TMS320DM647/DM648 DSP
3 Port Switch (3PSW) Ethernet Subsystem

User's Guide

Literature Number: SPRUF57B
July 2009

Contents

List of Figures

List of Tables

Read This First

About This Manual

This document describes the operation of the 3 port switch (3PSW) ethernet subsystem in the TMS320DM647/DM648 Digital Signal Processor (DSP). The 3 port switch gigabit ethernet subsystem provides ethernet packet communication and can be configured as an ethernet switch TMS320DM647/DM648. It provides the serial gigabit media independent interface (SGMII), the management data input output (MDIO) for physical layer device (PHY) management.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Note: Acronyms 3PSW, CPSW, CPSW_3G, and 3pGSw are interchangeable and all refer to the 3 port gigabit switch.

Related Documentation From Texas Instruments

The following documents describe the TMS320DM647/DM648 Digital Signal Processor (DSP). Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

SPRU732 — ***TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide*** describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C64x and TMS320C64x+ digital signal processors (DSPs) of the TMS320C6000 DSP family. The C64x/C64x+ DSP generation comprises fixed-point devices in the C6000 DSP platform. The C64x+ DSP is an enhancement of the C64x DSP with added functionality and an expanded instruction set.

SPRUEK5 —***TMS320DM647/DM648 DSP DDR2 Memory Controller User's Guide*** describes the DDR2 memory controller in the TMS320DM647/DM648 Digital Signal Processor (DSP). The DDR2/mDDR memory controller is used to interface with JESD79D-2A standard compliant DDR2 SDRAM devices and standard Mobile DDR SDRAM devices.

SPRUEK6 —***TMS320DM647/DM648 DSP External Memory Interface (EMIF) User's Guide*** describes the operation of the asynchronous external memory interface (EMIF) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The EMIF supports a glueless interface to a variety of external devices.

SPRUEK7 ***—TMS320DM647/DM648 DSP General-Purpose Input/Output (GPIO) User's Guide*** describes the general-purpose input/output (GPIO) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). The GPIO peripheral provides dedicated general-purpose pins that can be configured as either inputs or outputs. When configured as an input, you can detect the state of the input by reading the state of an internal register. When configured as an output, you can write to an internal register to control the state driven on the output pin.

SPRUEK8 ***—TMS320DM647/DM648 DSP Inter-Integrated Circuit (I2C) Module User's Guide*** describes the inter-integrated circuit (I2C) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). The I2C peripheral provides an interface between the DSP and other devices compliant with the I2C-bus specification and connected by way of an I2C-bus. External components attached to this 2-wire serial bus can transmit and receive up to 8-bit wide data to and from the DSP through the I2C peripheral. This document assumes the reader is familiar with the I2C-bus specification.

SPRUEL0 ***—TMS320DM647/DM648 DSP 64-Bit Timer User's Guide*** describes the operation of the 64-bit timer in the TMS320DM647/DM648 Digital Signal Processor (DSP). The timer can be configured as a general-purpose 64-bit timer or dual general-purpose 32-bit timers.

SPRUEL1 ***—TMS320DM647/DM648 DSP Multichannel Audio Serial Port (McASP) User's Guide*** describes the multichannel audio serial port (McASP) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The McASP functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, Inter-Integrated Sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT).

SPRUEL2 ***—TMS320DM647/DM648 DSP Enhanced DMA (EDMA) Controller User's Guide*** describes the operation of the enhanced direct memory access (EDMA3) controller in the TMS320DM647/DM648 Digital Signal Processor (DSP). The EDMA3 controller’s primary purpose is to service user-programmed data transfers between two memory-mapped slave endpoints on the DSP.

SPRUEL4 ***—TMS320DM647/DM648 DSP Peripheral Component Interconnect (PCI) User's Guide*** describes the peripheral component interconnect (PCI) port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The PCI port supports connection of the C642x DSP to a PCI host via the integrated PCI master/slave bus interface. The PCI port interfaces to the DSP via the enhanced DMA (EDMA) controller. This architecture allows for both PCI master and slave transactions, while keeping the EDMA channel resources available for other applications.

SPRUEL5 ***—TMS320DM647/DM648 DSP Host Port Interface (UHPI) User's Guide*** describes the host port interface (HPI) in the TMS320DM647/DM648 Digital Signal Processor (DSP). The HPI is a parallel port through which a host processor can directly access the CPU memory space. The host device functions as a master to the interface, which increases ease of access. The host and CPU can exchange information via internal or external memory. The host also has direct access to memory-mapped peripherals. Connectivity to the CPU memory space is provided through the enhanced direct memory access (EDMA) controller.

SPRUEL8 ***—TMS320DM647/DM648 DSP Universal Asynchronous Receiver/Transmitter (UART) User's Guide*** describes the universal asynchronous receiver/transmitter (UART) peripheral in the TMS320DM647/DM648 Digital Signal Processor (DSP). The UART peripheral performs serial-to-parallel conversion on data received from a peripheral device, and parallel-to-serial conversion on data received from the CPU.

SPRUEL9 ***—TMS320DM647/DM648 DSP VLYNQ Port User's Guide*** describes the VLYNQ port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The VLYNQ port is a high-speed point-to-point serial interface for connecting to host processors and other VLYNQ compatible devices. It is a full-duplex serial bus where transmit and receive operations occur separately and simultaneously without interference.

SPRUEM1 **—*TMS320DM647/DM648 DSP Video Port/VCXO Interpolated Control (VIC) Port User's Guide*** discusses the video port and VCXO interpolated control (VIC) port in the TMS320DM647/DM648 Digital Signal Processor (DSP). The video port can operate as a video capture port, video display port, or transport channel interface (TCI) capture port. The VIC port provides single-bit interpolated VCXO control with resolution from 9 bits to up to 16 bits. When the video port is used in TCI mode, the VIC port is used to control the system clock, VCXO, for MPEG transport channel.

SPRUEM2 **—*TMS320DM647/DM648 DSP Serial Port Interface (SPI) User's Guide*** discusses the Serial Port Interface (SPI) in the TMS320DM647/DM648 Digital Signal Processor (DSP). This reference guide provides the specifications for a 16-bit configurable, synchronous serial peripheral interface. The SPI is a programmable-length shift register, used for high speed communication between external peripherals or other DSPs.

SPRUEU6 **—TMS320DM647/DM648 DSP Subsystem User's Guide** describes the subsystem in the TMS320DM647/DM648 Digital Signal Processor (DSP). The subsystem is responsible for performing digital signal processing for digital media applications. The subsystem acts as the overall system controller, responsible for handling many system functions such as system-level initialization, configuration, user interface, user command execution, connectivity functions, and overall system control.

SPRUF57 **—TMS320DM647/DM648 DSP 3 Port Switch (3PSW) Ethernet Subsystem User's Guide** describes the operation of the 3 port switch (3PSW) ethernet subsystem in the TMS320DM647/DM648 Digital Signal Processor (DSP). The 3 port switch gigabit ethernet subsystem provides ethernet packet communication and can be configured as an ethernet switch (DM648 only). It provides the serial gigabit media independent interface (SGMII), the management data input output (MDIO) for physical layer device (PHY) management.

3 Port Switch (3PSW) Ethernet Subsystem

1 Introduction

This document describes the 3 port switch (3PSW) ethernet subsystem in the device. The 3 port switch gigabit ethernet subsystem provides ethernet packet communication and can be configured as an ethernet switchTMS320DM647/DM648. It provides the serial gigabit media independent interface (SGMII), the management data input output (MDIO) for physical layer device (PHY) management.

Note: Acronyms 3PSW, CPSW, CPSW_3G, and 3pGSw are interchangeable and all refer to the 3 port gigabit switch.

