP pri / VLAN-table Tag pri criteria order<br>0 : ACL DSCP pri > VLAN-table tag pri<br>1 : VLAN-table tag pr > ACL DSCP pri<br>(Note: IF DSCP pri / Tag pri remark are both enabled<br>this reg only affect rx packet to which one queue) | R/W | 0x0000 |
| 0x1B | (Mac priority/ VLAN-table remarked tag priority) to DSCP auto remarking (port 1 ~ port 8)<br>bit[7:0] : mac_priority to DSCP auto remarking<br>0: Disable 1: Enable | R/W | 0x0000 |
| 0x1C | (Mac priority/ VLAN-table remarked tag priority) to DSCP auto remarking (port 9 ~ port 10)<br>bit[12][8] : mac_priority to DSCP auto remarking<br>0: Disable 1: Enable | R/W | 0x0000 |
| 0x1D | (Mac priority/ VLAN-table remarked tag priority) to DSCP auto remarking ( port 1 ~ port 8)<br>bit[7:0] : VLAN-table remarked tag priority to DSCP auto remarking | R/W | 0x0000 |
| 0x1E | (Mac priority/ VLAN-table remarked tag priority) to DSCP auto remarking ( port 9 ~ port 10)<br>bit[12][8] : VLAN-table remarked tag priority to DSCP auto remarking<br><br>bit[15] : mac_pri / remarked tag pri criteria order<br>0 : mac pri > remarked tag pri<br>1 : remarked pri > tag pri<br>(Criteria ACL DSCP pri > above description)<br>(If QinQ VID Sel is enable , ACL DSCP remark will be disabled) | R/W | 0x0000 |
| 0x1F | TCP Flag attribute priority:<br>bit[0] : <= 256 Bytes TCP SYN highest priority enable<br>bit[1] : <= 256 Bytes TCP SYN/ACK highest priority enable<br>bit[2] : <= 256 Bytes TCP ACK highest priority enable | R/W | 0x0 |

| RxDMA Registers : Page 0x06 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x20 | WRED Drop packet setting:<br>bit[0] : WRED drop act on TCP packet only (1: tcp0: all)<br>bit[1] : WRED drop act on TCP exclude TCP SYN <= 256 B<br>bit[2] : WRED drop act on TCP exclude TCP SYN/ACK <= 256 B<br>bit[3] : WRED drop act on TCP exclude TCP ACK <= 256 B<br>bit[8] : sFlow packet force TX out<br>(0: normal drop 1: force tx, regardless any drop)<br>bit[9] : sFlow packet port map to cpu packet length cutoff<br>(0: Disable 1: Enable) | R/W | 0x009 |
| 0x21 | Out Queue Pause / Drop to RX setting<br>bit[7:0] : port 1 ~ port 8 out queue threshold to RX pause/drop disable<br>0: Enable 1: Disable (1bit/port) | R/W | 0x0000 |
| 0x22 | Out Queue Rx Pause / Drop to RX setting<br>bit[12][8] : port 9 ~ port 10 out queue threshold to RX pause/drop disable | R/W | 0x0000 |
| 0x23 | Reserved | | |
| 0x24 | port 1 RX Priority remap data<br>bit[2:0] : Priority 0 remap<br>bit[5:3] : Priority 1 remap<br>bit[8:6] : Priority 2 remap<br>bit[11:9] : Priority 3 remap<br>bit[14:12] : Priority 4 remap | R/W | 0x4688 |
| 0x25 | port 1 RX Priority remap data<br>bit[2:0] : Priority 5 remap<br>bit[5:3] : Priority 6 remap<br>bit[8:6] : Priority 7 remap<br>Priority remapping :<br>TX priority remapping (PG08.0xB2~0xB3 > PG08.0x14 ~ 0x4D) ><br>RX priority remapping (PG06.0x24~0x5D ) ><br>ACL DSCP remark addr / VLAN Table Remark pri (PG06.0x17 ~ 0x1A ) | R/W | 0x1F5 |
| 0x26 | port 2 RX Priority remap data<br>bit[2:0] : Priority 0 remap<br>bit[5:3] : Priority 1 remap<br>bit[8:6] : Priority 2 remap<br>bit[11:9] : Priority 3 remap<br>bit[14:12] : Priority 4 remap | R/W | 0x4688 |
| 0x27 | port 2 RX Priority remap data<br>bit[2:0] : Priority 5 remap<br>bit[5:3] : Priority 6 remap<br>bit[8:6] : Priority 7 remap | R/W | 0x1F5 |
| 0x28 ~ 0x33 | port 3 ~ port 8 RX priority remap data | R/W | 0x4688<br>0x1F5 |
| 0x34 ~ 0x53 | Reserved | | |

| RxDMA Registers : Page 0x06 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x54<br>0x55 | port 9 RX priority remap data | R/W | 0x4688<br>0x1F5 |
| 0x56<br>~<br>0x5B | Reserved | | |
| 0x5C<br>0x5D | port 10 RX Priority remap data | R/W | 0x4688<br>0x1F5 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.10 TxDMA Control Register

<table>
<tr><th colspan="4">TxDMA Registers : Page 0x07</th></tr>
<tr><th>Reg Addr.</th><th>Register Description</th><th>R/W</th><th>Default value</th></tr>
<tr><td>0x01</td><td>QinQ double tag remove service tag operation configuration<br>bit[7:0] : output frame remove service tag (port 1 ~ port 8)<br>0 : Disable 1 : remove service tag to output frame</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x02</td><td>QinQ double tag remove service tag operation configuration<br>bit[12][8]: output frame remove service tag (port 9 ~ port 10)<br>0 : Disable 1 : remove service tag to output frame</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x03</td><td>QinQ double tag add service tag operation configuration<br>bit[7:0] : output frame add service tag ( port 1 ~ port 8)<br>0 : Disable 1 : add service tag to output frame</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x04</td><td>QinQ double tag add service tag operation configuration<br>bit[12][8] : output frame add service tag ( port 9 ~ port 10)<br>0 : Disable 1 : add service tag to output frame<br>(QinQ detect enable set from 0x05 ~ 0x06)
<table>
<tr><td></td><td></td><td colspan="4">0x05~0x06 detect enable</td><td colspan="4">0x05~0x06 detect disable</td></tr>
<tr><td>Qing (add,rmv)</td><td></td><td>00</td><td>01</td><td>10</td><td>11</td><td>00</td><td>01</td><td>10</td><td>11</td></tr>
<tr><td rowspan="2">Service Tag rx</td><td>0</td><td>x</td><td>x</td><td>add</td><td>x</td><td>x</td><td>x</td><td>add</td><td>x</td></tr>
<tr><td>1</td><td>x</td><td>rmv</td><td>mdf</td><td>x</td><td>x</td><td>x</td><td>add</td><td>x</td></tr>
</table></td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x05</td><td>Detect RX QinQ double tag enable for port 1 ~ port 8<br>bit[7:0] : 0 : Disable 1 : Enable</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x06</td><td>Detect RX QinQ double tag enable for port 9 ~ port 10<br>bit[12][8] : 0 : Disable 1 : Enable</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x07</td><td>802.1Q tagged PVID use TX port enable for port 1 ~ port 8<br>bit[7:0] : 0 : Disable 1 : Enable<br>Criteria priority (0x75 ~ 0x78):<br>00: TX PVID (0x07~0x08) > ACL VID > QID register 0x70 ~0x74 > RX PVID<br>01: TX PVID (0x07~0x08) > ACL VID > RX PVID<br>10: TX PVID (0x07~0x08) > ACL VID > Vlan table pri remark > RX PVID<br>11: TX PVID (0x07~0x08) > ACL VID > Vlan table pri remark > QID register 0x70 ~0x74 > RX PVID<br>(QID register 0x70 ~ 0x74 only relative to { pri, cfi } , VID index will use RXPVID)</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x08</td><td>802.1Q tagged PVID use TX port enable for port 9 ~ port 10<br>bit[12][8] : 0 : Disable 1 : Enable</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x09</td><td>QinQ double tag type-length definition</td><td>R/W</td><td>0x88A8</td></tr>
<tr><td>0x0A</td><td>QinQ double tag {PCP, DEI, VID} definition0 – Data (S-Tag VID)<br>by ACL rule / register setting<br>Used to insert into/modify frame which should add/modify a double tag</td><td>R/W</td><td>0x0000</td></tr>
<tr><td>0x0B ~ 0x49</td><td>QinQ double tag {PCP, DEI, VID} definition1~63 – Data (S-Tag VID)<br>by ACL rule / register setting<br>Used to insert into frame which should add a double tag</td><td>R/W</td><td>0x0000</td></tr>
</table>

| TxDMA Registers : Page 0x07 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x4A | QinQ double tag VID r/w address and select method port 1<br>bit[5:0] : port 1 S-TAG VID r/w register select address<br>bit[11:8] : port 1 S-TAG VID r/w address select method<br>0000 : address = mg_stag_vid_adr_p1[5:0]<br>0001 : address = acl_dvid_sel[5:0], 6'd0 means no ACL act<br>( if ACL entry is not hit , mg_stag_vid_adr_p1[5:0] will be used)<br>0010 : address = 802.1Q-VID[11:6] (C-TAG VID[11:6])<br>0011 : address = 802.1Q-VID[5:0] (C-TAG VID[5:0])<br>0100 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[11:8] (C-TAG VID[11:8])<br>0101 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[7:4] } (C-TAG VID[7:4])<br>0110 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[3:0]} (C-TAG VID[3:0])<br>( use RX 1Q-VID before TX remove/modify 1Q-VID,<br>If RX packet is non-1Q-VID , then DVID0 will be used)<br>1010 : address = 802.1Q-VID[11:6] (C-TAG VID[11:6])<br>1011 : address = 802.1Q-VID[5:0] (C-TAG VID[5:0])<br>1100 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[11:8] (C-TAG VID[11:8])<br>1101 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[7:4] (C-TAG VID[7:4])<br>1110 : address = {mg_stag_vid_adr_p1[1:0], 802.1Q-VID[3:0] (C-TAG VID[3:0])<br>( Insert / modify : use TX 1Q-VID , come from ACL result)<br>( Remove : use RX port's PVID )<br>bit[12] : QinQ double tag {PCP, DEI} field keep original packet<br>During qinq tag modification. (0: modified 1: keep org) | R/W | 0x0000 |
| 0x4B | QinQ double tag VID r/w address and select method port 2<br>bit[5:0] : port 2 S-TAG VID r/w Register select address<br>bit[10:8] : port 2 S-TAG VID r/w address select method<br>bit[12] : QinQ double tag {PCP, DEI} field keep original packet<br>During qinq tag modification. (0: modified 1: keep org) | R/W | 0x0000 |
| 0x4C ~ 0x51 | QinQ double tag VID r/w address and select method port 3 ~ port 8<br>bit[5:0] : port 3 ~ port 8 S-TAG VID r/w Register select address<br>bit[10:8] : port 3 ~ port 8 S-TAG VID r/w address select method<br>bit[12] : QinQ double tag {PCP, DEI} field keep original packet<br>During qinq tag modification. (0: modified 1: keep org) | R/W | 0x0000 |
| 0x52 ~ 0x61 | Reserved | | |
| 0x62 | QinQ double tag VID r/w address and select method port 9<br>bit[5:0] : port 2 S-TAG VID r/w Register select address<br>bit[10:8] : port 2 S-TAG VID r/w address select method<br>bit[12] : QinQ double tag {PCP, DEI} field keep original packet<br>During qinq tag modification. (0: modified 1: keep org) | R/W | 0x0000 |

| TxDMA Registers : Page 0x07 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x63 ~ 0x65 | Reserved | | |
| 0x66 | QinQ double tag VID r/w address and select method port 10<br>bit[5:0] : port 10 S-TAG VID r/w Register select address<br>bit[10:8] : port 10 S-TAG VID r/w address select method<br>bit[12] : QinQ double tag {PCP, DEI} field keep original packet During qinq tag modification. (0: modified 1: keep org) | R/W | 0x0000 |
| 0x67 | IPv4/v6 DSCP/IPP TOS remarking (ACL Select 0, 1) / (QID = 0,1)<br>bit[5:0] : ACL select 0 / Queue 0 , remarking Value<br>bit[6] : ACL select 0 / Queue 0 DSCP[2:0]/ IPP remarking 0 : Enable 1: Disable<br>bit[7] : ACL select 0 / Queue 0 DSCP[5:3]/ TOS remarking 0 : Enable 1: Disable<br>bit[13:8] : ACL select 1 / Queue 1 , remarking Value<br>bit[14] : ACL select 1 / Queue 1 DSCP[2:0]/ IPP remarking 0 : Enable 1: Disable<br>bit[15] : ACL select 1 / Queue 1 DSCP[5:3]/ TOS remarking 0 : Enable 1: Disable | R/W | 0x0000 |
| 0x68 | IPv4/v6 DSCP/IPP TOS remarking (ACL Select 2, 3) / (QID = 2,3)<br>bit[5:0] : ACL select 2 / Queue 2 , remarking Value<br>bit[6] : ACL select 2 / Queue 2 DSCP[2:0]/ IPP remarking<br>bit[7] : ACL select 2 / Queue 2 DSCP[5:3]/ TOS remarking<br>bit[13:8] : ACL select 3 / Queue 3 , remarking Value<br>bit[14] : ACL select 3 / Queue 3 DSCP[2:0]/ IPP remarking<br>bit[15] : ACL select 3 / Queue 3 DSCP[5:3]/ TOS remarking | R/W | 0x0000 |
| 0x69 | IPv4/v6 DSCP/IPP TOS remarking (ACL Select 4, 5) / (QID = 4,5)<br>bit[5:0] : ACL select 4 / Queue 4 , remarking Value<br>bit[6] : ACL select 4 / Queue 4 DSCP[2:0]/ IPP remarking<br>bit[7] : ACL select 4 / Queue 4 DSCP[5:3]/ TOS remarking<br>bit[13:8] : ACL select 5 / Queue 5 , remarking Value<br>bit[14] : ACL select 5 / Queue 5 DSCP[2:0]/ IPP remarking<br>bit[15] : ACL select 5 / Queue 5 DSCP[5:3]/ TOS remarking | R/W | 0x0000 |
| 0x6A | IPv4/v6 DSCP/IPP TOS remarking (ACL Select 6, 7) / (QID = 6,7)<br>bit[5:0] : ACL select 6 / Queue 6 , remarking Value.<br>bit[6] : ACL select 6 / Queue 6 DSCP[2:0]/ IPP remarking<br>bit[7] : ACL select 6 / Queue 6 DSCP[5:3]/ TOS remarking<br>bit[13:8] : ACL select 7 / Queue 7 , remarking Value<br>bit[14] : ACL select 7 / Queue 7 DSCP[2:0]/ IPP remarking<br>bit[15] : ACL select 7 / Queue 7 DSCP[5:3]/ TOS remarking | R/W | 0x0000 |
| 0x6B | IPv4/v6 DSCP/IPP TOS Setup command (0x6E/0x6F QID rule per queue setting, per queue default is disabled)<br>bit[7:0] : queue enable for Q7 ~ Q0<br>bit[13:8] : Read/ Write dscp enable data From/to Pxx<br>bit[15] : 0: Read 1: Write | R/W | 0x0000 |