1.1 Features

The 3 port switch (3PSW) ethernet subsystem provides the following functions:

- Ethernet communication/routing by way of two dedicated 10/100/1000 ports with SGMII interfaces:
  - Wire-rate switching (802.1d), non-blocking switch fabric.
  - Four priority levels of QoS TX support (802.1p) in hardware.
  - Programmable interrupt pacing on RX/TX plus interrupt threshold on RX.
  - Supports forwarding frame sizes of 64-2020 bytes.
- Address Lookup:
  - 1024 total address lookup engine (ALE) entries of VLANs and/or Media Access Control (MAC) addresses.
  - L2 address lock and L2 filtering support.
  - Multicast/broadcast filtering and forwarding state control.
  - Receive-based or destination-based multicast and broadcast rate limits.
  - MAC address blocking.
  - Source port locking.
  - OUI (Vendor ID) host accept/deny feature.
  - Host controlled time-based aging.
  - MAC authentication (802.1x).
  - Remapping of priority level of VLAN or ports.
  - Multiple spanning tree support (spanning tree per VLAN).
- VLAN support:
  - 802.1Q compliant:
    - Auto add port VLAN for untagged frames on ingress.
    - Auto VLAN removal on egress and auto pad to minimum frame size.
  - Flow control (IEEE 802.3x).
  - Programmable priority escalation to specify delivery of lower priority level packets in the event of over-subscribed TX high priority traffic.
  - Host pass CRC mode (enables CRC protection through host).
  - Write-protect option for Ethernet module registers (3PGSW, CPPI RAM, MDIO, SGMII0, SGMII1, control).
  - Ethernet statistics:
    - EtherStats and 802.3 Stats RMON statistics gathering (shared).

- Programmable statistics interrupt mask when a statistic is above one half its 32-bit value.

– MDIO module for physical layer device (PHY) management.

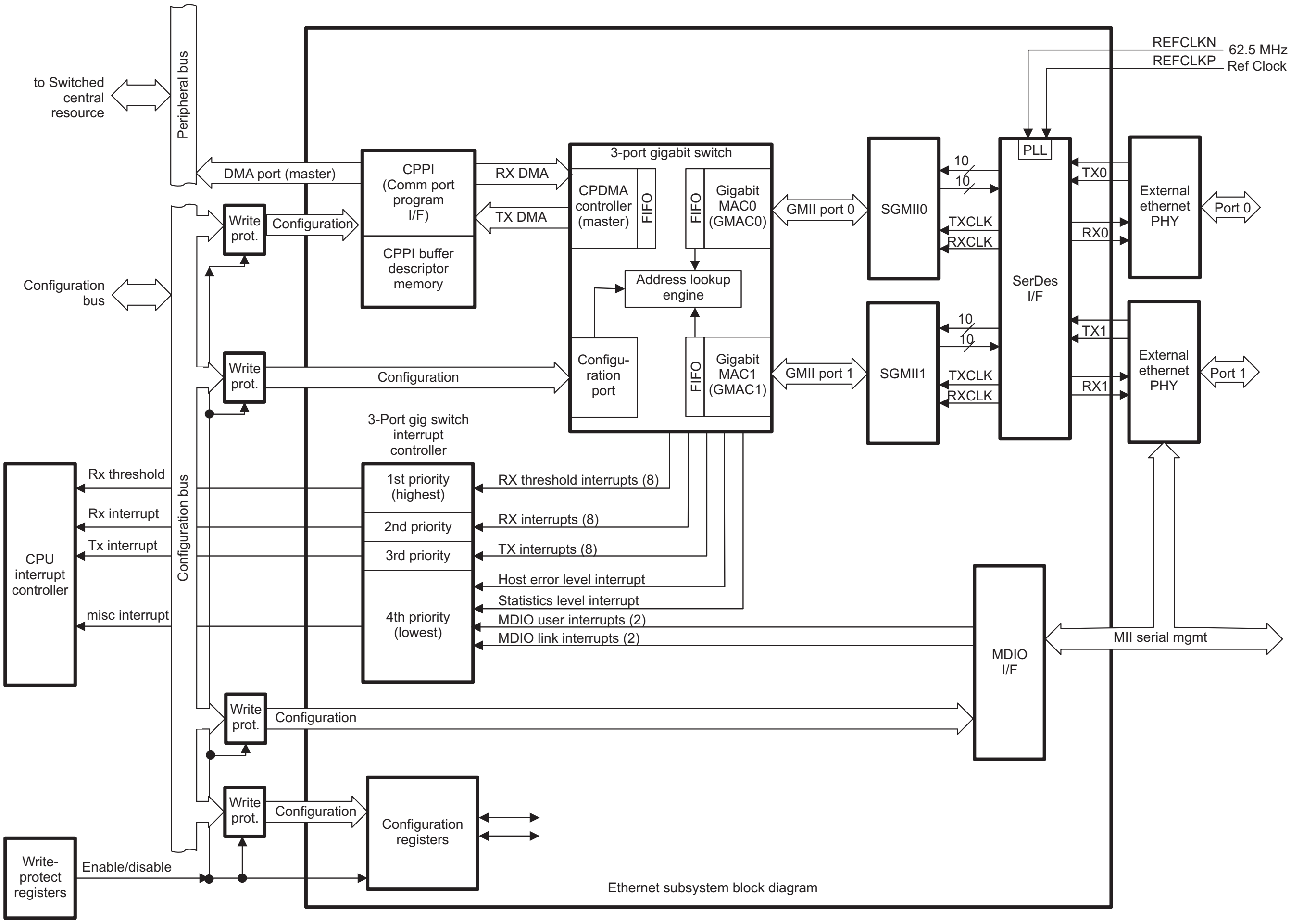
– SGMII gigabit current mode logic (CML) differential Serializer/Deserializer (SerDes) I/O receiver/transmitters.

- Adaptive active equalization for superior data dependent jitter tolerance in the presence of a lossy channel.
- Loss of signal detector with programmable threshold levels in receive channels.
- Integrated receiver and transmitter termination.
- IEEE 802.3 gigabit Ethernet conformant.

1.2 Block Diagram

The 3 port switch (3PSW) ethernet subsystem functional block diagram is shown in Figure 1.

Figure 1. 3PSW Block Diagram



Note: The 8k bytes of Ethernet Subsystem CPPI RAM begin at address 0x02c82000 and end at 0x02c83FFF from the 3PSW perspective. The buffer descriptors programmed to access the CPPI RAM memory should use the address range from 0x02c82000.

1.3 Signal Summary

The 3 port switch (3PSW) ethernet subsystem signals are described in Table 1.

Table 1. 3PSW Signal Descriptions

Signal	Description
SGMII0RXN	Differential SGMII port 0 RX input (negative)
SGMII0RXP	Differential SGMII port 0 RX input (positive)
SGMII0TXN	Differential SGMII port 0 TX output (negative)
SGMII0TXP	Differential SGMII port 0 TX output (positive)
SGMII1RXN	Differential SGMII port 1 RX input (negative)
SGMII1RXP	Differential SGMII port 1 RX input (positive)
SGMII1TXN	Differential SGMII port 1 TX output (negative)
SGMII1TXP	Differential SGMII port 1 TX output (positive)
MDCLK	MDIO serial clock (MDCLK) for GMII
MDIO	MDIO serial data (MDIO) for GMII
REFCLKP	Differential clock input (positive)
REFCLKN	Differential clock input (negative)

1.4 Clocks

The 3 port switch (3PSW) ethernet subsystem clock is derived from both the REFCLKP/N and SerDes PLL programming. The SerDes PLL configuration is done through the SerDes PLL configuration register (CFGPLL) in the chip level register space. (See Section 4.1.) To enable the internal PLL, the ENPLL bit in the PLL configuration register must be set to 1. After setting this bit, it is necessary to allow 1μs + 200 REFCLKP/N cycles for it to stabilize. When ENPLL is 0, the PLL will be fully powered down.

During normal operation the integrated PLL uses REFCLKP/N to generate a higher frequency clock from which the bit rate can be derived. REFCLKP/N should be either in the range of 50-62.5 Mhz or 125Mhz. The clock generated by the PLL will be between 10, 20 or 25 times the frequency of REFCLKP/N, according to the multiplication factor selected via the MPY field. The multiplication factor is based on the MPY field value shown below.

MPY Field Value	Multiplication Factor
0000-1000	Reserved
0101	10x
0110-1000	Reserved
1001	20x
1010	25x
1011 - 1111	Reserved

$$\text{REFCLKP/N}_{\text{Freq}} = \text{LINERATE} \times \text{RATESCALE} / \text{MPY}$$

The RATESCALE is programmed in the RATE field in the SerDes transmitter and receiver configuration registers. The Possible values of RATE field are shown below.

RATE	RATESCALE	Meaning
00	Full Rate	Two data samples are taken per one PLL output clock cycle.
01	Half Rate	One data sample is taken per one PLL output clock cycle.
10	Quarter Rate	One data sample is taken per every two PLL output clock cycles.
11	Reserved	

The following table summarizes the possible suggested combinations between MPY, RATE, REFCLKP/ N, and the corresponding Line Rate for the BUSWIDTH of 10.