| TxDMA Registers : Page 0x07 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x6C | IPv4/v6 DSCP/IPP TOS remarking Method (from ACL table)<br>bit[7:0] : ACL ruled Ipv4 DSCP/IPP TOS remarking (1 bit / port)<br>(ACL has higher priority than queue ID ) | R/W | 0xFFFF |
| 0x6D | IPv4/v6 DSCP/IPP TOS remarking Method (from ACL table)<br>bit[12][8] : ACL ruled Ipv4 DSCP/IPP TOS remarking (1 bit / port)<br>(ACL has higher priority than queue ID ) | R/W | 0x1FFF |
| 0x6E | IPv4/v6 DSCP/IPP TOS remarking Method<br>(from QID)(0x6B set per queue based)<br>bit[7:0] : queue ID rules Ipv4 DSCP/IPP TOS remarking (1 bit / port)<br>(0x6C, 0x6D ACL has higher priority than 0x6E 0x6F queue ID ) | R/W | 0xFFFF |
| 0x6F | IPv4/v6 DSCP/IPP TOS remarking Method<br>(from QID) (0x6B set per queue based)<br>bit[12][8] : queue ID rules Ipv4 DSCP/IPP TOS remarking (1 bit / port)<br>(0x6C, 0x6D ACL has higher priority than 0x6E 0x6F queue ID ) | R/W | 0x1FFF |
| 0x70 | 802.1Q TAG {PCP} remarking (Q1 , Q0)<br>bit[0] : Queue 0 original CFI value 0 remarking<br>bit[1] : Queue 0 original CFI value 1 remarking<br>bit[6:4] : Queue 0 , 802.1Q priority remarking value<br>bit[8] : Queue 1 original CFI value 0 remarking<br>bit[9] : Queue 1 original CFI value 1 remarking<br>bit[14:12] : Queue 1 , 802.1Q priority remarking value<br>CFI remarking will be regardless of 802.1Q TAG enable/disable | R/W | 0x1202 |
| 0x71 | 802.1Q TAG {PCP} remarking (Q3 , Q2)<br>bit[0] : Queue 2 original CFI value 0 remarking<br>bit[1] : Queue 2 original CFI value 1 remarking<br>bit[6:4] : Queue 2 , 802.1Q priority remarking value<br>bit[8] : Queue 3 original CFI value 0 remarking<br>bit[9] : Queue 3 original CFI value 1 remarking<br>bit[14:12] : Queue 3 , 802.1Q priority remarking value<br>CFI remarking will be regardless of 802.1Q TAG enable/disable | R/W | 0x3222 |
| 0x72 | 802.1Q TAG {PCP} remarking (Q5 , Q4)<br>bit[0] : Queue 4 original CFI value 0 remarking<br>bit[1] : Queue 4 original CFI value 1 remarking<br>bit[6:4] : Queue 4 , 802.1Q priority remarking value<br>bit[8] : Queue 5 original CFI value 0 remarking<br>bit[9] : Queue 5 original CFI value 1 remarking<br>bit[14:12] : Queue 5 , 802.1Q priority remarking value<br>CFI remarking will be regardless of 802.1Q TAG enable/disable | R/W | 0x5242 |
| 0x73 | 802.1Q TAG {PCP} remarking (Q7 , Q6)<br>bit[0] : Queue 6 original CFI value 0 remarking<br>bit[1] : Queue 6 original CFI value 1 remarking<br>bit[6:4] : Queue 6 , 802.1Q priority remarking value<br>bit[8] : Queue 7 original CFI value 0 remarking<br>bit[9] : Queue 7 original CFI value 1 remarking<br>bit[14:12] : Queue 7 , 802.1Q priority remarking value<br>CFI remarking will be regardless of 802.1Q TAG enable/disable | R/W | 0x7262 |

| TxDMA Registers : Page 0x07 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x74 | 802.1Q TAG {PCP} remarking Setup command (qid rule)<br>bit[7:0] : queue enable for Q7 ~ Q0<br>bit[13:8] : Read/Write remarking enable data to From/to Pxx<br>bit[15] : 0: Read 1: Write | R/W | 0x0000 |
| 0x75 | 802.1Q TAG {PCP, CFI} remarking criteria ( port 1 ~ port 8 , 2bit / port)<br>bit[1:0] : port 1 remarking criteria<br>00: TX PVID (0x07~0x08) > ACL VID > QID register 0x70 ~0x74 > RX PVID<br>01: TX PVID (0x07~0x08) > ACL VID > RX PVID<br>10: TX PVID (0x07~0x08) > ACL VID > VLAN table pri remark > RX PVID<br>11: TX PVID (0x07~0x08) > ACL VID > VLAN table pri remark > QID register 0x70 ~0x74 > RX PVID<br>(QID register 0x70 ~ 0x74 only relative to { pri, cfi } , VID index will use RXPVID)<br>bit[15:2] : port 2 ~ port 8 remarking criteria | RW | 0xAAAA |
| 0x76<br>0x77 | Reserved | | |
| 0x78 | 802.1Q TAG {PCP, CFI} remarking criteria ( port 9 ~ port 10 , 2bit / port)<br>bit[1:0] : Port 9 remarking criteria<br>bit[7:2]: Reserved<br>bit[9:8]: port 10 remarking criteria | R/W | 0x2AA |
| 0x79 | TX (802.1Q TAG) type-length field definition<br>bit[15:0] : CTAG T/L field ( Add / MDF ) | R/W | 0x8100 |
| 0x7A | Port 10 Special Tag setting<br>bit[3:0] : sFlow packet length cutoff setting<br>4'd0 : 64 Bytes<br>4'd1 : 256 Bytes<br>4'd2 : 512 Bytes<br>4'd3 : 768 Bytes<br>4'd4 : 1024 Bytes<br>4'd5 : 1280 Bytes<br>4'd6 : 1536 Bytes<br>4'd7 : 1792 Bytes<br>4'd8 : 2048 Bytes<br>4'd9 : 2304 Bytes<br>4'd10 : 2560 Bytes<br>4'd11 : 2816 Bytes<br>4'd12 : 3072 Bytes<br>4'd13 : 3328 Bytes<br>4'd14/15: Disable length cutoff<br>Note: (1) If original packet length < length cutoff setting ,packet length will not modify<br>bit[4] : sFlow information add to special tag ( 28 Bytes)<br>1: Enable 0 : Disable<br>bit[5] : PTP stamp information add to special tag (8 Bytes)<br>1: Enable 0 : Disable | R/W | 0x3F |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

## 6.11 Output Queue Control Register

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | Port Group A Out queue schedule mode / weight selection<br>bit[2:0] : Schedule mode<br>0 : First come First schedule<br>1 : WRR/WFQ/BW/TWRR<br>2 : SPx1 + WRR/WFQ/BW/TWRRx7<br>3 : SPx2 + WRR/WFQ/BW/TWRRx6<br>4 : SPx4 + WRR/WFQ/BW/TWRRx4<br>5 : SPx8<br>6 : LLQx1 + WFQ/BW/TWRRx7 (LLQ in WRR will switch to SP)<br>7 : LLQx2 + WFQ/BW/TWRRx6 (LLQ in WRR will switch to SP)<br>bit[4:3] : BW throttle period sel / TWRR tickle unit<br>00 : 64 Kbit/s / (800 us/8000 us/80000 us @ 1G/100M/10M)<br>01 : 1 Mbit/s / (49.92 us/499.2 us/4992 us @ 1G/100M/10M)<br>10 : 2 Mbit/s / (24.32 us/243.2 us/2432 us @ 1G/100M/10M)<br>11 : 4 Mbit/s / (12.8 us/128 us/1280 us @1G/100M/10M)<br>bit[6:5] : Queue method( 00: WRR 01: BW<br>10: WFQ 11: TWRR)<br>bit[7] : Delay Bound mode (Q7/Q6/Q5/Q4/Q3/Q2/Q1 )<br>(Q0 queue out only when Q7~Q1 is empty )<br>must set bit[6:5] = 10/01, at the same time<br><br>bit[8] : Q0 BE(best effort) queue disable (only bit[7] = 1)<br>(Q0 is also delay bound queue if disable BE)<br><br>bit[9] : WRED Drop enable (must also set 0x53/0x54)<br>bit[10] : Delay Bound/LLQ Pause-Backpressure enable<br>(will enable automatically if 0x01[6:5]=01/10 & 0x01[7] =1)<br>bit[11] : Priority remap enable (0: CPU 1: Std)<br>bit[14:12] : Reserved<br>bit[15] : Queue schedule ratio value “0” definition<br>BW/WFQ/TWRR (0: eq value1 1: Stop & hold packet) | R/W | 0x0100 |
| 0x02 | Port Group A Out queue schedule ratio 0<br>bit[7:0] : Q0 weight / bandwidth allocate /<br>bit[15:8] : Q1 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |
| 0x03 | Port Group A Out queue schedule ratio 1<br>bit[7:0] : Q2 weight / bandwidth allocate /<br>bit[15:8] : Q3 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |
| 0x04 | Port Group A Out queue schedule ratio 2<br>bit[7:0] : Q4 weight / bandwidth allocate /<br>bit[15:8] : Q5 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x05 | Port Group A Out queue schedule ratio 3<br>bit[7:0] : Q6 weight / bandwidth allocate /<br>bit[15:8] : Q7 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |
| | Note 1: if 0x0E ~ 0x13 SRP parameter is setup with value > 0 ,<br>Q7 BW = mg_oq_lan_ratio3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_lan_ratio3[7:0] x 64 Bytes / slot time<br>Note 2: The LAN port max BW (if port SRP parameter > 0)<br>Q7 BW = mg_oq_lan_max_bw3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_lan_max_bw3[7:0] x 64 Bytes / slot time<br>Note 3: If Delay bound mode is enable , Per Queue latency limit<br>= ratio x {16us, 64us, 256us, 1024us}(ref 0x69[10:9] ) | | |
| 0x06 | Port Group A Out queue 0 latency threshold<br>bit[7:0] : Q0 latency threshold<br>bit[15:8] : Q1 latency threshold | R/W | 0x80FF |
| 0x07 | Port Group A Out queue 1 latency threshold<br>bit[7:0] : Q2 latency threshold<br>bit[15:8] : Q3 latency threshold | R/W | 0x2040 |
| 0x08 | Port Group A Out queue 2 latency threshold<br>bit[7:0] : Q4 latency threshold<br>bit[15:8] : Q5 latency threshold | R/W | 0x0810 |
| 0x09 | Port Group A Out queue 3 latency threshold<br>bit[7:0] : Q6 latency threshold<br>bit[15:8] : Q7 latency threshold | R/W | 0x0204 |
| 0x0A | Port Group A Out queue Max Bandwidth<br>bit[7:0] : Q0 max bandwidth<br>bit[15:8] : Q1 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |
| 0x0B | Port Group A Out queue Max Bandwidth<br>bit[7:0] : Q2 max bandwidth<br>bit[15:8] : Q3 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |
| 0x0C | Port Group A Out queue Max Bandwidth<br>bit[7:0] : Q4 max bandwidth<br>bit[15:8] : Q5 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |
| 0x0D | Port Group A Out queue Max Bandwidth<br>bit[7:0] : Q6 max bandwidth<br>bit[15:8] : Q7 max bandwidth<br>(value =0 : unlimited)<br>Ethernet AV setup:<br>The LAN port max BW (if port 0x0E ~0x13 SRP parameter > 0)<br>Q7 BW = mg_oq_lan_max_bw3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_lan_max_bw3[7:0] x 64 Bytes / slot time | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x0E | Ethernet AV SRP BW Credit-shaper parameter setup<br>bit[2:0] : port 01<br>bit[5:3] : port 02<br>bit[8:6] : port 03<br>bit[11:9] : port 04<br>bit[14:12] : port 05<br>SRP parameter (sche mode must : WFQ / BW /TWRR)<br>( 250 us *n : reg0x66[13:11] )<br>000 : Disable<br>001 : Q7.125 us<br>010 : Q7.250 us * n<br>011 : Q7.125 us & Q6.125 us<br>100 : Q7.125 us & Q6.250 us * n<br>101 : Q7.250 us *n & Q6.250 us * n<br><br>(Must set: Group A<br>0x0D Q7/Q6 max bandwidth → As MAX Threshold<br>0x09 Q7/Q6 latency threshold → As Idle Slope<br>0x05 Q7/Q6 bandwidth allocate → As min BW threshold<br><br>Group B<br>0x84 Q7/Q6 max bandwidth → As MAX Threshold<br>0x80 Q7/Q6 latency threshold → As Idle Slope<br>0x7C Q7/Q6 bandwidth allocate → As min BW threshold<br><br>For Credit Shaped BW control, MAX must > min BW Threshold, | R/W | 0x0000 |
| 0x0F | Ethernet AV SRP BW Credit-shaper parameter setup<br>bit[2:0] : port 6<br>bit[5:3] : port 7<br>bit[8:6] : port 8<br>bit[14:9] : Reserved | R/W | 0x0000 |
| 0x10<br>0x11 | Reserved | | |
| 0x12 | Ethernet AV SRP BW Credit-shaper parameter setup<br>bit[11:0] : Reserved<br>bit[14:12] : port 9 | R/W | 0x0000 |
| 0x13 | Ethernet AV SRP BW Credit-shaper parameter setup<br>bit[8:0] : Reserved<br>bit[11:9] : port 10 | R/W | 0x000 |
| 0x14 | port 1 TX Priority remap data<br>bit[2:0] : Priority 0 remap<br>bit[5:3] : Priority 1 remap<br>bit[8:6] : Priority 2 remap<br>bit[11:9] : Priority 3 remap<br>bit[14:12] : Priority 4 remap | R/W | 0x4688 |