REFCLKP/N	Multiplication Factor (MPY)	PLL Frequency (REFCLK × MPY)	RATE	RXBCLK/TXBCLK[1]	Line Rate[2]
125 MHz	10x (0101)	1250 MHz	Half (01b)	125 MHZ	1250 Mbps
125 MHz	10x (0101)	1250 MHz	Quarter (10b)	62.5 MHz	625 Mbps
62.5 MHz	20x (1001)	1250 MHz	Half (01b)	125 MHz	1250 Mbps
62.5 MHz	20x (1001)	1250 MHz	Quarter (10b)	62.5 MHz	625 Mbps
50 MHz	25x (1010)	1250 MHz	Half (01b)	125 MHz	1250 Mbps
50 MHz	25x (1010)	1250 MHz	Quarter (10b)	62.5 MHz	625 Mbps

[1] =PLL Freq/5, when RATE=Full (00b)
=PLL Freq/10, when RATE=Half (01b)
=PLL Freq/20, when RATE=Quarter (10b)

[2] PLL freq × 2 for Full rate
PLL freq × 1 for Half rate
PLL freq × 0.5 for Quarter rate

2 Components

2.1 Communication Processor Gigabit Ethernet Switch (3pGSw)

The 3pGSw contains the following submodules.

2.1.1 Media Independent Interface (GMII)

The 3pGSw has two GMAC submodules. Each GMAC has a single GMII interface. The two GMAC submodules are ports 0 and 1.

2.1.2 CPDMA RX and TX Interfaces

The CPDMA submodule is a CPPI 3.0 compliant packet DMA transfer controller. The CPPI 3.0 interface is port 2.

2.1.2.1 CPPI 3.0 Host Software Interface

Host software sends and receives network frames via the CPPI 3.0 compliant (CPDMA) host interface. The host interface includes module registers and host memory data structures. The host memory data structures are CPPI 3.0 buffer descriptors and data buffers. Buffer descriptors are data structures that contain information about a single data buffer. Buffer descriptors may be linked together to describe frames or queues of frames for data transmission and free buffer queues that are available for received data.

2.1.3 Statistics Interface

For list of statistics interface registers, see Section 5.76.

2.1.4 Embedded Memories

Memory Type Description Number of Instantiations:

- Single port 32-word by 32-bit RAM 1 (STATERAM).
- Two port 36-word by 32-bit RAM 1 (STATISTICS).
- Single port 64-word by 64-bit RAM 17 (ALE).

2.1.5 Gigabit Ethernet MAC Sliver (GMAC)

Features:

- Synchronous 10/100/1000 Mbit operation.
- GMII Interface.
- Hardware Error handling including CRC.
- Full Duplex Gigabit operation (half duplex gigabit is not supported).
- EtherStats and 802.3 Stats RMON statistics gathering support for external statistics collection module.
- Transmit CRC generation selectable on a per channel basis.
- Emulation Support.
- VLAN Aware Mode Support.

2.2 Serial Gigabit Media Independent Interface (SGMII)

Features:

- SGMII interface to GMAC GMII interface.
- 8B/10B encoding and decoding.
- SGMII mode supports SGMII auto-negotiation (with 1.6ms link timer).
- = 802.3z clause 37 auto-negotiation (with 10ms link timer).
- SGMII mode supports direct SGMII to SGMII connection (master/slave configurable via MMR) with auto-negotiation or with forced link.
- SGMII support through Serializer/Deserializer (SerDes) Technology.
- SerDes configuration is programmable through the SGMII peripheral bus slave interface.
- SerDes supports power down.

The SGMII receive (RX) interface converts the encoded receive input from the SerDes into the required GMAC GMII signals. The SGMII transmit (TX) interface converts the GMAC GMII input data in to the required encoded transmit outputs.

Please refer to Section 7 for SGMII register definitions and memory map.

2.3 MDIO

The MII management I/F module implements the 802.3 serial management interface to interrogate and control two ethernet PHYs simultaneously using a shared two-wire bus. Figure 1 shows a device with two MACs, each connected to a PHY, being managed by the MII interface module using a shared bus.

Table 2 and Table 3 show the read and write frame format of the 32-bit MII management interface frames.

Table 2. MDIO Read Frame Format

Pre-amble	Start Delimiter	Operation Code	PHY Address	Register Address	Turnaround	Data
FFFF FFFFh	01	10	AAAAA	RRRRR	20	DDDD.DDDD.DDDD.DDDD

Table 3. MDIO Write Frame Format

Pre-amble	Start Delimiter	Operation Code	PHY Address	Register Address	Turnaround	Data
FFFF FFFFh	01	01	AAAAA	RRRRR	10	DDDD.DDDD.DDDD.DDDD

The default or idle state of the two wire serial interface is at logic one. All tri-state drivers should be disabled and the PHY's pull-up resistor will pull the MDIO line to logic one. Prior to initiating any other transaction, the station management entity shall send a preamble sequence of 32 contiguous logic one bits on the MDIO line with 32 corresponding cycles on MDCLK to provide the PHY with a pattern that it can use to establish synchronization. A PHY shall observe a sequence of 32 contiguous logic one bits on MDIO with 32 corresponding MDCLK cycles before it responds to any other transaction.

Table header descriptions are:

- *Preamble* - The start of a frame is indicated by a preamble. A preamble consists of a sequence of 32 contiguous bits all of which are a 1. This sequence provides the PHY with a pattern it can use to establish synchronization.
- *Start Delimiter* - The preamble is followed by the start delimiter which is indicated by a 01 pattern. The pattern assures transitions from the default logic one state to zero and back to one.
- *Operation Code* - The operation code for a read is 10, while the operation code for a write is 01.
- *PHY Address* - The PHY address is 5 bits allowing 32 unique values. The first bit transmitted is the MSB of the PHY address.
- *Register Address* - The register address is 5 bits allowing 32 registers to be addressed within each PHY. Refer to the 10/100 PHY address map for addresses of individual registers.

- *Turnaround* - An idle bit time during which no device actively drives the MDIO signal shall be inserted between the register address field and the data field of a read frame in order to avoid contention. During a read frame, the PHY shall drive a zero bit onto MDIO for the first bit time following the idle bit and preceding the data field. During a write frame, this field shall consist of a one bit followed by a zero bit.
- *Data* - The data field is 16 bits. The first bit transmitted and received is the MSB of the data word.

2.4 Serializer/Deserializer Module (SerDes)

The serializer/deserializer (SerDes) converts parallel data to serial data and vice-versa. The transmitter section is a parallel-to-serial converter, and the receiver section is a serial-to-parallel converter.

For a list of chip level SerDes registers that are used to configure the PLL, receiver, and transmitter, refer to Section 4.

2.5 Interrupts

The 3 Port Gigabit Switch Subsystem generates 4 interrupt events.

2.5.1 Receive Packet Completion Pulse Interrupt (RX_PULSE)

The RX_PULSE interrupt is a pulse interrupt selected from the 3pGSw RX_PEND [7:0] interrupts. The receive DMA controller has eight channels with each channel having a corresponding (RX_PEND[7:0]).

The following steps will enable the receive packet completion interrupt.

- Enable the required channel interrupts of the DMA engine by setting 1 to the appropriate bit in the RX_INTMASK_SET register (see Section 5.64).
- The receive completion interrupt(s) to be routed to RX_PULSE is selected by setting one or more bits in the receive interrupt enable register RX_EN (see Section 6.6). The masked interrupt status can be read in the address location of RX_STAT bit in the receive status register.

When the 3pGSW completes a packet reception, the subsystem issues an interrupt to the CPU by writing the packet's last buffer descriptor address to the appropriate channel queue's receive completion pointer located in the state RAM block. The interrupt is generated by the write when enabled by the interrupt mask, regardless of the value written.

Upon interrupt reception, the CPU processes one or more packets from the buffer chain and then acknowledges one or more interrupt(s) by writing the address of the last buffer descriptor processed to the queue's associated receive completion pointer (RX*n*_CP) in the receive DMA state RAM.