## Output Queue Register : Page 0x08

<table>
<tr><th>Reg Addr.</th><th>Register Description</th><th>R/W</th><th>Default value</th></tr>
<tr><td>0x15</td><td>port 1 TX Priority remap data<br>bit[2:0] : Priority 5 remap<br>bit[5:3] : Priority 6 remap<br>bit[8:6] : Priority 7 remap<br>Priority remapping :<br>TX priority remapping (PG08.0xB2~0xB3 > PG08.0x14 ~ 0x4D) ><br>RX priority remapping (PG06.0x24~0x5D ) ><br>ACL DSCP remark addr / VLAN Table Remark pri (PG06.0x17 ~ 0x1A )</td><td>R/W</td><td>0x1F5</td></tr>
<tr><td>0x16</td><td>port 2 TX Priority remap data<br>bit[2:0] : Priority 0 remap<br>bit[5:3] : Priority 1 remap<br>bit[8:6] : Priority 2 remap<br>bit[11:9] : Priority 3 remap<br>bit[14:12] : Priority 4 remap</td><td>R/W</td><td>0x4688</td></tr>
<tr><td>0x17</td><td>port 2 TX Priority remap data<br>bit[2:0] : Priority 5 remap<br>bit[5:3] : Priority 6 remap<br>bit[8:6] : Priority 7 remap</td><td>R/W</td><td>0x1F5</td></tr>
<tr><td>0x18<br>~<br>0x23</td><td>port 3 ~ port 8 TX priority remap data</td><td>R/W</td><td>0x4688<br>~<br>0x1F5</td></tr>
<tr><td>0x24<br>~<br>0x43</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>0x44<br>0x45</td><td>port 9 TX priority remap data</td><td>R/W</td><td>0x4688<br>0x1F5</td></tr>
<tr><td>0x46<br>~<br>0x4B</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>0x4C<br>0x4D</td><td>port 10 TX Priority remap data</td><td>R/W</td><td>0x4688<br>0x1F5</td></tr>
<tr><td>0x4E</td><td>Out queue Random Early Detect threshold<br>bit[3:0] : WRED threshold for ACT0 bit[7:4] : WRED threshold for ACT1<br>bit[11:8] : WRED threshold for ACT2<br>bit[15:12] : WRED threshold for ACT3
<table>
<tr><th>WRED Drop probability</th><th>Q7/Q6</th><th>Q5/Q4</th><th>Q3/Q2</th><th>Q1/Q0</th></tr>
<tr><td>oq_wred_cnt > ACT3</td><td>1/(0x52 thr)</td><td>1/(0x51 thr)</td><td>1/(0x50 thr)</td><td>1/(0x4F thr)</td></tr>
<tr><td>ACT2 <oq_wred_cnt < ACT3</td><td>x</td><td>1/(0x52 thr)</td><td>1/(0x51 thr)</td><td>1/(0x50 thr)</td></tr>
<tr><td>ACT1 <oq_wred_cnt < ACT2</td><td>x</td><td>x</td><td>1/(0x52 thr)</td><td>1/(0x51 thr)</td></tr>
<tr><td>ACT0<oq_wred_cnt < ACT1</td><td>x</td><td>x</td><td>x</td><td>1/(0x52 thr)</td></tr>
<tr><td>oq_wred_cnt < ACT0</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
</table></td><td>R/W</td><td>0x8421</td></tr>
</table>

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x4F | WRED markA weight threshold<br>bit[8:0] : mark weight threshold drop rate(1/packet) = 1/(threshold) | R/W | 0x001 |
| 0x50 | WRED markB weight threshold<br>bit[8:0] : mark weight threshold drop rate(1/packet) = 1/(threshold) | R/W | 0x002 |
| 0x51 | WRED markC weight threshold<br>bit[8:0] : mark weight threshold drop rate(1/packet) = 1/(threshold) | R/W | 0x003 |
| 0x52 | WRED markD weight threshold<br>bit[8:0] : mark weight threshold drop rate(1/packet) = 1/(threshold) | R/W | 0x004 |
| 0x53 | Out queue WRED enable<br>bit[7:0] : port 1 ~ port 8 Out Queue WRED ability<br>1: Enable 0 : Disable | R/W | 0x0000 |
| 0x54 | Out queue WRED enable<br>bit[12][8] : port 9 ~ port 10 Out Queue WRED ability<br>1: Enable 0 : Disable | R/W | 0x0000 |
| 0x55 | port 1 ~ port 2 queue based SBM (static bandwidth management / DBM (Dynamic bandwidth management) select<br>bit[7:0] : port 1 Q7~Q0 SBM /DBM select ( 1: SBM 0: DBM)<br>bit[15:8] : port 2 Q7~Q0 SBM /DBM select<br><br>must set mg_oq_sche_mode[6:5] = 01<br>(Out queue BW-mode )<br>**(DBM must work with 0x64~0x65 BW range set)**<br>**(DBM will be disabled internally if 0x64 ~ 0x65 TX port BW range is not set.)**<br><br>ex: throttle enable:<br>1: traffic stop when hit bandwidth limit threshold ( SBM)<br>0: Traffic between Bandwidth MAX , min (DBM) | | 0xFFFF |
| | Note1: Per queue bandwidth range control must set<br>PG08.0x01[6:5] = 2'b01 if belong to Port Group A<br>PG08.0x76[6:5] = 2'b01 if belong to Port Group B<br>PG08.0x55~0x63 (0: enable queue bw range control<br>1: only bandwidth limited)<br>For Port Group A , PG08.0x02~0x05 used as Bandwidth-min<br>PG08.0x0A~0x0D used as Bandwidth-MAX<br>For Port Group B , PG08.0x79~0x7C used as Bandwidth-min<br>PG08.0x81~0x84 used as Bandwidth-MAX<br><br>Note2: If MAX < min , min BW will not be assured. It only show BW limited at MAX value | | |
| 0x56 ~ 0x58 | port 3 ~ port 8 queue based SBM (static bandwidth management / DBM (Dynamic bandwidth management) select<br>bit[7:0] : Pn Q7~Q0 SBM /DBM select ( 1: SBM 0: DBM)<br>bit[15:8] : Pn+1 Q7~Q0 SBM /DBM select | R/W | 0xFFFF |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0X59<br>0X60 | Reserved | | |
| 0x61 | port 9 queue based SBM (static bandwidth management /<br>DBM (Dynamic bandwidth management) select<br>bit[7:0] : Pn Q7~Q0 SBM /DBM select ( 1: SBM 0: DBM)<br>bit[15:8] : Reserved | R/W | 0xFFFF |
| 0x62 | Reserved | | |
| 0x63 | port 10 queue based SBM (static bandwidth management /<br>DBM (Dynamic bandwidth management) select<br>bit[7:0] : Pn+1 Q7~Q0 SBM /DBM select ( 1: SBM 0: DBM)<br>bit[15:8] : Reserved | R/W | 0xFF |
| 0x64 | Queue DBM (Dynamic Bandwidth management ) enable<br>bit[7:0] : mg_oq_bw_range_en port 1 ~ port 8 | R/W | 0x0000 |
| 0x65 | Queue DBM (Dynamic Bandwidth management ) enable<br>bit[12][8] : mg_oq_bw_range_en port 9 ~ port 10 | R/W | 0x0000 |
| 0x66 | Out queue aging time<br>bit[7:0]: Out queue aging time =(1~2)* value* 100 ms<br>bit[8] : fast aging enable ( unit = 1.638 ms)<br>bit[10:9] : Out queue latency timing unit ( Delay Bound mode)<br>0: 16 us/ 64us / 1024 us (Giga/100/10)<br>1 : 64 us 2: 256 us 3: 1024 us<br>bit[13:11] : SRP macro slot period threshold (unit 250us)<br>000 : 250 us<br>001 : 500 us<br>010 : 1 ms<br>011 : 2 ms<br>100 : 4 ms<br>101/110/111 : 8 ms<br>bit[14]: mg_oq_latency_en<br>(rdma pause/drop on instantly when out queue port WRED pause )<br>bit[15] : mg_oq_bw_min_resol (ref PG08.0x66[15]) for queue<br>(0: Per Queue BW control normal period burst mode)<br>(1 & PG08.0xB1[7]=0: Per Queue BW control min burst mode)<br>(1 & PG08.0xB1[7]=1: Per Queue BW control leaky bucket mode)<br><br>Note: Token Bucket BW control Per queue can be implemented from bandwidth range enable and set PG08.0x66[15] = 1, PG08.0xB1[7]=1 (Max bandwidth will be the token size) | R/W | 0x8000 |
| 0x67 | port 1 ~ port 2 Out queue aging enable<br>bit[7:0] : port 1 Q7~Q0 queue aging enable<br>bit[15:8] : port 2 Q7~Q0 queue aging enable | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x68 ~ 0x6A | port 3 ~ port 8 Out queue aging enable<br>bit[7:0] : port xx Q7~Q0 queue aging enable<br>bit[15:8] : port xy Q7~Q0 queue aging enable | R/W | 0x0000 |
| 0x6B ~ 0x72 | Reserved | | |
| 0x73 | port 9 Out queue aging enable<br>bit[7:0] : port 9 Q7~Q0 queue aging enable<br>bit[15:8] : Reserved | R/W | 0x0000 |
| 0x74 | Reserved | | |
| 0x75 | port 10 Out queue aging enable<br>bit[7:0] : port 10 Q7~Q0 queue aging enable<br>bit[15:8] : Reserved | R/W | 0x0000 |
| 0x76 | Port Group B Out queue schedule mode<br>bit[2:0] : Schedule mode<br>0 : First come First schedule<br>1 : WRR/WFQ/BW/TWRR<br>2 : SPx1 + WRR/WFQ/BW/TWRR x7<br>3 : SPx2 + WRR/WFQ/BW/TWRR x6<br>4 : SPx4 + WRR/WFQ/BW/TWRR x4<br>5 : SPx8<br>6 : LLQx1 + WFQ/BW/TWRR x7 (LLQ in WRR will switch to SP)<br>7 : LLQx2 + WFQ/BW/TWRR x6 (LLQ in WRR will switch to SP)<br>bit[4:3] : BW throttle period sel / TWRR tickle unit<br>00 : 64 kbit/s / 51.2 ms 01: 1 Mbit/s / 3.19 ms<br>10 : 2 Mbit/s / 1.55 ms 11: 4 Mbit/s / 0.82 ms<br>bit[6:5] : Queue method( 00: WRR 01: BW<br>10: WFQ 11: TWRR)<br>bit[7] : Delay Bound mode (Q7/Q6/Q5/Q4/Q3/Q2/Q1 )<br>(Q0 queue out only when Q7~Q1 is empty )<br>must set bit[12:11] = 10, at the same time<br>bit[8] : Q0 BE(best effort) queue disable (only bit[13] = 1)<br>(Q0 is also delay bound queue if disable BE)<br>bit[9] : Port Group B schedule mode setting enable<br>0: set as Group A 1: set as Group B<br>(PG08.0x01[8:0]) (PG08.0x76[8:0]) | R/W | 0x200 |
| 0x77 | Out queue Port Group A/B selection<br>bit[7:0] : port 1 ~ port 8 Group A/B select<br>1: Group B 0 : Group A | R/W | 0x0000 |
| 0x78 | Out queue Port Group A/B selection<br>bit[12][8] : port 9 ~ port 10 Group A/B select<br>1: Group B 0 : Group A | R/W | 0x0000 |
| 0x79 | Port Group B Out queue schedule ratio 0<br>bit[7:0] : Q0 weight / bandwidth allocate /<br>bit[15:8] : Q1 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x7A | Port Group B Out queue schedule ratio 1<br>bit[7:0] : Q2 weight / bandwidth allocate /<br>bit[15:8] : Q3 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |
| 0x7B | Port Group B Out queue schedule ratio 2<br>bit[7:0] : Q4 weight / bandwidth allocate /<br>bit[15:8] : Q5 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit | R/W | 0x0000 |