Upon reception of an interrupt, software should perform the following:

- Read the RX_STAT bit address location to determine which channel(s) caused the interrupt.
- Process received packets for the interrupting channel(s).
- Write the 3pGSw completion pointer(s) (RX*n*_CP). The data written by the host (buffer descriptor address of the last processed buffer) is compared to the data in the register written by the subsystem (address of last buffer descriptor used by the subsystem). If the two values are not equal (which means that the 3pGSW has received more packets than the CPU has processed), the receive packet completion interrupt signal remains asserted. If the two values are equal (which means that the host has processed all packets that the system has received), the pending interrupt is de-asserted. The value that the 3pGSW is expecting is found by reading the receive channel*n* completion pointer register (RX*n*_CP).
- Write the value 1h to the CPDMA_EOI_VECTOR register.

To disable the interrupt:

- The eight channel interrupts may be individually disabled by writing to 1 the appropriate bit in the RX_INTMASK_CLEAR (see Section 5.65).
- The receive completion pulse interrupt could be disabled by clearing to 0 all the bits of the RX_EN. The software could still poll for the RX_INTSTAT_RAW and RX_INTSTAT_MASKED registers if the corresponding interrupts are enabled.

2.5.2 Transmit Packet Completion Pulse Interrupt (TX_PULSE)

The TX_PULSE interrupt is a pulse interrupt selected from the 3pGSw TX_PEND [7:0] interrupts. The transmit DMA controller has eight channels with each channel having a corresponding (TX_PEND[7:0]).

To enable the transmit packet completion interrupt:

- Enable the required channel interrupts of the DMA engine by setting 1 to the appropriate bit in the TX_INTMASK_SET register (see Section 5.58).
- The transmit completion interrupt(s) to be routed to TX_PULSE is selected by setting one or more bits in the transmit interrupt enable register TX_EN (see Section 6.6). The masked interrupt status can be read in the address location of TX_STAT bit in the transmit status register.

When the 3pGSW completes the transmission of a packet, the 3pGSW subsystem issues an interrupt to the CPU by writing the packet's last buffer descriptor address to the appropriate channel queue's transmit completion pointer located in the state RAM block. The interrupt is generated by the write when enabled by the interrupt mask, regardless of the value written.

Upon reception of an interrupt, software should perform the following:

- Read the TX_STAT bit address location to determine which channel(s) caused the interrupt
- Process received packets for the interrupting channel(s).
- Write the 3pGSw completion pointer(s) (TX*n*_CP). The data written by the host (buffer descriptor address of the last processed buffer) is compared to the data in the register written by the 3pGSW (address of last buffer descriptor used by the 3pGSW). If the two values are not equal (which means that the 3pGSW has transmitted more packets than the CPU has processed), the transmit packet completion interrupt signal remains asserted. If the two values are equal (which means that the host has processed all packets that the subsystem has transferred), the pending interrupt is cleared. The value that the 3pGSW is expecting is found by reading the transmit channel*n* completion pointer register (TX*n*_CP).
- Write the 2h to the CPDMA_EOI_VECTOR register.

To disable the interrupt:

- The eight channel interrupts may be individually disabled by writing to 1 the appropriate bit in the TX_INTMASK_CLEAR (see Section 5.59).
- The receive completion pulse interrupt could be disabled by clearing to 0 all the bits of the TX_EN. The software could still poll for the TX_INTSTAT_RAW and TX_INTSTAT_MASKED registers if the corresponding interrupts are enabled.

2.5.3 Receive Threshold Pulse Interrupt (RX_THRESH_PULSE)

The RX_THRESH_PULSE interrupt is an immediate (non-paced) pulse interrupt selected from the CPSW_3G RX_THRESH_PEND[7:0] interrupts. The receive DMA controller has eight channels with each channel having a corresponding threshold pulse interrupt (RX_THRESH_PEND [7:0]).

To enable the receive threshold pulse Interrupt:

- Enable the required channel interrupts of the DMA engine by setting 1 to the appropriate bit in the RX_INTMASK_SET register (see Section 5.64).
- The receive threshold interrupt(s) to be routed to RX_THRESH_PULSE is selected by setting one or more bits in the receive threshold interrupt enable register RX_THRESH_EN (see Section 6.5). The masked interrupt status can be read in the address location of RX_THRESH_STAT bit in the receive threshold status register.

The RX_THRESH_PULSE is asserted when enabled when the channel's associated free buffer count RX*n*_FREEBUFFER is less than or equal to the corresponding RX*n*_PENDTHRESH register.

Upon reception of an interrupt, software should perform the following:

- Read the RX_THRESH_STAT bit address location to determine which channel(s) caused the interrupt.
- Process the received packets in order to add more buffers to any channel that is below the threshold value.
- Write the CPSW_3G completion pointer(s).
- Write the value 0h to the CPDMA_EOI_VECTOR register.

The threshold pulse interrupt is an immediate interrupt intended to indicate that software should immediately process packets to preclude an overrun condition from occurring for the particular channels.

To disable the interrupt:

- The eight channel receive threshold interrupts may be individually disabled by writing to 1 the appropriate bit in the RX_INTMASK_CLEAR (see Section 5.65).
- The receive threshold pulse interrupt could be disabled by clearing to Zero the corresponding bits of the RX_THRESH_EN. The software could still poll for the RX_INTSTAT_RAW and INTSTAT_MASKED registers if the corresponding interrupts are enabled.

2.5.4 Miscellaneous Pulse Interrupt (MISC_PULSE)

The MISC_PULSE interrupt is an immediate pulse interrupt selected from the miscellaneous interrupts (STAT_PEND, HOST_PEND, MDIO_LINKINT, MDIO_USERINT).

To enable the miscellaneous pulse interrupt:

- The miscellaneous interrupt(s) is selected by setting one or more bits in the miscellaneous interrupt enable register (MISC_EN).The masked interrupt status can be read in the MISC_STAT bit in the miscellaneous interrupt status register (MISC_STAT) address location.
- The Statistics interrupt is enabled by setting to 1 the STAT_INT_MASK bit in the DMA_INTMASK_SET register.
- The HOST_PEND is enabled by setting to 1 the HOST_ERR_INTMASK in the DMA_INTMASK_SET register.

The STAT_PEND will be issued if enabled when any statistics value is greater than or equal to 0x80000000. The statistics interrupt is removed by writing to decrement any statistics value greater than 8000 0000h.

The host error interrupt (HOST_PEND) will be asserted if enabled when a host error is detected during transmit or receive CPDMA transactions. The host error interrupt is intended for software debug, and is cleared by a warm reset or a system reset. The raw and masked statistics interrupt status can be read by reading the DMA_INTSTAT_RAW and DMA_INTSTAT_MASKED registers, respectively.

The following list shows the transmit host error conditions:

- SOP error
- OWNERSHIP bit not set in SOP buffer
- next buffer descriptor pointer without EOP set to 0
- buffer pointer set to 0
- buffer length set to 0
- packet length error

The receive host error conditions include: the OWNERSHIP bit not set in the input buffer, buffer pointer set to 0, buffer length on non-SOP descriptor set to 0, and SOP buffer length not greater than offset.

The host error interrupt is disabled by clearing to 0 the appropriate bit in the DMA_INTMASK_CLEAR register.

MDIO_LINKINT is set if there is a change in the link state of the PHY corresponding to the address in the PHYADDRMON field of the MDIOUSERPHYSEL*n* register and the corresponding LINKINTENB bit is set. The MDIO_LINKINT event is also captured in the MDIOLINKINTMASKED register.

When the GO bit in the MDIOUSERREGISTER*n* registers transitions from 1 to 0, indicating the completion of a user access, and the corresponding USERINTMASKSET bit in the MDIOUSERINTMASKSET register is set, the MDIO_USERINT signal is asserted 1. The MDIO_USERINT event is also captured in the MDIOUSERINTMASKED register.

Upon reception of an interrupt, software should perform the following:

- Read the MISC_STAT bit address location to determine the source of the interrupt.
- Process the interrupt.
- Write the value 3h to the CPDMA_EOI_VECTOR register.

3 Software Operation

3.1 Buffer Descriptors

The buffer descriptor is a central part of the 3pGSW module and is how the application software describes Ethernet packets to be sent and empty buffers to be filled with incoming packet data.

Note: The 8k bytes of Ethernet Subsystem CPPI RAM begin at address 0x02c82000 and end at 0x02c83FFF from the 3PSW perspective. The buffer descriptors programmed to access the CPPI RAM memory should use the address range from 0x02c82000.

3.1.1 TX Buffer Descriptors

A transmit buffer descriptor is a contiguous block of four 32-bit data words aligned on a 32-bit word boundary.

The transmit buffer descriptor is shown in Figure 2.

Figure 2. Transmit Buffer Descriptor Format

Word 0

31 0
NEXT_DESCRIPTOR_POINTER

Word 1

31 0
BUFFER_POINTER

Word 2

31 16	15 0
BUFFER_OFFSET	BUFFER_LENGTH

Word 3

31	30	29	28	27	26	25 24
SOP	EOP	OWNERSHIP	EOQ	TEARDOWN_COMPLETE	PASS_CRC	Reserved

23 21	20	19 17	16
Reserved	TO_PORT_EN	Reserved	TO_PORT

15 11	10 0
Reserved	PACKET_LENGTH

3.1.1.1 Next Descriptor Pointer (NEXT_DESCRIPTOR_POINTER)

This field indicates that the 32-bit word aligned memory address of the next buffer descriptor is in the transmit queue. This is the mechanism used to reference the next buffer descriptor from the current buffer descriptor. If the value of this pointer is zero, then the current buffer is the last buffer in the queue. The host sets the NEXT_DESCRIPTOR_POINTER.

3.1.1.2 Buffer Pointer (BUFFER_POINTER)

This field indicates that the byte aligned memory address of the buffer is associated with the buffer descriptor. The host sets the BUFFER_POINTER.

3.1.1.3 Buffer Offset (BUFFER_OFFSET)

This field indicates how many unused bytes are at the start of the buffer. A value of 0 indicates that no unused bytes are at the start of the buffer and that valid data begins on the first byte of the buffer. A value of 000Fh (decimal 15) indicates that the first 15 bytes of the buffer are to be ignored by the port and that valid buffer data starts on byte 16 of the buffer. The host sets the BUFFER_OFFSET value. The range of legal values for this field is 0 to (BUFFER_LENGTH - 1). Valid only on SOP.

3.1.1.4 Buffer Length (BUFFER_LENGTH)

This field indicates how many valid data bytes are in the buffer. Unused or protocol specific bytes at the beginning of the buffer are not counted in the BUFFER_LENGTH field. The host sets the BUFFER_LENGTH. The BUFFER_LENGTH must be greater than zero.

3.1.1.5 Start of Packet (SOP)

When set, this field indicates that the descriptor buffer is the first buffer in the packet.

3.1.1.6 End of Packet (EOP)

When set, this field indicates that the descriptor buffer is the last buffer in the packet.

3.1.1.7 Ownership (OWNERSHIP)

When set, this field indicates ownership of the packet and is valid only on SOP. This bit is set by the host and cleared by the port when the packet has been received. The host uses this bit to reclaim buffers.

3.1.1.8 End of Queue (EOQ)

This field is set by the port to indicate that all packets in the queue have been transmitted and that the transmit queue is empty. The EOQ is determined by the port when the NEXT_DESCRIPTOR_POINTER is zero on an EOP buffer. This bit is valid only on EOP.

3.1.1.9 Teardown Complete (TEARDOWN_COMPLETE)

This field is set by the port to indicate that the host commanded teardown process is complete, and that the channel buffers may be reclaimed by the host. This bit is valid only on SOP.

3.1.1.10 Pass CRC (PASS_CRC)

This field is valid only on SOP.

When PASS_CRC is 0, transmit the GMAC generated CRC. A CRC (or placeholder) at the end of the data is allowed but not required and the BUFFER_LENGTH and PACKET_LENGTH fields should not include the CRC bytes if they are present.

When PASS_CRC is 1, transmit the host supplied CRC. The BUFFER_LENGTH and PACKET_LENGTH fields should include the four CRC bytes. The GMAC generated CRC will not be used (unless replaced due to VLAN aware mode) and the host supplied CRC should be in the last four bytes of the data.

3.1.1.11 To Port Enable (TO_PORT_EN)

When set, this field indicates that the packet is a directed packet to be sent to the TO_PORT field port number. This field is set by the host. The packet is sent to one port only (index not mask). This bit is valid on SOP.

3.1.1.12 To Port (TO_PORT)

This field indicates the port number to which the directed packet should be sent. This field is set by the host. This field is valid on SOP.

When TO_PORT is 0, send the packet to port 0 if TO_PORT_EN is asserted.

When TO_PORT is 1, send the packet to port 1 if TO_PORT_EN is asserted.

3.1.1.13 Packet Length (PACKET_LENGTH)

This field specifies the number of bytes in the entire packet. Offset bytes are not included. The sum of the BUFFER_LENGTH fields should equal the PACKET_LENGTH. Valid only on SOP. The PACKET_LENGTH must be greater than zero. The packet data will be truncated to the PACKET_LENGTH if the PACKET_LENGTH is shorter than the sum of the packet buffer descriptor buffer lengths. A host error occurs if the PACKET_LENGTH is greater than the sum of the packet buffer descriptor buffer lengths.

3.1.2 RX Buffer Descriptors

A receive buffer descriptor is a contiguous block of four 32-bit data words aligned on a 32-bit word boundary.

The receive buffer descriptor is shown in Figure 3.

Figure 3. Receiver Buffer Descriptor Format

Word 0

31 … 0
NEXT_DESCRIPTOR_POINTER

Word 1

31 … 0
BUFFER_POINTER

Word 2

31 … 27	26 … 16	15 … 11	10 … 0
Reserved	BUFFER_OFFSET	Reserved	BUFFER_LENGTH

Word 3

31	30	29	28	27	26	25	24
SOP	EOP	OWNERSHIP	EOQ	TEARDOWN_COMPLETE	PASSED_CRC	JABBER	OVERSIZE

23	22	21	20	19	18	17	16
FRAGMENT	UNDERSIZED	CONTROL	OVERRUN	CODE_ERROR	ALIGN_ERROR	CRC_ERROR	VLAN_ENCAP

15 … 11	10 … 8
Reserved	PACKET_LENGTH

7 … 0
PACKET_LENGTH

3.1.2.1 Next Descriptor Pointer (NEXT_DESCRIPTOR_POINTER)

This field indicates that the 32-bit word aligned memory address of the next buffer descriptor is in the receive queue. This is the mechanism used to reference the next buffer descriptor from the current buffer descriptor. If the value of this pointer is zero, then the current buffer is the last buffer in the queue. The host sets the NEXT_DESCRIPTOR_POINTER.

3.1.2.2 Buffer Pointer (BUFFER_POINTER)

This field indicates that the byte aligned memory address of the buffer is associated with the buffer descriptor. The host sets the BUFFER_POINTER.

3.1.2.3 Buffer Offset (BUFFER_OFFSET)

This field indicates how many unused bytes are at the start of the buffer. The BUFFER_OFFSET is reduced to 12-bits. A value of 0 indicates that there are no unused bytes at the start of the buffer and that valid data begins on the first byte of the buffer. A value of 000Fh (decimal 15) indicates that the first 15 bytes of the buffer are to be ignored by the port and that valid buffer data starts on byte 16 of the buffer. The port writes the BUFFER_OFFSET with the value from the RX_BUFFER_OFFSET register value. The host initializes the BUFFER_OFFSET to zero for free buffers. The BUFFER_LENGTH must be greater than the RX_BUFFER_OFFSET value. The BUFFER_OFFSET is valid only on SOP.

3.1.2.4 Buffer Length (BUFFER_LENGTH)

This field indicates how many valid data bytes are in the buffer. The BUFFER_LENGTH is reduced to 12-bits. Unused or protocol specific bytes at the beginning of the buffer are not counted in the BUFFER_LENGTH field. The host initializes the BUFFER_LENGTH, but the port may overwrite the host initiated value with the actual BUFFER_LENGTH value on SOP and/or EOP buffer descriptors. SOP BUFFER_LENGTH values will be overwritten if the packet size is less than the size of the buffer or if the offset is nonzero. EOP BUFFER_LENGTH values will be overwritten if the entire buffer is not filled up with data. The BUFFER_LENGTH must be greater than zero.

3.1.2.5 Start of Packet (SOP)

When set, this field indicates that the descriptor buffer is the first buffer in the packet. The port sets the SOP bit.

3.1.2.6 End of Packet (EOP)

When set, this field indicates that the descriptor buffer is the last buffer in the packet. The port sets the EOP bit.

3.1.2.7 Ownership (OWNERSHIP)

When set, this field indicates ownership of the packet and is valid only on SOP. This bit is set by the host and cleared by the port when the packet has been transmitted. The host uses this bit to reclaim buffers.