| 0x7C | Port Group B Out queue schedule ratio 3<br>bit[7:0] : Q6 weight / bandwidth allocate /<br>bit[15:8] : Q7 weight / bandwidth allocate /<br>(value =0 : weight 128 /unlimited bandwidth / queue in prohibit<br><br>Note 1: if 0x61 SRP parameter is setup with value > 0 ,<br>Q7 BW = mg_oq_wan_ratio3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_wan_ratio3[7:0] x 64 Bytes / slot time<br><br>Note 2: The WAN port max BW (if port SRP parameter > 0)<br>Q7 BW = mg_oq_wan_max_bw3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_wan_max_bw3[7:0] x 64 Bytes / slot time | R/W | 0x0000 |
| 0x7D | Port Group B Out queue 0 latency threshold<br>bit[7:0] : Q0 latency threshold<br>bit[15:8] : Q1 latency threshold | R/W | 0x80FF |
| 0x7E | Port Group B Out queue 1 latency threshold<br>bit[7:0] : Q2 latency threshold<br>bit[15:8] : Q3 latency threshold | R/W | 0x2040 |
| 0x7F | Port Group B Out queue 2 latency threshold<br>bit[7:0] : Q4 latency threshold<br>bit[15:8] : Q5 latency threshold | R/W | 0x0810 |
| 0x80 | Port Group B Out queue 3 latency threshold<br>bit[7:0] : Q6 latency threshold<br>bit[15:8] : Q7 latency threshold | R/W | 0x0204 |
| 0x81 | Port Group B Out queue Max Bandwidth<br>bit[7:0] : Q0 max bandwidth<br>bit[15:8] : Q1 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |
| 0x82 | Port Group B Out queue Max Bandwidth<br>bit[7:0] : Q2 max bandwidth<br>bit[15:8] : Q3 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |
| 0x83 | Port Group B Out queue Max Bandwidth<br>bit[7:0] : Q4 max bandwidth<br>bit[15:8] : Q5 max bandwidth<br>(value =0 : unlimited) | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x84 | Port Group B Out queue Max Bandwidth<br>bit[7:0] : Q6 max bandwidth<br>bit[15:8] : Q7 max bandwidth<br>(value =0 : unlimited)<br><br>Ethernet AV setup:<br>The WAN port max BW (if port SRP parameter > 0)<br>Q7 BW = mg_oq_wan_max_bw3[15:8]x 64 Bytes/ slot time<br>Q6 BW = mg_oq_wan_max_bw3[7:0] x 64 Bytes / slot time | R/W | 0x0000 |
| 0x85 ~ 0x8C | Egress Rate control for port 1 ~ port 8<br>bit[13:0] : Output Rate control<br>bit[15:14] : 00 : write Both PIR/CIR<br>01 : write PIR<br>10 : read CIR<br>11 : read PIR<br>(CIR : committed information rate : 2nd bucket token rate )<br>(PIR : peak information rate : 1st bucket token rate) | R/W | 0x0000 |
| 0x8D ~ 0x9C | Reserved | | |
| 0x9D | Egress Rate control for port 9<br>bit[13:0] : Output Rate control<br>bit[15:14] : 00 : write Both PIR/CIR<br>01 : write PIR<br>10 : read CIR<br>11 : read PIR<br>(CIR : committed information rate : 2nd bucket token rate )<br>(PIR : peak information rate : 1st bucket token rate) | R/W | 0x0000 |
| 0x9E ~ 0xA0 | Reserved | | |
| 0xA1 | Egress Rate control for port 10<br>bit[13:0] : Output Rate control<br>bit[15:14] : 00 : write Both PIR/CIR<br>01 : write PIR<br>10 : read CIR<br>11 : read PIR<br>(CIR : committed information rate : 2nd bucket token rate )<br>(PIR : peak information rate : 1st bucket token rate) | R/W | 0x0000 |
| 0xA2 | port 1 ~ port 2 Out queue rate control setting<br>bit[7:0] : port 1 Q7~Q0, disable egress rate calculation.<br>bit[15:8] : port 2 Q7~Q0, disable egress rate calculation.<br>(ex: if WFQ mode, set Q6~Q0 has 75% bandwidth take into egress rate count , then Q7 take the (100% - 75%) Bandwidth. | R/W | 0x0000 |
| 0xA3 ~ 0xA5 | port 3 ~ port 8 queue rate control setting<br>bit[7:0] : port xx Q7~Q0, disable egress rate calculation.<br>bit[15:8] : port xy Q7~Q0, disable egress rate calculation.<br>(ex: if WFQ mode, set Q6~Q0 has 75% bandwidth take into egress rate count , then Q7 take the (100% - 75%) Bandwidth. | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xA6 ~ 0xAD | Reserved | | |
| 0xAE | port 9 queue rate control setting<br>bit[7:0] : port 9 Q7~Q0, disable egress rate calculation.<br><br>(ex: if WFQ mode, set Q6~Q0 has 75% bandwidth take into egress rate count , then Q7 take the (100% - 75%) Bandwidth. | R/W | 0x0000 |
| 0xAF | Reserved | | |
| 0xB0 | port 10 queue rate control setting<br>bit[7:0] : port 10 Q7~Q0, disable egress rate calculation.<br><br>(ex: if WFQ mode, set Q6~Q0 has 75% bandwidth take into egress rate count , then Q7 take the (100% - 75%) Bandwidth.<br>(ex: if BW mode, set Q3~Q0 has total 25 % bandwidth take into egress rate Count , and Q7~Q4 total can use 75 % , and per Queue can setup SBM/DBM separately. | R/W | 0x00 |
| 0xB1 | Out queue parameter setup<br>bit[3:0] : 2nd bucket size[3:0] (leaky bucket size)<br>4'b0000: 4 KB<br>4'b0001: 8 KB<br>4'b0010: 16 KB<br>4'b0011: 32 KB<br>4'b0100: 64 KB<br>4'b0101: 128 KB<br>4'b0110: 256 KB<br>4'b0111: 512 KB<br>4'b1000: 1024 KB<br>4'b1001: 2 KB<br>4'b1010: 1 KB<br>4'b1011: 512 B<br>4'b1100: 256 B<br>4'b1101: 128 B<br>4'b1110: 64 B<br>4'b1111: 32 B<br>The max. output bytes no. in one period time<br>= Rate control configure value x 64k bps<br>( 4096 Bytes / 500 ms )(512 Bytes / 62.5 ms)<br><br>bit[6:4] : Bandwidth period<br>3'b000: ( 4096 Bytes / 500 ms)<br>3'b001: ( 2048 Bytes / 250 ms)<br>3'b010: ( 1024 Bytes / 125 ms)<br>3'b011: ( 512 Bytes / 62.5 ms)<br>3'b100: ( 256 Bytes / 31.25 ms)<br>3'b101: ( 128 Bytes / 15.625 ms) | R/W | 0x82FF |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | 3'b110: ( 64 Bytes / 7.8125 ms)<br>3'b111: ( 32 Bytes / 3.90625 ms)<br>The max. output bytes no. in one period time<br>= Rate control configure value x 64k bps<br>( 4096 Bytes / 500 ms )(512 Bytes / 62.5 ms)<br><br>bit[7] : Out queue rate-jitter regulator enable for egress rate ctrl<br>( **Leaky Bucket Method** 0: dis 1:en**)**<br>0 : Egress rate control using periodically Burst mode<br>1 : Egress rate control using leaky bucket mode<br>bit[8] : egress rate control trigger pause / drop enable<br>(0 :shaping traffic 1: policing traffic )<br>bit[9] : Out queue rate-jitter regulator enable burst mode for egress rate control (**Token Bucket method** 0: dis 1:en)<br>bit[10] : enable per queue pause / drop for BW mode<br>( if PG08.0xB1[11:10] =10, enable depend on PG08.0x55~0x63<br>if PG08.0xB1[11:10] =11, all queue is enable, regardless 0x55~0x63<br>bit[11] : enable out queue BW / WFQ pause / drop (BW / WFQ throttle)<br>bit[13:12] : TWRR Tickle unit type<br>00 : fixed Giga (51.2ms /3.19ms /1.55ms /0.82ms)<br>01 : fixed 100M (512ms /31.9ms /15.5ms /8.2ms)<br>10 : fixed 10 M (5120ms /319ms /155ms /82ms)<br>11 : bit time (51.2 /3.19 /1.55 /0.82)x$10^6$ bit time<br><br>bit[14] : Out queue congest condition (1: port based 0: queue based)<br>bit[15] : Queue bandwidth bytes cnt keep residue at Bw-assured mode ( precise BW for jumbo packet) | | |
| 0xB2 | TX queue mapping from TX "Port priority"<br>bit[7:0]: mg_oq_wlan_pri_en port 1 ~ port 8 | R/W | 0x0000 |
| 0xB3 | TX queue mapping from TX "Port priority"<br>bit[12][8]: mg_oq_wlan_pri_en port 9 ~port 10<br><br>Priority remapping :<br>TX priority remapping (PG08.0xB2~0xB3 > PG08.0x14 ~ 0x4D) ><br>RX priority remapping (PG06.0x24~0x5D ) ><br>ACL DSCP remark addr / VLAN Table Remark pri (PG06.0x17 ~ 0x1A ) | R/W | 0x0000 |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xB4 | bit[2:0]: Out queue rate control rate jitter enable burst mode<br>**burst size ( 1st Token Bucket size)**.<br>0 : 4 KB<br>1 : 16 KB<br>2 : 32 KB<br>3 : 64 KB<br>4 : 128 KB<br>5 : 256 KB<br>6 : 512 KB<br>7 : 1024 KB<br>bit[3] : WRED TAIL DROP (1:Enable 0:Disable)<br>bit[4] : WRED IP ECN enable<br>WRED notify explicit congestion notification in IP header. (Rule: ip packet , hit WRED drop , ECN bit = 01/10)<br>bit[5] : WRED mode 1 enable ( drop instantly without oq_rd_holb_cnt)<br>bit[6] : WRED normal en ( WRED enable ignore Delay bound weighted amount setting )<br>bit[7] : WRED HEAD DROP (1:Enable 0:Disable)<br>bit[8] : Out queue aging priority<br>0: Normal packet > Aging out packet (send normal first , aging packet still in queue , Use more Buffer)<br>1: Aging out packet > normal packet ( stop TX and switch to aging, afftect TX rate , less buffer used) | R/W | 0x068 |
| 0xB5 | port 1 ~ port 8 Out queue pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x22F |
| 0xB6 | port 1 ~ port 8 Out queue pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x1E1 |
| 0xB7 | port 9 Out queue pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x22F |
| 0xB8 | port 9 Out queue pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x1E1 |
| 0xB9 | port 10 Out queue pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x22F |
| 0xBA | port 10 Out queue pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x1E1 |
| 0xBB | Out queue Q0 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xBC | Out queue Q0 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xBD | Out queue Q1 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xBE | Out queue Q1 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |

| Output Queue Register : Page 0x08 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0xBF | Out queue Q2 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xC0 | Out queue Q2 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xC1 | Out queue Q3 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xC2 | Out queue Q3 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xC3 | Out queue Q4 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xC4 | Out queue Q4 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xC5 | Out queue Q5 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xC6 | Out queue Q5 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xC7 | Out queue Q6 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xC8 | Out queue Q6 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xC9 | Out queue Q7 pause on threshold setting<br>bit[10:0] : out queue pause on threshold | R/W | 0x07F |
| 0xCA | Out queue Q7 pause off threshold setting<br>bit[10:0] : out queue pause off threshold | R/W | 0x01F |
| 0xCB<br>0xCD | Reserved | | |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

### 6.12 PTP Control Register

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x00 | PTP clock reset bit<br>bit[0] : 1: clear the PTP clock reference register configuration to default<br>reg.0x10 reg.0x11<br>reg.0x12 reg.0x13<br>reg.0x14 reg.0x15<br>reg.0x16 reg.0x17<br>0: no action | SC | 0x0 |
| 0x01 | PTP time stamps read command<br>bit[3:0] : time stamp address<br>bit[4] : ingress/egress index<br>1 : egress time stamps<br>0 : ingress time stamps<br>bit[9:5] : port index ,from 0x00 ~ 0x07, 0x18, 0x1C (port 1 ~ port 10 and event)<br>bit[11:10] : do not care<br>bit[12] : data register (0x0B/0x0C/0x0D/0x0E) read indication<br>1 : read back data is the data you last written in<br>0 : the read back data is time stamp data or current PTP time<br>bit[13] : Reserved<br>bit[14] : Reserved<br>bit[15] : command trigger<br>1 : issue a command<br>0 : idle | R/W | 0x0000 |
| 0x02 | PTP general configuration register<br>bit[0] : Enable the PTP clock<br>1: Enable the PTP real time clock<br>0: Disable the PTP real time clock<br>bit[1] : MAC PTP stamp buffer overwrite enable<br>1: write buffers from 0 to 15 in circles<br>0: no more write when buffers full<br>bit[2] : Programmable output enable<br>1: Enable the output of programmable clock<br>0: Disable the output<br>bit[3] :EVENT input trigger positive edge detection<br>1: Enable positive edge detection<br>0: Disable positive edge detection<br>bit[4] :EVENT input trigger falling edge detection<br>1: Enable falling edge detection<br>0: Disable Falling edge detection<br>bit[5] : EVENT PTP stamp buffer overwrite enable<br>1: write buffers from 0 to 7 in circles<br>0: no more write when buffers full | R/W | 0x000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[6]: PTP Trigger Out Function Enable<br>1: Enable PTP trigger out function<br>0: Disable PTP trigger out function<br>bit[7]: PTP Trigger Out "Type" Selection<br>1: Period<br>0: Pulse<br>bit[8]: PTP Trigger Out "High Duration" Setting<br>1: Multiple Cycles (refers to 0x18 register, bit[15:8])<br>0: One Cycle<br>bit[10:9]: PTP Pulse Per Second Pin (ptp_pps) source selection<br>0: PTP Pulse Per Second (ptp_pps)<br>1: PTP programming output clock (reference: 0x02[2])<br>2: PTP Trigger output clock<br>3: Reserve | | |
| 0x03 | PTP port time stamp enable<br>bit[0] : port 1 ingress time stamp enable<br>bit[1] : port 1 egress time stamp enable<br>bit[2] : port 2 ingress time stamp enable<br>bit[3] : port 2 egress time stamp enable<br>bit[4] : port 3 ingress time stamp enable<br>bit[5] : port 3 egress time stamp enable<br>bit[6] : port 4 ingress time stamp enable<br>bit[7] : port 4 egress time stamp enable<br>bit[8] : port 5 ingress time stamp enable<br>bit[9] : port 5 egress time stamp enable<br>bit[10] : port 6 ingress time stamp enable<br>bit[11] : port 6 egress time stamp enable<br>bit[12] : port 7 ingress time stamp enable<br>bit[13] : port 7 egress time stamp enable<br>bit[14] : port 8 ingress time stamp enable<br>1 : Enable, 0 : Disable<br>bit[15] : port 8 egress time stamp enable<br>1 : Enable, 0 : Disable | R/W | 0x0000 |
| 0x04<br>0x05 | Reserved | | |
| 0x06 | EVENT and port 9 ~ port 10 PTP time stamp enable<br>bit[1:0]: (same definition with register 0x03, except port 9)<br>bit[7:2]: Reserved<br>bit[9:8] : (same definition with register 0x03, except port 10)<br>bit[10] : EVENT input time stamp enable | R/W | 0x000 |
| 0x07 | PTP time stamp clear register<br>bit[0] : port 1 ingress stamp counter/full notification clear<br>bit[1] : port 1 egress stamp counter/full notification clear<br>bit[2] : port 2 ingress stamp counter/full notification clear<br>bit[3] : port 2 egress stamp counter/full notification clear | SC | 0x0000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[4] : port 3 ingress stamp counter/full notification clear<br>bit[5] : port 3 egress stamp counter/full notification clear<br>bit[6] : port 4 ingress stamp counter/full notification clear<br>bit[7] : port 4 egress stamp counter/full notification clear<br>bit[8] : port 5 ingress stamp counter/full notification clear<br>bit[9] : Pport 5 egress stamp counter/full notification clear<br>bit[10] : port 6 ingress stamp counter/full notification clear<br>bit[11] : port 6 egress stamp counter/full notification clear<br>bit[12] : port 7 ingress stamp counter/full notification clear<br>bit[13] : port 7 egress stamp counter/full notification clear<br>bit[14] : port 8 ingress stamp counter/full notification clear<br>1 : clear to zero, 0 : no action<br>bit[15] : port 8 egress stamp counter/full notification clear<br>1 : clear to zero, 0 : no action | | |
| 0x08<br>0x09 | Reserved | | |
| 0x0A | EVENT and port 9 ~port 10 PTP time stamp clear register<br>bit[1:0] : same definition with register 0x07, except port 9)<br>bit[7:2]: Reserved<br>bit[9:8] : (same definition with register 0x07, except port 10)<br>bit[10] : EVENT input time stamp clear | SC | 0x000 |
| 0x0B | PTP time data for nano-second<br>bit[15:0] : nano-second[15:0] | R/W | 0x0000 |
| 0x0C | PTP time data for nano-second<br>bit[13:0] : nano-second[29:16] | R/W | 0x0000 |
| 0x0D | PTP time data for second<br>bit[15:0] : second[15:0] | R/W | 0x0000 |
| 0x0E | PTP time data for second<br>bit[15:0] : second[31:16] | R/W | 0x0000 |
| 0x0F | PTP clock control register<br>bit[0] : command trigger<br>1: issue a command 0: idle<br>bit[2:1] : 00 : read back the current PTP clock into register 0x0B~0x0E<br>01 : Set the PTP time date 0x0B~0x0E to current PTP clock<br>10 : Add the PTP time date 0x0B~0x0E to current PTP clock<br>11 : Sub the PTP time date 0x0B~0x0E from current PTP clock<br>Note! The register bit 0x01[12] should be “0” for reading back the current clock from 0x0B~0x0E | R/W<br><br><br><br><br><br>SC | 0x0000 |
| 0x10 | PTP frequency addend value configuration<br>Set the PTP real time clock divided from source (125MHz)<br>bit[15:0] : addend value for the frequency compensation [15:0] | R/W | 0x0000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x11 | PTP frequency addend value configuration<br>Set the PTP real time clock divided from source (125MHz)<br>bit[15:0] : addend value for the frequency compensation [31:16] | R/W | 0x8000 |
| 0x12 | PTP clock period register<br>The period of PTP real time clock set<br>bit[15:0] : 16bits clock period in nano-second<br>Note : (a) The default clock period is 16ns, modification is not recommend<br>(b) this register should be derived from register 0x10 and 0x11 | R/W | 0x0010 |
| 0x13 | PTP frequency compensation configuration based on reg.0x10 and 0x11<br>bit[15:0] : the frequency compensation value [15:0] | R/W | 0x0000 |
| 0x14 | PTP frequency compensation configuration based on reg.0x10 and 0x11<br>bit[9:0] : the frequency compensation value [25:16] | R/W | 0x000 |
| 0x15 | PTP frequency compensation duration in numbers of PTP clock cycles<br>bit[15:0] : duration in numbers of PTP clock cycles [15:0] | R/W | 0x0000 |
| 0x16 | PTP frequency compensation duration in numbers of PTP clock cycles<br>bit[9:0] : duration in numbers of PTP clock cycles [25:16]<br>Note : This register takes actively only when reg.0x17[1]=1, and the clock cycle is based PTP period defined in reg.0x10 and 0x11 | R/W | 0x000 |
| 0x17 | PTP frequency compensation control register<br>bit[0] : command trigger<br>1: frequency correction enable<br>0: Disable<br>bit[1] : 1 : use the rate duration set in registers 0x15 and 0x16<br>0 : always correction<br>bit[2] : 1 : frequency sub<br>0 : frequency add | R/W | 0x0 |
| 0x18 | PTP programmable output configuration<br>bit[7:0] : the “low” duration in numbers of PTP clock cycles<br>bit[15:8] : the “high” duration in numbers of PTP clock cycles<br>Note : This register takes actively only when reg.0x02[2]=1, and the clock cycle is based PTP period defined in reg.0x10 and 0x11 | R/W | 0x0000 |
| 0x19 | PTP interrupt indication enable when a frame has been time stamped<br>bit[0] : port 1 ingress time stamp interrupt enable<br>bit[1] : port 1 egress time stamp interrupt enable<br>bit[2] : port 2 ingress time stamp interrupt enable<br>bit[3] : port 2 egress time stamp interrupt enable<br>bit[4] : port 3 ingress time stamp interrupt enable<br>bit[5] : port 3 egress time stamp interrupt enable<br>bit[6] : port 4 ingress time stamp interrupt enable<br>bit[7] : port 4 egress time stamp interrupt enable<br>bit[8] : port 5 ingress time stamp interrupt enable | R/W | 0x0000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | bit[9] : port 5 egress time stamp interrupt enable<br>bit[10] : port 6 ingress time stamp interrupt enable<br>bit[11] : port 6 egress time stamp interrupt enable<br>bit[12] : port 7 ingress time stamp interrupt enable<br>bit[13] : port 7 egress time stamp interrupt enable<br>bit[14] : port 8 ingress time stamp interrupt enable<br>1: Enable 0: Disable<br>bit[15] : port 8 egress time stamp interrupt enable<br>1: Enable 0: Disable | | |
| 0x1A<br>0x1B | Reserved | | |
| 0x1C | EVENT and port 9 ~port 10 PTP time stamp interrupt enable<br>bit[1:0] : (same definition with register 0x19, except port 9)<br>bit[7:2] : Reserved<br>bit[9:8] : (same definition with register 0x19, except port 10 )<br>bit[10] : EVENT input time stamp interrupt enable | R/W | 0x000 |
| 0x1D | PTP time stamp status register<br>bit[0] : port 1 stamp status for ingress<br>1: notify a time stamp has been write into buffer<br>0: no time stamp wrote after last buffer read<br>bit[1] : port 1 stamp status for egress<br>1: notify a time stamp has been write into buffer<br>0: no time stamp wrote after last buffer read<br>bit[2] : port 2 stamp status for ingress<br>bit[3] : port 2 stamp status for egress<br>bit[4] : port 3 stamp status for ingress<br>bit[5] : port 3 stamp status for egress<br>bit[6] : port 4 stamp status for ingress<br>bit[7] : port 4 stamp status for egress<br>bit[8] : port 5 stamp status for ingress<br>bit[9] : port 5 stamp status for egress<br>bit[10] : port 6 stamp status for ingress<br>bit[11] : port 6 stamp status for egress<br>bit[12] : port 7 stamp status for ingress<br>bit[13] : port 7 stamp status for egress<br>bit[14] : port 8 stamp status for ingress<br>bit[15] : port 8 stamp status for egress<br>Note : the status is cleared when any of the correspond buffers has been read | RO | 0x0000 |
| 0x1E<br>0x1F | Reserved | | |
| 0x20 | EVENT and port 9 ~ port 10 PTP time stamp status register<br>bit[1:0] :(same definition with register 0x1D, except port 9)<br>bit[7:2]:Resrved<br>bit[9:8] : (same definition with register 0x1D, except port 10)<br>bit[10] : EVENT input time stamp status register | RO | 0x000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x21 | port 9 PTP ingress latency correction<br>bit[15:0] : correct the ingress time by subtracting the latency correction (ns) | R/W | 0x0000 |
| 0x22 ~ 0x24 | Reserved | | |
| 0x25 | port 10 PTP ingress latency correction<br>bit[15:0] : correct the ingress time by subtracting the latency correction (ns) | R/W | 0x0000 |
| 0x26 | port 9 PTP egress latency correction<br>bit[15:0] : correct the egress time by adding the latency correction (ns) | R/W | 0x0000 |
| 0x27 ~ 0x29 | Reserved | | |
| 0x2A | port 10 PTP egress latency correction<br>bit[15:0] : correct the egress time by adding the latency correction (ns) | R/W | 0x0000 |
| 0x2B | 10M TP ingress latency correction (embedded PHY)<br>bit[15:0] : correct the ingress time by subtracting the latency correction (ns) | RO | 0x0564 |
| 0x2C | 100M TP ingress latency correction (embedded PHY)<br>bit[15:0] : correct the ingress time by subtracting the latency correction (ns) | RO | 0x0140 |
| 0x2D | 100M Fiber ingress latency correction (embedded PHY)<br>bit[15:0] : correct the ingress time by subtracting the latency correction (ns) | RO | 0x0128 |
| 0x2E | 10M TP egress latency correction (embedded PHY)<br>bit[15:0] : correct the egress time by adding the latency correction (ns) | RO | 0x0474 |
| 0x2F | 100M TP egress latency correction (embedded PHY)<br>bit[15:0] : correct the egress time by adding the latency correction (ns) | RO | 0x0088 |
| 0x30 | 100M Fiber egress latency correction (embedded PHY)<br>bit[15:0] : correct the egress time by adding the latency correction (ns) | RO | 0x0088 |
| 0x31 | PTP Specific Time Trigger Out data for nano-second<br>bit[15:0]: nano-second [15:0] | R/W | 0x0000 |
| 0x32 | PTP Specific Time Trigger Out data for nano-second<br>bit[15:0]: nano-second[29:16] | R/W | 0x0000 |
| 0x33 | PTP Specific Time Trigger Out data for second<br>bit[15:0] : second[15:0] | R/W | 0x0000 |
| 0x34 | PTP Specific Time Trigger Out data for second<br>bit[15:0] : second[31:16] | R/W | 0x0000 |
| 0x35 | PTP Specific Time Trigger Out Period Setting<br>bit[15:0]: Period Value Setting[15:0] | R/W | 0x0000 |

| PTP Register : Page 0x09 | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x36 | PTP Specific Time Trigger Out Period Setting. (Unit: RTC Rate)<br>bit[15:0]: Period Value Setting[31:16]<br><br>[Example] : The value is calculated as follows:<br>If PTP real time clock is 62.5MHz (where Reg0x11: 0x8000), the 125us period should be setting as : 0x1e84 = 125us / 16ns<br><br>Note :<br>● PTP Specific Time Trigger out Period Value == **0x0000h** indicates **disable** the period function.<br>● PTP Specific Time Trigger out Period Value must large than "**high duration cycles** (0x18 register, bit[15:8])"; otherwise, it will be **disable** the period function. | R/W | 0x0000 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

### 6.13 IRMP Control Register

| IRMP Register : Page 0x0A | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | IRMP operation<br><br>bit[8:0] : The period of IRMP time out, the unit is “second”<br>If no IRMP command is received in (bit[8:0]~bit[8:0]+1)sec., a granted port will be released and send out a goodbye frame<br>(default value is 0x12C, 300sec)<br>bit[11:9] : (Reserved)<br>bit[12] : password is needed<br>1 : The IRMP should check password (default)<br>0 : no password needed<br>bit[13] : The IRMP power abnormal indication enable<br>1 : when switch detects the power level is under threshold, a last gasp frame is send out by granted port (default)<br>0 : No power detection function<br>bit[14] : IRMP topology function enable<br>1 : Enable (default)<br>0 : Disable<br>bit[15] : Destination Address is limited to switch MAC address.<br>1 : IRMP check the DA of frame with the switch MAC address only<br>0 : IRMP check the DA of frame with switch MAC address and broadcast address (default) | R/W | 0x712C |
| 0x02 | IRMP function enable for port 1 ~ port 8, 1 bit per port<br><br>bit[0] : IRMP function enable for port 1<br>1 : IRMP enable (default)<br>0 : IRMP disable<br>bit[7:1] : IRMP function enable for port 2 ~ port 8<br>The same definition with bit[0] | R/W | 0xFFFF |
| 0x03 | IRMP function enable for port 9 ~ port 10, 1 bit per port<br>bit[12][8] : IRMP function enable for port 9 ~ port 10<br>The same definition with register 0x02 | R/W | 0x1FFF |
| 0x04 | IRMP ether-type definition<br>bit[15:0] : The ether-type definition of IRMP (default : 16’h0806) | R/W | 0x8931 |
| 0x05 | IRMP sub-type definition<br>bit[15:0] : The sub-type definition of IRMP (default : 16’hFFFE) | R/W | 0xFFFE |
| 0x06 | Vender ID [15:0]<br>bit[15:0] : The vender identifier [15:0] (default : 0xC300) | R/W | 0xC300 |
| 0x07 | Vender ID [31:16]<br>bit[15:0] : The vender identifier [31:16] (default : 0x0090) | R/W | 0x0090 |
| 0x08 | Device ID and Version<br><br>bit[3:0] : The IRMP version control (default : 4’h0)<br>bit[15:4] : The device identifier [11:0] (default : 12’h829) | R/W | 0x8290 |

| IRMP Register : Page 0x0A | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x09 | IRMP Password [15:0]<br>bit[15:0] : The IRMP password [15:0] (default : 16'h0000) | R/W | 0x0000 |
| 0x0A | IRMP password [31:16]<br>bit[15:0] : The IRMP password [31:16] (default : 16'h0000) | R/W | 0x0000 |
| 0x0B | IRMP connect ID<br>bit[15:0] : The random ID generated for each connection | RO | 0x0000 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

Note : The IRMP operation flow, please refer to IRMP application note.

### 6.14 Advance EEPROM code Register

| Advance EEPROM code Register : Page 0x0B | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x01 | advance EEPROM operation<br><br>bit[0] : The advance EEPROM function enable<br>1 : Enable<br>0 : Disable (default)<br>bit[1] : The advance EEPROM code start point bit[16] (MSB)<br>(refer to register 0x02)<br>bit[2] : action code hold<br>1 : hold the command for debugging<br>0 : normal processing command code (default)<br>bit[3] : action code trigger again<br>1 : re-start the action code from start address<br>0 : normal (default) | R/W | 0x0 |
| 0x02 | The advance EEPROM code start bit[15:0]<br>bit[15:0] : start point<br><br>Note : Four EEPROM bytes to form a command code,<br>The code begin at EEPROM address is<br>{start point [16:0], 2'b00}<br>Note : MSB bit[16] is defined in 0x01[1] | R/W | 0x0000 |
| 0x03 | Data buffer 0 for advance EEPROM code<br>bit[15:0] : buffer_0 | RO | 0x0000 |
| 0x04 ~ 0x0A | Data buffer 1 ~ data buffer 7 for advance EEPROM code<br>bit[15:0] : buffer | RO | 0x0000 |
| 0x0B | operation temporary 0 for advance EEPROM code<br>bit[15:0] : temp_0 | RO | 0x0000 |
| 0x0C ~ 0x0E | operation temporary 1 ~ 3 for advance EEPROM code<br>bit[15:0] : temp_1 ~ 3 | RO | 0x0000 |
| 0x0F | Logic result of advance EEPROM code<br>bit[3:0] : logic temporary bit[3:0]<br>1 : true<br>0 : false<br>bit[4] : The final result after command : result check<br>1 : true<br>0 : false | RO | 0x00 |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

Note : The advance EEPROM operation flow, please refer to Advance_EEPROM_code application note.

### 6.15 MISC Register

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x00 | Switch software reset<br>bit[0] : 1: reset the switch of IP1810I, all parameters are maintained<br>0: no action<br>bit[4] : 1: reset the switch of IP1810I, all parameters are reset to default<br>0: no action | SC | 0x00 |
| 0x01 | BIST Enable and reset mask set<br>bit[0] : Software reset mask for BIST complete<br>bit[1] : Software reset mask for buffer over-/under-flow<br>bit[2] : Software reset mask for output queue critical event<br>bit[3] : Software reset mask for share management<br>bit[4] : Software reset mask for output queue overflow<br>bit[5] : Frame buffer ECC enable<br>bit[6] :MIB memory block BIST enable<br>1 : Enable BIST 0 : Disable BIST<br>bit[7] :LUT memory block BIST enable<br>bit[8] :Source List memory block BIST enable<br>bit[9] :Multicast Table block BIST enable<br>bit[10] :VLAN memory block BIST enable<br>bit[11] :Buffer management memory block BIST enable<br>bit[12] : (Reserved)<br>bit[13] :Frame buffer memory block BIST enable<br>bit[14] :Output queue memory block BIST enable<br>bit[15] :PTP memory block BIST enable<br><br>Note : some memory blocks share the same BIST enable bit | R/W | 0xFFE0 |
| 0x02 | Build in self test information<br>bit[3:0] : MIBs 4 Blocks BIST result<br>1 : OK 0 : Fail<br>bit[4] : LUT BIST result<br>bit[5] : Source List BIST result<br>bit[6] : Multicast table BIST result<br>bit[7] : VLAN BIST result<br>bit[8] : Buffer management BIST result<br>bit[9] : Frame buffer BIST result<br>bit[10] : Output queue BIST result<br>bit[11] : PTP BIST result<br>bit[12] : LUT memory is under BIST<br>1 : BIST is going 0 : BIST finish<br>bit[13] : VLAN /Multicast memory is under BIST<br>bit[14] : Output queue memory is under BIST<br>bit[15] : Frame Buffer is under BIST | RO | 0x0FFF |

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x03 | Switch input/output pin configuration<br>bit[0] : port 10 is connected to CPU indication<br>1: connect to CPU<br>0: normal port<br>bit[1] : port 10 special tag function enable when port 10 is CPU port (bit[0]=1)<br>1 : Enable 0: Disable<br>bit[2] : RGMII pads internal LDO power output disable<br>1: Power down the LDO output<br>0: LDO power output normally<br>bit[7:3] : Reserved<br>bit[8] : port 10 MII PHY mode enable<br>1 : MII clocks is provided by IP1810I<br>0 : MII clocks is provided by external device<br>bit[12:9] : Reserved<br>bit[13] : port 10 turbo MII enable<br>1 : Enable turbo MII (MII TX/RX clock is set to 50MHz)<br>0 : Disable turbo MII (MII TX/RX clock is 25/2.5MHz)<br>bit[14] : CPU I/F speed selection<br>1: normal speed (2 bits “ID” + 8 bits address)<br>0: high speed (8 bits address + 2 bits “00”)<br>bit[15] : (Reserved should be “0”) | R/W | 0x4008 |
| 0x04 | Special TAG Type/Len setting<br>bit[15:0]: special tag type/length item | R/W | 0x9126 |
| 0x05 | Interrupt output configuration<br>bit[0] : interrupt enable for CPU read/write SMI done<br>bit[1] : interrupt enable for link status change<br>bit[2] : interrupt enable for link operation mode change<br>bit[3] : interrupt enable for CPU read/write EEPROM done<br>bit[4] : interrupt enable for last gasp<br>bit[5] : interrupt enable for PTP<br>bit[6] : interrupt enable for loop detection<br>bit[7] : interrupt enable for critical event<br>1: interrupt enable 0: Disable<br>bit[13:8] : interrupt pulse period when bit[14]=1,<br>Pulse period = (bit[13:8]+1) x 8ns<br>bit[14] : interrupt trigger type<br>1: pulse trigger<br>0: level trigger (read and clear)<br>bit[15] : Interrupt is action high<br>1: high active<br>0: low active | R/W | 0x8B02 |

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x06 | Interrupt status<br>bit[0] : interrupt notification for CPU read/write SMI done<br>bit[1] : interrupt notification for link status change<br>bit[2] : interrupt notification for link operation mode change<br>bit[3] : interrupt notification for CPU read/write EEPROM done<br>bit[4] : interrupt notification for last gasp<br>bit[5] : interrupt notification for PTP<br>bit[6] : interrupt notification for loop detection<br>bit[7] : interrupt notification for critical event<br>1: interrupt occurred 0: normal | RO<br>RC | 0x00 |
| 0x07 | Last gasp configuration<br>bit[0] : last gasp detect enable<br>1: Enable 0:Disable<br>bit[1] : last gasp detect method<br>1: detect high 0: detect low<br>bit[2] : critical event detect enable<br>1: Enable 0:Disable<br>bit[3] : critical event detect method<br>1: detect high 0: detect low<br>bit[4] : switch fast test mode enable (for test only)<br>1: Enable 0: Disable<br>bit[5] : Reserved<br>bit[7:6] : Output pad driving current set [1:0] (EEPROM/LED/CPU)<br>00 : 1mA<br>01 : 3mA (default)<br>10 : 6mA<br>11 : 11mA<br>(Based on 2.5V)<br>bit[9:8] : output pad slew rate set [1:0] (EEPROM/LED/CPU)<br>00 : default<br>bit[12:10] : MDC/MDIO pad driving current set<br>000 : 5mA<br>001 : 9mA<br>010 : 14mA<br>011 : 19mA (default)<br>100 : 24mA<br>101 : 28mA<br>110 : 31mA<br>111 : 33mA<br>(Based on 2.5V)<br>bit[15:13] : MDC/MDIO pad slew rate set<br>000 : default | R/W | 0x0C4A |
| 0x08 | I/O configuration 0<br>bit[2:0] : port 9 Rx clock input delay set (0~7ns), adjust the relative phase between Rx clock and Rx data path<br>default : 000<br>bit[5:3] : port 9 Tx clock output delay set (0~7ns), adjust the relative phase between Tx clock and Tx data path<br>default : 000<br>bit[8:6] : port 9 RGMII output driving current set<br>000 : 5mA<br>001 : 9mA | R/W | 0x0480 |

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| | 010 : 14mA (default)<br>011 : 19mA<br>100 : 24mA<br>101 : 28mA<br>110 : 31mA<br>111 : 33mA<br>(Based on 2.5V)<br>bit[11:9] : Reserved<br>bit[12] : CPU watch dog reset disable<br>0 : enable reset from CPU (default)<br>1 : disable<br>bit[13] : CPU clock source selection bit<br>0 : internal PLL (default)<br>1 : clock source from AFE | | |
| 0x09 | Reserved | | |
| 0x0A | I/O configuration 2<br>bit[2:0] : port 10 Rx clock input delay set (0~7ns), adjust the relative phase between Rx clock and Rx data path<br>default : 000<br>bit[5:3] : port 10 Tx clock output delay set (0~7ns), adjust the relative phase between Tx clock and Tx data path<br>default : 000<br>bit[8:6] : port 10 RGMII/MII output driving current set<br>000 : 1mA<br>001 : 2mA<br>010 : 3mA (default)<br>011 : 4mA<br>100 : 5mA<br>101 : 6mA<br>110 : 7mA<br>111 : 15mA<br>(Based on 2.5V)<br>bit[11:9] : Reserved | R/W | 0x0480 |
| 0x0B | port 1 ~ port 8 Fiber mode set<br>bit[6:0] : port 1 ~ port 7 fiber mode enable<br>bit[7] : port 8 fiber mode enable<br>1: fiber mode 0: TP mode | R/W | 0x0000 |
| 0x0C | Reserved | | |
| 0x0D | CPU read/write EEPROM command<br>bit[7:0] : Byte Address<br>bit[10:8] : Device Address<br>bit[12:11] :(Reserved)<br>bit[13] : do not care<br>bit[14] : read/write operation<br>0 : read operation<br>1 : write operation<br>bit[15] : the read/write command trigger<br>0 : idle<br>1 : start command | R/W | 0x0000 |

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x0E | CPU read/write EEPROM command<br>bit[7:0] : High Byte Address<br>bit[11:10] : EEPROM force mode after initial read procedure<br>0: as initial Auto-detect 1: reserved<br>2: Force 24C01 ~ 24C16 3: unused<br>bit[14:12] : EEPROM clock period<br>000: 3200 ns<br>001: 2400 ns<br>01x: 1600 ns<br>1xx: 800 ns<br>bit[15] : restart EEPROM initial read procedure<br>Note: use for **24C01 ~ 24C16**<br>EEPROM address {00h, 01h} = 0x000B : command mode<br>EEPROM address {00h, 01h} = 0x000A : data mode | R/W | 0x0000 |
| 0x0F | CPU read/write EEPROM command data<br>bit[7:0] : in read command – the read back data<br>in write command – data want to write<br>bit[9:8] : Unused<br>bit[11:10]: size mode selection<br>bit[14:12]: EEPROM initial Read command mode area size<br>Size_mode = 0<br>1xx : 1Kbit (ROM addr : 0x00000 ~ 0x0007F)<br>011 : 2Kbit (ROM addr : 0x00000 ~ 0x000FF)<br>010 : 4Kbit (ROM addr : 0x00000 ~ 0x001FF)<br>001 : 8Kbit (ROM addr : 0x00000 ~ 0x003FF)<br>000 : 16Kbit (ROM addr : 0x00000 ~ 0x007FF)<br>Size_mode = 1<br>1xx : 16Kbit (ROM addr : 0x00000 ~ 0x007FF)<br>011 : 32Kbit (ROM addr : 0x00000 ~ 0x00FFF)<br>010 : 64Kbit (ROM addr : 0x00000 ~ 0x01FFF)<br>001 : 128Kbit (ROM addr : 0x00000 ~ 0x03FFF)<br>000 : 256Kbit (ROM addr : 0x00000 ~ 0x07FFF)<br>Size_mode = 2/3<br>1xx : 256Kbit (ROM addr : 0x00000 ~ 0x07FFF)<br>011 : 512Kbit (ROM addr : 0x00000 ~ 0x0FFFF)<br>010 : 1024Kbit (ROM addr : 0x00000 ~ 0x1FFFF)<br>001 : 2048Kbit (ROM addr : 0x00000 ~ 0x3FFFF)<br>000 : 4096Kbit (ROM addr : 0x00000 ~ 0x7FFFF)<br>bit[15] : unused<br>(EEPROM read stop criteria : (a) STOP sign or<br>(b) Read out configured Area size) | R/W | 0x0000 |
| 0x10 | Reserved | | |

| Misc/Syetem Register : Page 0x0C | | | |
|---|---|---|---|
| **Reg Addr.** | **Register Description** | **R/W** | **Default value** |
| 0x11 | IEEE802.3az TX clock silence enable<br><br>bit[0] : port 9 RGMII GTXCLK clock silence enable<br>1 : enable clock off during LPI 0:disable<br>bit[7:1] : Reserved<br>bit[8] : port 10 RGMII GTXCLK clock silence enable<br>1 : enable clock off during LPI 0:disable<br>bit[9] : port 10 MII MTXCLK clock silence enable<br>1 : enable clock off during LPI 0:disable | R/W | 0x000 |
| 0x12<br>~<br>0x21 | Reserved | | |
| 0xFF | Register Page selection<br>bit[5:0] : page selection | R/W | 0x00 |

# 7 Electrical Characteristics

## 7.1 Absolute Maximum Rating

Permanent device damage may occur if Absolute Maximum Ratings are applied. Functional operation should be restricted to the conditions as specified in the following section. Exposure to the Absolute Maximum Conditions for extended periods may affect device reliability.

| PARAMETER | | SYMBOL | MIN. | MAX. | UNIT |
|---|---|---|---|---|---|
| Supply Voltage | I/O | $V_{DDIO}$ | – 0.5 | +3.63V | V |
| | Core | $V_{DDCore}$ | – 0.5 | +1.26V | V |
| Input Voltage | | $V_I$ | – 0.5 | $V_{DDIO}$ | V |
| Output Voltage | | $V_O$ | – 0.5 | $V_{DDIO}$ | V |
| Storage Temperature | | $T_{STG}$ | -65 | +150 | °C |
| Operation Temperature | | $T_{OPT}$ | -40 | +85 | °C |
| IC Junction Temperature | | $T_J$ | | +125 | °C |

**Note:** The maximum ratings are the limit value that must never be exceeded even for short time.

# 8 AC Characteristics

## 8.1 Power On Sequence and Reset Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{DIFF}$ | Time difference among power sources | 0 | - | - | ms |
| $T_{PWR\_LEAD}$ | All power source ready before reset released | 10 | - | - | ms |
| $T_{CLK\_LEAD}$ | X1 clock valid before reset released | 10 | - | - | ms |
| $T_{SYS\_INIT}$ | System initial completed, this time to keep silence. | 10 | - | - | ms |