3.1.2.8 End of Queue (EOQ)

This field is set by the port to indicate that the receive queue is empty. The EOQ is determined by the port when the NEXT_DESCRIPTOR_POINTER is zero on an EOP buffer. This bit is valid only on EOP.

3.1.2.9 Teardown Complete (TEARDOWN_COMPLETE)

This field is set by the port to indicate that the host commanded teardown process is complete, and the channel buffers may be reclaimed by the host. This bit is valid only on SOP.

3.1.2.10 Pass CRC (PASS_CRC)

This field is set by the port to indicate that the CRC was passed with the data. The PACKET_LENGTH includes the CRC bytes. The PASS_CRC bit is valid only on SOP.

3.1.2.11 Jabber Frame (JABBER)

When set, this field indicates that the frame is a jabber frame and was not discarded because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.12 Oversize Frame (OVERSIZE)

This field indicates that the frame is an oversized frame and was not discarded because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.13 Fragment Frame (FRAGMENT)

This field Indicates that the frame is a fragment and was not filtered because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.14 Undersize Frame (UNDERSIZED)

This field indicates that the frame is undersized and was not filtered because the RX_CSF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.15 Control Frame (CONTROL)

This field indicates that the frame is a MAC control frame and was not discarded because the RX_CMF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.16 Overrun Frame (OVERRUN)

This field is set by the port to indicate that the frame reception was aborted due to buffer overrun. This bit is valid only on SOP.

3.1.2.17 Code Error (CODE_ERROR)

This field indicates that the frame contained a code error and was not filtered because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.18 Alignment Error (ALIGN_ERROR)

This field indicates that the frame contained an alignment error and was not discarded because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.19 CRC Error (CRC_ERROR)

This field indicates that the frame contained a CRC error and was not discarded because the RX_CEF_EN bit was set in the GMAC*n*_MACCONTROL register. Valid only on SOP.

3.1.2.20 VLAN Encapsulated Packet (VLAN_ENCAP)

When set, this field indicates that the packet data contains a 32-bit VLAN header word that is included in the packet byte count. This field is set by the port to be the value of the 3pGSw switch control register RX_VLAN_ENCAP bit.

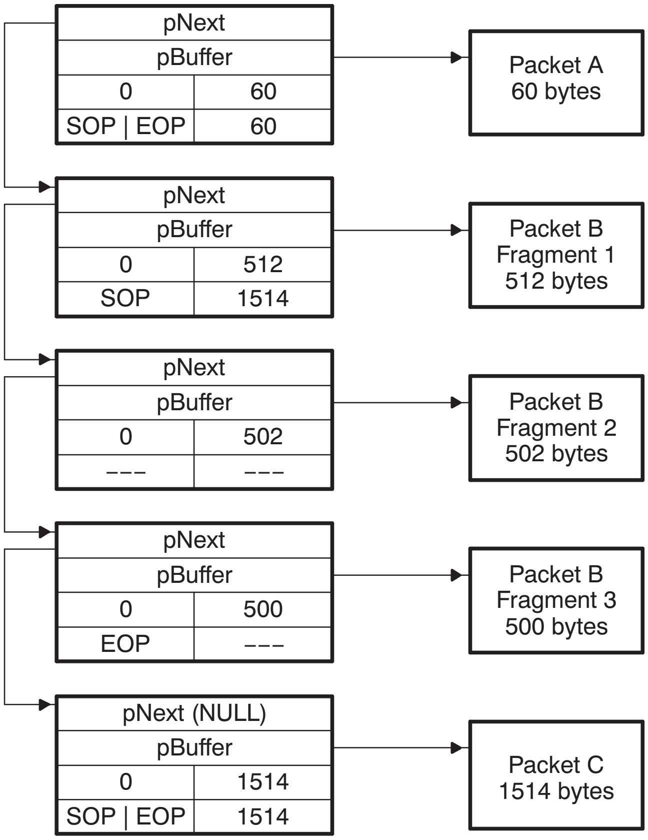
3.1.2.21 Packet Length (PACKET_LENGTH)

This field specifies the number of bytes in the entire packet. The PACKET_LENGTH is reduced to 12-bits. Offset bytes are not included. The sum of the BUFFER_LENGTH fields should equal the PACKET_LENGTH. Valid only on SOP.

3.1.3 Buffer Descriptor Example

Consider, for example, three packets to be transmitted when; Packet A is a single fragment (60 bytes), Packet B is fragmented over three buffers (1514 bytes total), and Packet C is a single fragment (1514 bytes). The linked list of transmit buffer descriptors used to describe these three packets is shown in Figure 4.

Figure 4. Descriptor Linked List



3.1.4 Transmit and Receive Descriptor Queues

The Ethernet subsystem module processes descriptors in linked list chains. The descriptor lists controlled by the module are maintained by the application software through the use of the head descriptor pointer registers (HDP). The subsystem supports eight transmit channels (0 - 7) and eight receive channels (0 -7), There is a head descriptor pointer register every channel. The head descriptor pointer registers are:

- TX*n*_HDP - Transmit Channel*n* DMA Head Descriptor Pointer Register (*n* represents channels 0 - 7).
- RX*n*_HDP - Receive Channel*n* DMA Head Descriptor Pointer Register (*n* represents channels 0 - 7).

After subsystem reset and before enabling the GMAC for send or receive, all 16 head descriptor pointer registers must be initialized to 0. The GMAC uses a simple system to determine if a descriptor is currently owned by the GMAC or by the application software. There is a flag in the buffer descriptor flags called OWNERSHIP. When this flag is set, the packet that is referenced by the descriptor is considered to be owned by the GMAC.

Note that ownership is done on a packet based granularity, not on descriptor granularity, so only SOP descriptors make use of the OWNERSHIP flag.

As packets are processed, the GMAC patches the SOP descriptor of the corresponding packet and clears the OWNERSHIP flag. This is an indication that the subsystem has finished processing all descriptors up to and including the first with the EOP flag set, indicating the end of the packet.

To add a descriptor or a linked list of descriptors, the software application simply writes the pointer to the descriptor or first descriptor of a list to the corresponding HDP register. The last descriptor in the list must have its next pointer cleared to 0. This is the only way the EMAC has of detecting the end of the list. So in the case where only a single descriptor is added, its next descriptor pointer must be initialized to 0.

3.2 Transmit Operation

After reset, the host must write zeroes to all transmit DMA state head descriptor pointers. The transmit port may then be enabled. To initiate packet transmission, the host constructs transmit queues in memory (one or more packets for transmission) and then writes the appropriate transmit DMA state head descriptor pointers. For each buffer added to a transmit queue, the host must initialize the transmit buffer descriptor values as follows:

- Write the NEXT_DESCRIPTOR_POINTER with the 32-bit aligned address of the next descriptor in the queue (zero if last descriptor).
- Write the BUFFER_POINTER with the byte aligned address of the buffer data.
- Write the BUFFER_LENGTH with the number of bytes in the buffer.
- Write the BUFFER_OFFSET with the number of bytes in the offset to the data (nonzero with SOP only).
- Set the SOP, EOP, and OWNERSHIP bits as appropriate.
- Clear the EOQ bit.

The port begins TX packet transmission on a given channel when the host writes the channel's transmit queue head descriptor pointer with the address of the first buffer descriptor in the queue (nonzero value). Each channel may have one or more queues, so each channel may have one or more head descriptor pointers. The first buffer descriptor for each transmit packet must have the start of packet (SOP) bit and the OWNERSHIP bit set to one by the host. The last buffer descriptor for each transmit packet must have the end of packet (EOP) bit set to one by the host. The port will transmit packets until all queued packets have been transmitted and the queue(s) are empty. When each packet transmission is complete, the port will clear the OWNERSHIP bit in the packet's SOP buffer descriptor and issue an interrupt to the host by writing the packet's last buffer descriptor address to the queue's transmit DMA state completion pointer. The interrupt is generated by the write, regardless of the value written. When the last packet in a queue has been transmitted, the port sets the end of queue (EOQ) bit in the EOP buffer descriptor, clears the OWNERSHIP bit in the SOP descriptor, zeroes the appropriate DMA state head descriptor pointer, and then issues a transmit interrupt to the host by writing to the queue's associated transmit completion pointer (address of the last buffer descriptor processed by the port). The port issues a maskable level interrupt (which may then be routed through external interrupt control logic to the host).