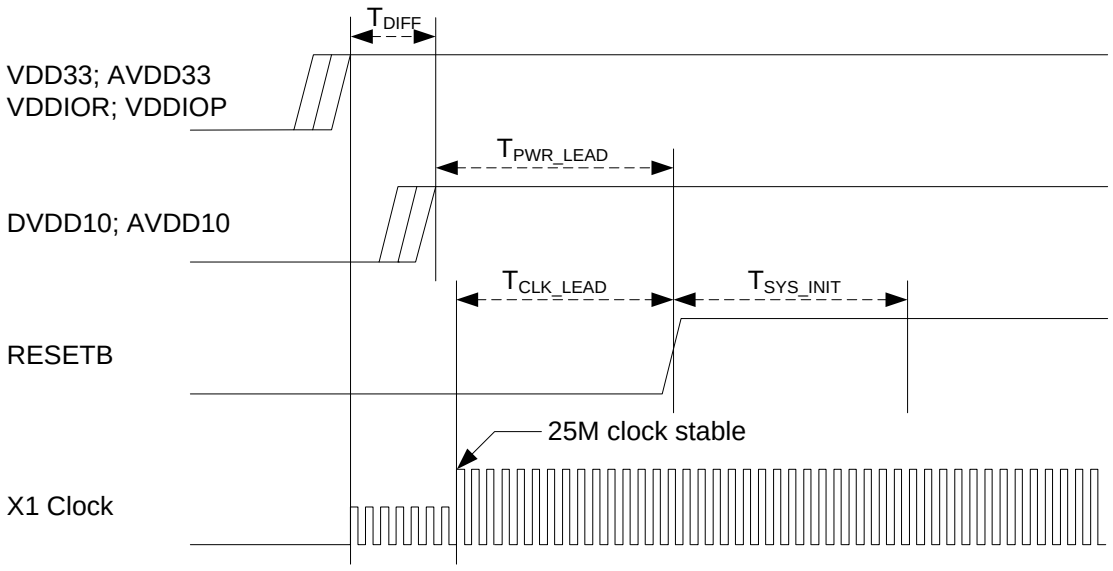
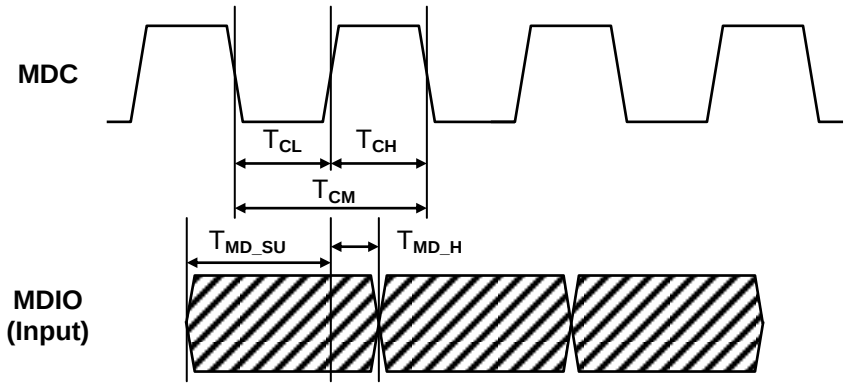


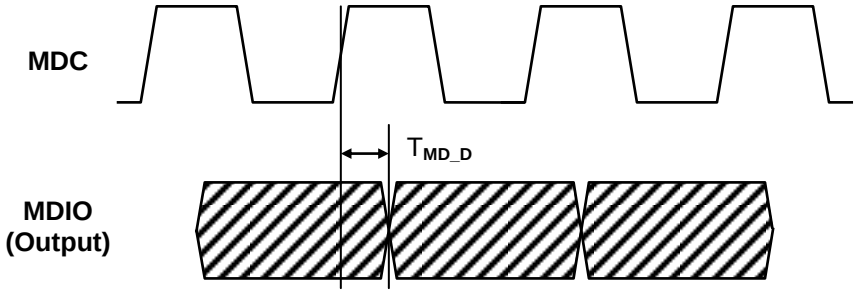

Figure 27 Power On Sequence and Reset timing Diagram

## 8.2 PHY Management (MDIO) Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{CH}$ | MDC High Time | - | 200 | - | ns |
| $T_{CL}$ | MDC Low Time | - | 200 | - | ns |
| $T_{CM}$ | MDCK cycle time | - | 400 | - | ns |
| $T_{MD\_SU}$ | MDIO set up time | 10 | - | - | ns |
| $T_{MD\_H}$ | MDIO hold time | 0 | - | - | ns |
| $T_{MD\_D}$ | MDIO output delay time | 200 | - | 210 | ns |




**MDIO Input Cycle**

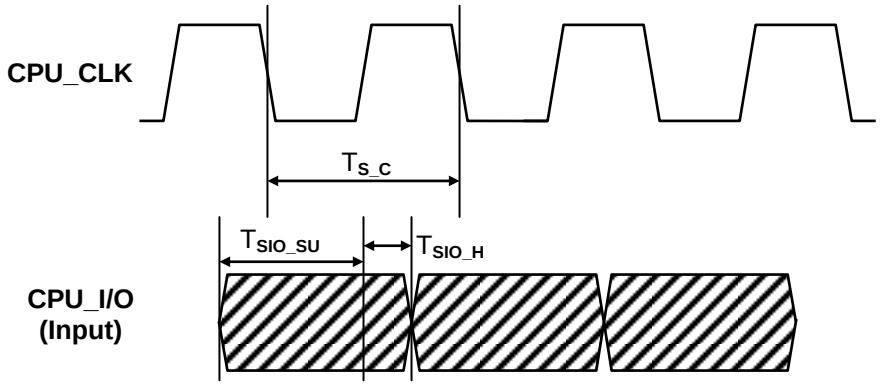



**MDIO Output Cycle**

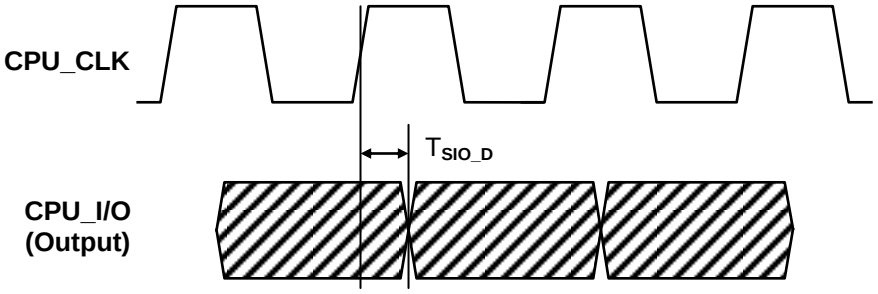
Figure 28 MDC/MDIO Read/Write cycle timing Diagram

## 8.3 CPU Serial Bus Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{S\_C}$ | Serial CPU Clock cycle time | 26 | - | - | ns |
| $T_{SIO\_SU}$ | Serial I/O set up time | 10 | - | - | ns |
| $T_{SIO\_H}$ | Serial I/O hold time | 10 | - | - | ns |
| $T_{SIO\_D}$ | Serial I/O output delay time | - | - | 20 | ns |




**Serial I/O Input Cycle**




**Serial I/O Output Cycle**

Figure 29 CPUIO Serial Bus timing Diagram

## 8.4 RGMII Timing

### 8.4.1 Rx Part Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{RGRxCLK}$ | Period of receive clock in giaga mode | - | 8 | - | ns |
| | Period of transmit clock in 100M mode | - | 40 | - | ns |
| | Period of transmit clock in 10M mode | - | 400 | - | ns |
| $T_{RGS}$ | RXDV, RXD to RXCLK setup time | 1 | - | - | ns |
| $T_{RGH}$ | RXCTL, RXD to RXCLK hold time | 1 | - | - | ns |

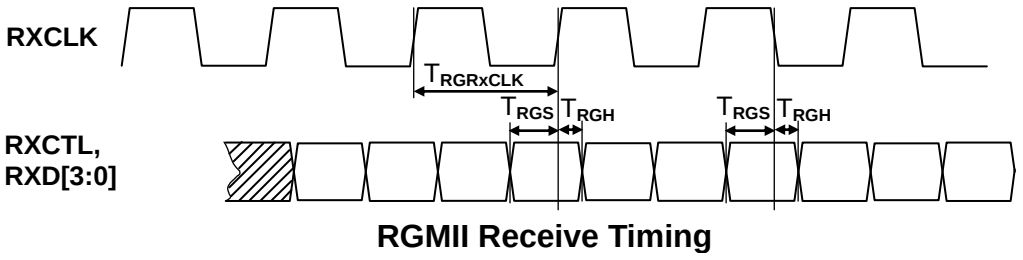



Figure 30 RGMII Receive Timing

### 8.4.2 Tx Part Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{RGTxCLK}$ | Period of transmit clock in giga mode | - | 8 | - | ns |
| | Period of transmit clock in 100M mode | - | 40 | - | ns |
| | Period of transmit clock in 10M mode | - | 400 | - | ns |
| $T_{RGTxD}$ | TXCTL, TXD output delay from GTXCLK edge (Note1) | 1.1 | - | 2.7 | ns |

Note1: $T_{RGTxD}$ allows implementation of delay on GTXCLK inside the transmitter. Refer to register 0x8 and 0xA of page 0xC.

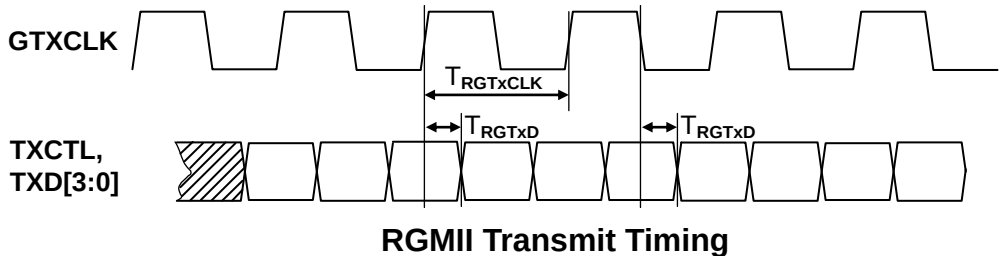



Figure 31 RGMII Transmit Timing

## 8.5 MII Timing

### 8.5.1 Receive Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{MRxCLK}$ | Receive clock period 100Mbps MII | - | 40 | - | ns |
| | Receive clock period 10Mbps MII | - | 400 | - | ns |
| $T_{MRx\_S}$ | RXDV, RXD to RXCLK setup time | 2 | - | - | ns |
| $T_{MRx\_H}$ | RXDV, RXD to RXCLK hold time | 0.5 | - | - | ns |

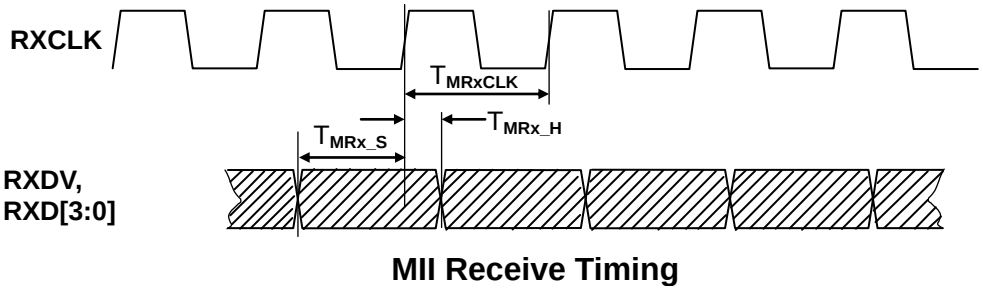



Figure 32 MII Receive Timing

### 8.5.2 Transmit Timing

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{MTxCLK}$ | Transmit clock period 100Mbps MII | - | 40 | - | ns |
| | Transmit clock period 10Mbps MII | - | 400 | - | ns |
| $T_{MTxD}$ | MTXCLK rising edge to TXEN, TXD | 5 | - | 25 | ns |

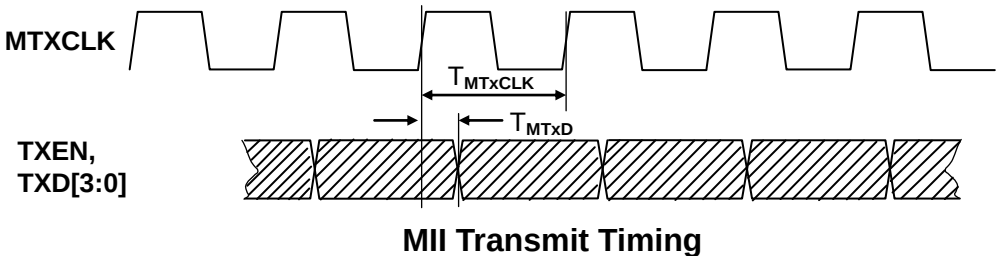



Figure 33 MII Transmit Timing

## 8.6 EEPROM Timing

### 8.6.1 Data read cycle

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{SCL}$ | Receive clock period | - | 12800 | - | ns |
| $T_{sSCL}$ | EE_DAT to EE_CLK setup time | 20 | - | - | ns |
| $T_{hSCL}$ | EE_DAT to EE_CLK hold time | 20 | - | - | ns |

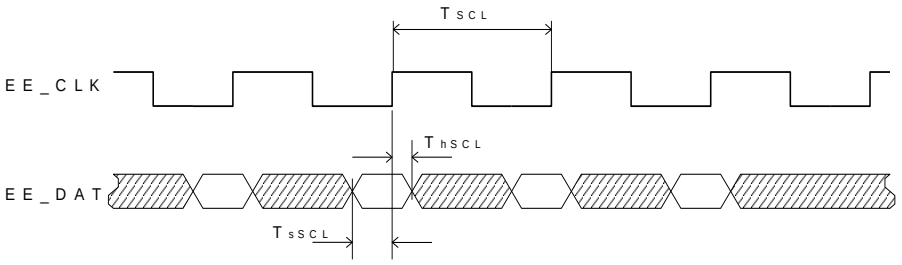



Figure 34 Data read cycle

### 8.6.2 Command cycle

| Symbol | Description | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $T_{SCL}$ | Transmit clock period | - | 12800 | - | ns |
| $T_{dSCL}$ | EE_CLK falling edge to EE_DAT | - | - | 3200 | ns |