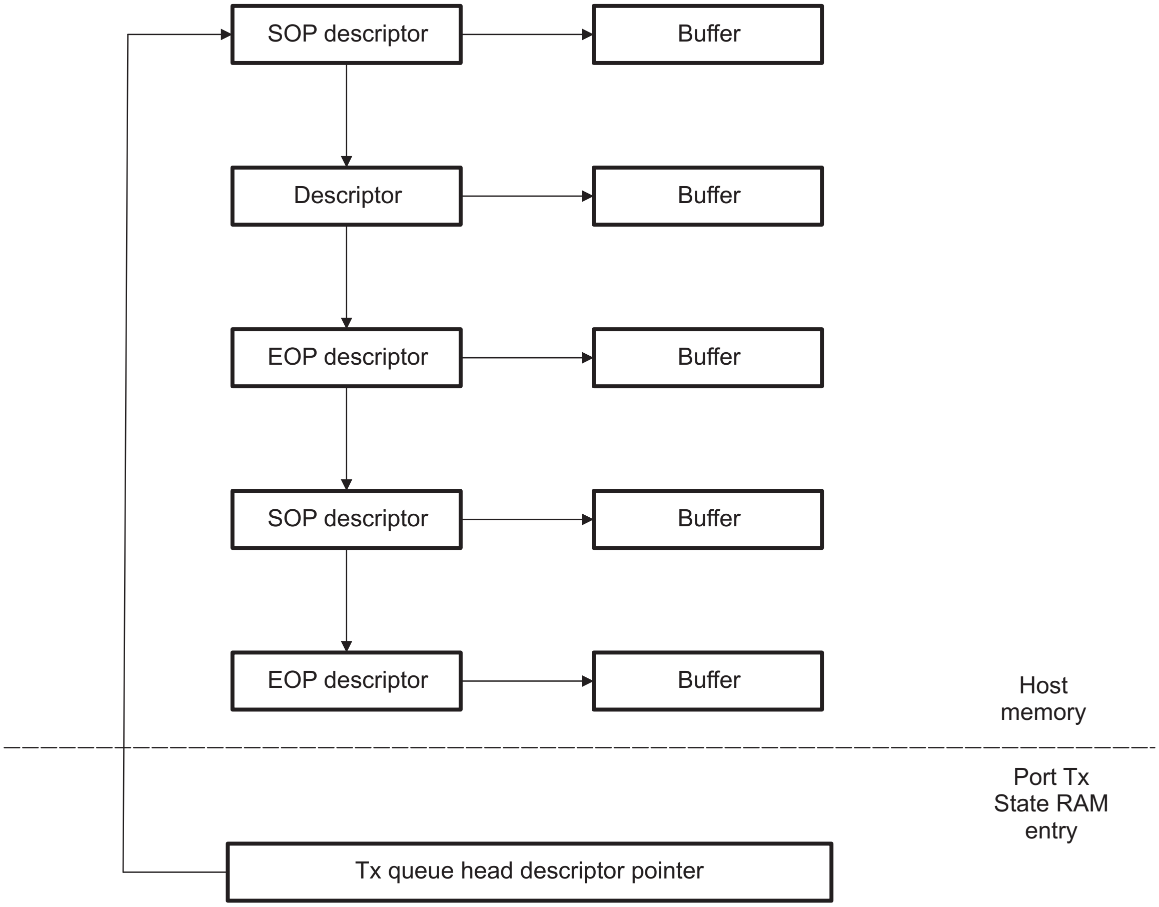
On interrupt from the port, the host processes the buffer queue, detecting transmitted packets by the status of the OWNERSHIP bit in the SOP buffer descriptor. If the OWNERSHIP bit is cleared to zero, then the packet has been transmitted and the host may reclaim the buffers associated with the packet. The host continues queue processing until the end of the queue or until a SOP buffer descriptor is read that contains a set OWNERSHIP bit indicating that the packet transmission is not complete. The host determines that all packets in the queue have been transmitted when the last packet in the queue has a cleared OWNERSHIP bit in the SOP buffer descriptor, the EOQ bit is set in the last packet EOP buffer descriptor, and the NEXT_DESCRIPTOR_POINTER of the last packet EOP buffer descriptor is zero. The host acknowledges an interrupt by writing the address of the last buffer descriptor to the queue's associated transmit completion pointer in the transmit DMA state. If the host written buffer address value is different from the buffer address written by the port, then the level interrupt remains asserted. If the host written buffer address value is equal to the port written value, then the level interrupt is de-asserted. The port writes to the completion pointer and actually stores the value in the state register (ram). The host written value is not written to the register location. The host written value is compared to the register contents (which were written by the port) and if the two values are equal, the interrupt is removed, otherwise the interrupt remains asserted. The host may process multiple packets prior to acknowledging an interrupt, or the host may acknowledge interrupts for every packet.

A mis-queued packet condition may occur when the host adds a packet to a queue for transmission as the port finishes transmitting the previous last packet in the queue. The mis-queued packet is detected by the host when queue processing detects a cleared OWNERSHIP bit in the SOP buffer descriptor, a set EOQ bit in the EOP buffer descriptor, and a nonzero NEXT_DESCRIPTOR_POINTER in the EOP buffer descriptor. A mis-queued packet means that the port read the last EOP buffer descriptor before the host added the new last packet to the queue, so the port determined the queue empty just before the last packet was added. The host corrects the mis-queued packet condition by initiating a new packet transfer for the mis-queued packet by writing the mis-queued packet's SOP buffer descriptor address to the appropriate DMA state transmit queue head descriptor pointer.

The host may add packets to the tail end of an active transmit queue at any time by writing the NEXT_DESCRIPTOR_POINTER to the current last descriptor in the queue. If a transmit queue is empty (inactive), the host may initiate packet transmission at any time by writing the appropriate transmit DMA state head descriptor pointer. The host software should always check for and reinitiate transmission for mis-queued packets during queue processing on interrupt from the port. In order to preclude software underrun, the host should avoid adding buffers to an active queue for any transmit packet that is not complete and ready for transmission.

The port determines that a packet is the last packet in the queue by detecting the EOP bit set with a zero NEXT_DESCRIPTOR_POINTER in the packet buffer descriptor. If the EOP bit is set and the NEXT_DESCRIPTOR_POINTER is nonzero, then the queue still contains one or more packets to be transmitted. If the EOP bit is set with a zero NEXT_DESCRIPTOR_POINTER, then the port will set the EOQ bit in the packet's EOP buffer descriptor and then zero the appropriate head descriptor pointer previous to interrupting the port (by writing the completion pointer) when the packet transmission is complete.

Figure 5. Transmit Queue



3.3 Receive Operation

After reset, the host must write zeroes to all of the receive DMA state head descriptor pointers. The receive port may then be enabled. To initiate packet reception, the host constructs receive queues in the memory and then writes the appropriate receive DMA state head descriptor pointer. For each receive buffer descriptor added to the queue, the host must initialize the receive buffer descriptor values as follows:

- Write the NEXT_DESCRIPTOR_POINTER with the 32-bit aligned address of the next descriptor in the queue (zero if last descriptor).
- Write the BUFFER_POINTER with the byte aligned address of the buffer data.
- Clear the Offset field.
- Write the BUFFER_LENGTH with the number of bytes in the buffer.
- Clear the SOP, EOP, and EOQ bits.
- Set the OWNERSHIP bit.