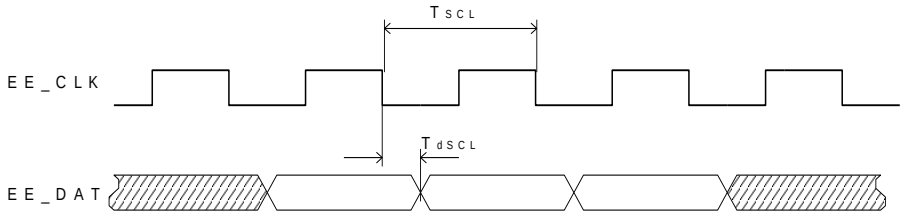



Figure 35 Command cycle

# 9 DC Characteristics

| PARAMETER | SYMBOL | MIN. | TYP. | MAX. | UNIT |
|---|---|---|---|---|---|
| Output low voltage | $V_{OL}$ | --- | --- | 0.1xVDD33 | V |
| Output high voltage | $V_{OH}$ | 0.9xVDD33 | --- | --- | V |
| VDD33 supply current $I_{VDD33}$ | 100M+2G | --- | 1 | --- | mA |
| | 10M+2G | --- | 1 | --- | |
| | IDLE | --- | 1 | --- | |
| AVDD33 supply current $I_{AVDD33}$ | 100M+2G | --- | 183 | --- | mA |
| | 10M+2G | --- | 220 | --- | |
| | IDLE | --- | 55 | --- | |
| DVDD10 supply current $I_{DVDD10}$ | 100M+2G | --- | 517 | --- | mA |
| | 10M+2G | --- | 496 | --- | |
| | IDLE | --- | 416 | --- | |
| AVDD10 supply current $I_{AVDD10}$ | 100M+2G | --- | 50 | --- | mA |
| | 10M+2G | --- | 30 | --- | |
| | IDLE | --- | 30 | --- | |
| VDDIOR supply current $I_{VDDIOR}$ | VDDIOR=2.0V | --- | 15 | --- | mA |
| | VDDIOR=2.5V | --- | 20 | --- | |
| | VDDIOR=3.3V | --- | 30 | --- | |
| VDDIOP supply current $I_{VDDIOP}$ | VDDIOP =2.5V | --- | 10 | --- | mA |
| | VDDIOP =3.3V | --- | 20 | --- | |
| VDD33 supply voltage | $V_{VDD33}$ | 2.97 | 3.3 | 3.63 | V |
| AVDD33 supply voltage | $V_{AVDD33}$ | 2.97 | 3.3 | 3.63 | V |
| DVDD10 supply voltage | $V_{DVDD10}$ | 1.1 | 1.15 | 1.2 | V |
| AVDD10 supply voltage | $V_{AVDD10}$ | 1.1 | 1.15 | 1.2 | V |
| VDDIOR supply voltage | VDDIOR=2.0V | 1.9 | 2.0 | 2.1 | V |
| | VDDIOR=2.5V | 2.38 | 2.5 | 2.63 | |
| | VDDIOR=3.3V | 2.97 | 3.3 | 3.63 | |
| VDDIOP supply voltage | VDDIOP=2.5V | 2.38 | 2.5 | 2.63 | V |
| | VDDIOP=3.3V | 2.97 | 3.3 | 3.63 | |
| RESET Threshold voltage | | 0.4*VDDIOP | --- | 0.6*VDDIOP | V |
| X1 Clock Input Low Voltage (3.3V operation) | $V_{ILX1}$ | --- | --- | 0.6 | V |
| X1 Clock Input High Voltage (3.3V operation) | $V_{IHX1}$ | 1.5 | --- | --- | V |
| Pull-up/down resistor | $R_{PUD}$ | 51 | --- | 127 | KΩ |

| | | | | | | |
|---|---|---|---|---|---|---|
| RGMII | | | | | | |
| Input Low to High | VDDIOR =2.0V | $V_{IH}$ | 1.2 | --- | --- | V |
| Input High to Low | | $V_{IL}$ | --- | --- | 0.8 | |
| Input Low to High | VDDIOR =2.5V | $V_{IH}$ | 1.5 | --- | --- | |
| Input High to Low | | $V_{IL}$ | --- | --- | 1 | |
| Input Low to High | VDDIOR =3.3V | $V_{IH}$ | 1.98 | --- | --- | |
| Input High to Low | | $V_{IL}$ | --- | --- | 1.32 | |
| RGMII/MII | | | | | | |
| Input Low to High | VDDIOP =2.5V | $V_{IH}$ | 1.5 | --- | --- | V |
| Input High to Low | | $V_{IL}$ | --- | --- | 1 | |
| Input Low to High | VDDIOP =3.3V | $V_{IH}$ | 1.98 | --- | --- | |
| Input High to Low | | $V_{IL}$ | --- | --- | 1.32 | |

# 10 Serial Transmitter/Receiver DC characteristic

**Transmitter** DC Characteristics

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $V_{OD}$ | Output Voltage Swing(single-ended) | 200 | 300 | 400 | $mV_{p-p}$ |
| $R_O$ | Output Impedance(single-ended) | 42.5 | 50 | 57.5 | Ωs |

**Receiver DC Characteristics**

| Symbol | Parameter | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $V_{RCOM}$ | Common mode voltage(Supplied by IC itself) | 1.4 | 1.6 | 1.8 | V |
| $V_{IDTH}$ | Input Differential Threshold | 200 | | | $mV_{p-p}$ |
| $R_{IN}$ | Receiver 100Ω Differential Input Impedance | 85 | 100 | 115 | Ω |

# 11 Power Consumption

| Symbol | Parameter | Power Consumption (@25°C) | Units |
|---|---|---|---|
| $P_{10MF}$ | 8-Port 10M Full + 2G active | 1.476 | W |
| $P_{100MF}$ | 8-Port 100M Full + 2G active | 1.461 | W |
| $P_{LPI}$ | IEEE802.3az low power idle mode | 1.369 | W |
| $P_{LDPS}$ | Link down power saving mode | 0.782 | W |

# 12 Crystal & X1 Clock Input Specifications

## 12.1 Crystal Specifications

**Table 35 Crystal Specifications**

| Item | Parameter | Range |
|---|---|---|
| 1 | Nominal Frequency | 25.000 MHz |
| 2 | Oscillation Mode | Fundamental Mode |
| 3 | Frequency Tolerance at 25°C | +/- 50 ppm |
| 4 | Temperature Characteristics | +/- 50 ppm |
| 5 | Operating Temperature Range | -40°C ~ +85°C |
| 6 | Equivalent Series Resistance | 40 ohm Max. |
| 7 | Drive Level | 100 $\mu$W |
| 8 | Load Capacitance | 20 pF |
| 9 | Shunt Capacitance | 7 pF Max |
| 10 | Insulation Resistance | Mega ohm Min./DC 100V |
| 11 | Aging Rate A Year | +/- 5 ppm/year |

## 12.2 X1 Clock Input Specifications

**Table 36 X1 Clock Input Specifications**

| Item | Parameter | Range |
|---|---|---|
| 1 | Frequency | 25.000 MHz |
| 2 | Duty cycle | 40~60% |
| 3 | Rise time | Max: 4 ns |
| 4 | Fall time | Max: 4 ns |
| 5 | Max voltage | 3.63V |
| 6 | Jitter | 300 ps |

## 13 Thermal Data

The junction temperature of chip (Tj) is a well-known key factor of IC characteristics and life time. There are there thermal resistance coefficients, JA, ψJT, and Jc , which are defined in JEDEC 51-2 and 51-6, will be provided to estimate Tj. If the Tj is over maximum limit, it is necessary to disperse heat by extra dissipation device, such as heat sink and electrical fan.

Thermal resistance represents the capability of an IC package to carry out the heat inside an IC chip. It is a complex function of package structure, material property, input power, environment variables such as air flow speed and PCB layers. The major thermal dissipation paths can be illustrated as following.

**Table 37 Operation Range**

| Parameter | Min | Typical | Max | unit |
|---|---|---|---|---|
| Tj | -40 | - | +125 | ℃ |
| Ta | -40 | +25 | +85 | ℃ |

**Table 38 Thermal Resistance**

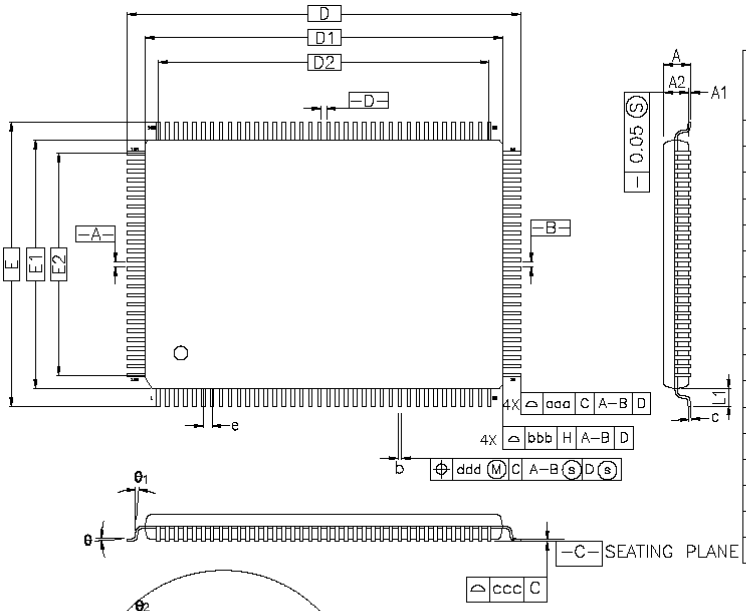
| Parameter | Condition | Value | Unit |
|---|---|---|---|
| $\Theta_{JA}$ | 0 ft/s Air flow<br>2 layers PCB | 46.5 | ℃/W |
| | 0 ft/s Air flow<br>4 layers PCB | 18.0 | |
| | IC Plus 2 layers<br>210x130mm$^2$ PCB | 34.2 | |
| $\Theta_{JC}$ | 0 ft/s Air flow<br>2 layers PCB | 8.4 | ℃/W |
| | 0 ft/s Air flow<br>4 layers PCB | 6.0 | |
| | IC Plus 2 layers<br>210x130mm$^2$ PCB | 8.4 | |
| $\Psi_{JT}$ | 0 ft/s Air flow<br>2 layers PCB | 5.9 | ℃/W |
| | 0 ft/s Air flow<br>4 layers PCB | 3.0 | |
| | IC Plus 2 layers<br>210x130mm$^2$ PCB | 5.9 | |

## 14 Order Information

| Part No. | Package | Notice |
|---|---|---|
| IP1810I | 128 pin LQFP | - |

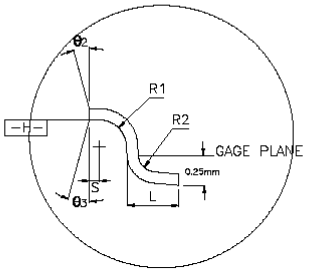
# 15 Package Outline

128pin LQFP Outline Dimensions

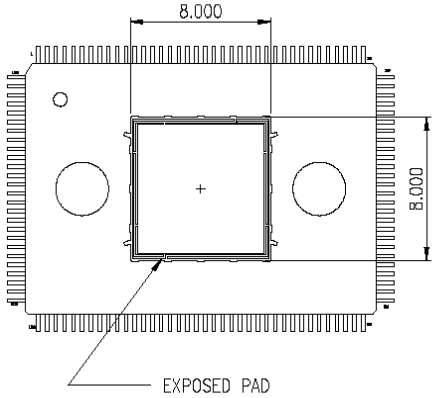



COTROL DIMENSIONS ARE IN MILLIMETERS.

| SYMBOL | MILLIMETER | | | INCH | | |
|---|---|---|---|---|---|---|
| | MIN. | NOM. | MAX. | MIN. | NOM. | MAX. |
| A | — | — | 1.60 | — | — | 0.063 |
| A1 | 0.05 | — | 0.15 | 0.002 | — | 0.006 |
| A2 | 1.35 | 1.40 | 1.45 | 0.053 | 0.055 | 0.057 |
| D | 22.00 BSC. | | | 0.866 BSC. | | |
| D1 | 20.00 BSC. | | | 0.787 BSC. | | |
| E | 16.00 BSC. | | | 0.630 BSC. | | |
| E1 | 14.00 BSC. | | | 0.551 BSC. | | |
| R2 | 0.08 | — | 0.20 | 0.003 | — | 0.008 |
| R1 | 0.08 | — | — | 0.003 | — | — |
| θ | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| θ1 | 0° | — | — | 0° | — | — |
| θ2 | 11° | 12° | 13° | 11° | 12° | 13° |
| θ3 | 11° | 12° | 13° | 11° | 12° | 13° |
| c | 0.09 | — | 0.20 | 0.004 | — | 0.008 |
| L | 0.45 | 0.60 | 0.75 | 0.018 | 0.024 | 0.030 |
| L1 | 1.00 REF | | | 0.039 REF | | |
| S | 0.20 | — | — | 0.008 | — | — |




TOP VIEW




BOTTOM VIEW

| SYMBOL | 128L | | | | | |
|---|---|---|---|---|---|---|
| | MILLIMETER | | | INCH | | |
| | MIN. | NOM. | MAX. | MIN. | NOM. | MAX. |
| b | 0.17 | 0.20 | 0.27 | 0.007 | 0.008 | 0.011 |
| e | 0.50 BSC. | | | 0.020 BSC. | | |
| D2 | 18.50 | | | 0.728 | | |
| E2 | 12.50 | | | 0.492 | | |
| TOLERANCES OF FORM AND POSITION | | | | | | |
| aaa | 0.20 | | | 0.008 | | |
| bbb | 0.20 | | | 0.008 | | |
| ccc | 0.08 | | | 0.003 | | |
| ddd | 0.08 | | | 0.003 | | |

NOTES :

1. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25mm PER SIDE. D1 AND E1 ARE MAXIMUN PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.

2. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM b DIMENSION BY MORE THAN 0.08mm.
   DAMBAR CAN NOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND AN ADJACENT LEAD IS 0.07mm FOR 0.4mm and 0.5mm PITCH PACKAGES.