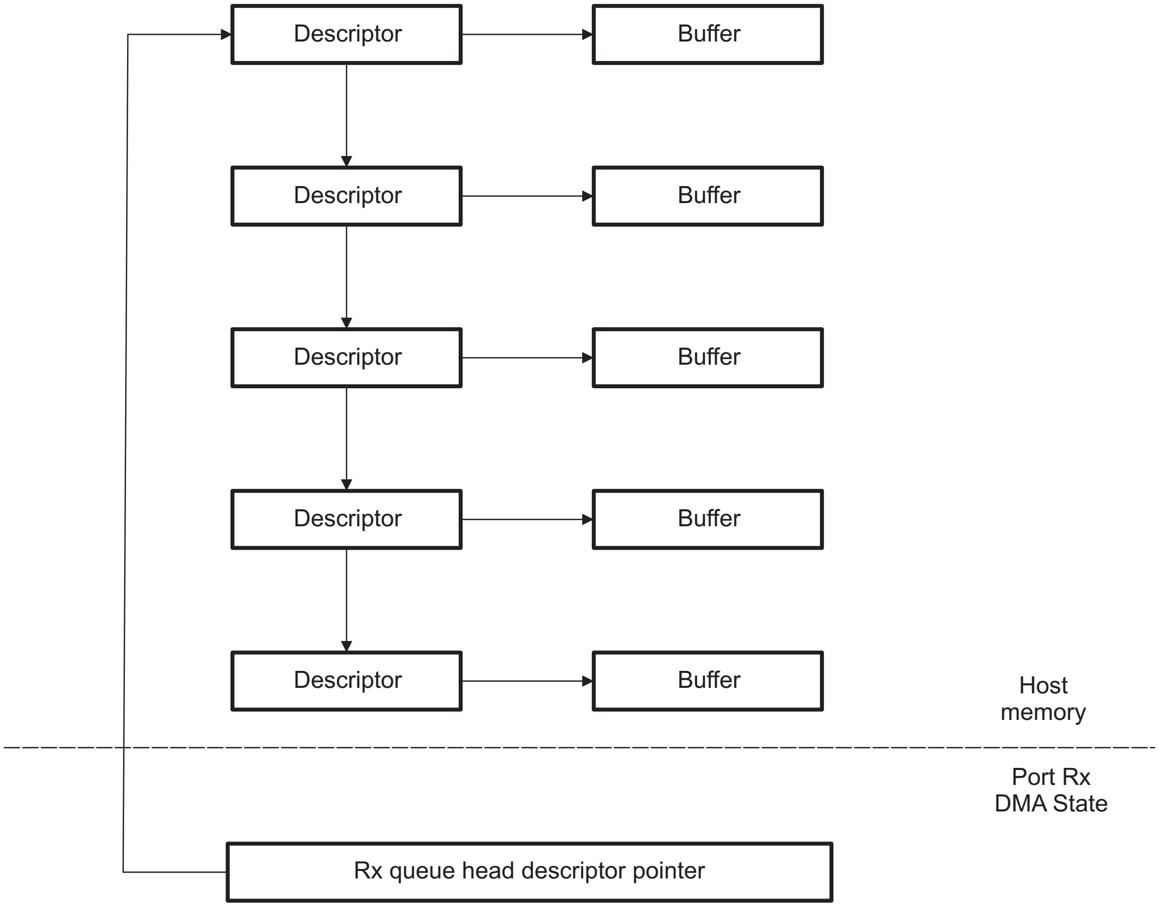
The host enables packet reception on a given channel by writing the address of the first buffer descriptor in the queue (nonzero value) to the channel's head descriptor pointer in the channel's receive DMA state. When packet reception begins on a given channel, the port fills each receive buffer with data in order starting with the first buffer and proceeding through the receive queue. If the BUFFER_OFFSET in the receive DMA state is nonzero, then the port will begin writing data after the offset number of bytes in the SOP buffer. The port performs the following operations at the end of each packet reception:

- Overwrite the BUFFER_LENGTH in the packet’s EOP buffer descriptor with the number of bytes actually received in the packet’s last buffer. The host initialized value is the buffer size. The overwritten value will be less than or equal to the host initialized value.
- Set the EOP bit in the packet's EOP buffer descriptor.
- Set the EOQ bit in the packet's EOP buffer descriptor if the current packet is the last packet in the queue.
- Overwrite the packet's SOP buffer descriptor BUFFER_OFFSET with the receive DMA state value (the host initialized the buffer descriptor BUFFER_OFFSET value to zero). All non SOP buffer descriptors must have a zero BUFFER_OFFSET initialized by the host.
- Overwrite the packet's SOP buffer descriptor BUFFER_LENGTH with the number of valid data bytes in the buffer. If the buffer is full, the BUFFER_LENGTH will be the buffer size minus BUFFER_OFFSET.
- Set the SOP bit in the packet's SOP buffer descriptor.
- Write the SOP buffer descriptor PACKET_LENGTH field.
- Clear the OWNERSHIP bit in the packet's SOP buffer descriptor.
- Issue a receive host interrupt by writing the address of the packet's last buffer descriptor to the queue's receive DMA state completion pointer. The interrupt is generated by the write to the receive DMA state completion pointer address location, regardless of the value written.

On interrupt, the host processes the receive buffer queue detecting received packets by the status of the OWNERSHIP bit in each packet's SOP buffer descriptor. If the OWNERSHIP bit is cleared, then the packet has been completely received and is available to be processed by the host. The host may continue receive queue processing until the end of the queue or until a buffer descriptor is read that contains a set OWNERSHIP bit indicating that the next packet's reception is not complete. The host determines that the receive queue is empty when the last packet in the queue has a cleared OWNERSHIP bit in the SOP buffer descriptor, a set EOQ bit in the EOP buffer descriptor, and the NEXT_DESCRIPTOR_POINTER in the EOP buffer descriptor is zero.

A mis-queued buffer may occur when the host adds buffers to a queue as the port finishes the reception of the previous last packet in the queue. The mis-queued buffer is detected by the host when queue processing detects a cleared OWNERSHIP bit in the SOP buffer descriptor, a set EOQ bit in the EOP buffer descriptor, and a nonzero NEXT_DESCRIPTOR_POINTER in the EOP buffer descriptor. A mis-queued buffer means that the port read the last EOP buffer descriptor before the host added buffer descriptor(s) to the queue, so the port determined queue empty just before the host added more buffer descriptor(s). In the transmit case, the packet transmission is delayed by the time required for the host to determine the condition and reinitiate the transaction, but the packet is not actually lost. In the receive case, receive overrun condition may occur in the mis-queued buffer case. If a new packet reception is begun during the time that the port has determined the EOQ condition, then the received packet will overrun (SOP overrun). If the mis-queued buffer occurs during the middle of a packet reception, then middle of packet overrun may occur. If the mis-queued buffer occurs after the last packet has completed and is corrected before the next packet reception begins, then overrun will not occur. The host acts on the mis-queued buffer condition by writing the added buffer descriptor address to the appropriate receive DMA state head descriptor pointer.

Figure 6. Receive Queue



3.4 Initialization and Configuration of 3-Port Gigabit Switch Subsystem

To configure the 3pGSw for operation, the host must perform the following:

- Configure the SerDes PLL (CFGPLL).
- Configure the SerDes Receiver and Transmitter (CFGRX*n* and CFGTX*n*).
- Configure the 3pGSw switch control register.
- Configure the GMAC0 and GMAC1 source address registers.
- Configure the 3pGSw statistics port enable register.
- Configure the ALE.
- Configure the CPDMA receive DMA controller.
- Configure the CPDMA transmit DMA controller.
- Configure GMAC0 and GMAC1.

An example of the chip level SerDes configuration is shown in the example below.

```
void Configure_SerDes(void)
{
     //Serdes Configurations
     #define KEY_REG                    (*(volatile Uint32 *)0x02049054)
     #define SERDES_RX0_CFG             (*(volatile Uint32 *)0x020490B0)
     #define SERDES_RX1_CFG             (*(volatile Uint32 *)0x020490B4)
     #define SERDES_TX0_CFG             (*(volatile Uint32 *)0x020490B8)
     #define SERDES_TX1_CFG             (*(volatile Uint32 *)0x020490BC)
     #define SerdesPLLReg               (*(volatile Uint32 *)0x020490A8)

     //Key Reg
      KEY_REG = 0xADDDECAF;

     //Assuming a 62.5 Mhz REFCLKP/N and 1.25Gbps Line Rate  operation
      SerdesPLLReg = 0x13;
      for(i=0;i<2500;i++)
      asm("    nop");

     //Rx0
      SERDES_RX0_CFG = 0x00089121;

     //Rx1
      SERDES_RX1_CFG = 0x00089121;

     //Tx0
      SERDES_TX0_CFG = 0x00000921;

     //Tx1
      SERDES_TX1_CFG = 0x00000921;

}
```

An example of the ethernet subsystem configuration is shown in the example below.

```
#define TEST_MR_ADV_ABILITY    0x9801

void Setup_Subsystem()
{

     //Enable all the 8 channel tx of the CPDMA
     ptrcpsw->TX_INTMASK_SET = CSL_CPSW_TX_INTMASK_SET_TX7_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX6_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX5_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX4_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX3_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX2_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX1_MASK_MASK
                    |CSL_CPSW_TX_INTMASK_SET_TX0_MASK_MASK;


     //Enabling 8 Channel  Rx interrupts
     ptrcpsw->RX_INTMASK_SET =  CSL_CPSW_RX_INTMASK_SET_RX7_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX6_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX5_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX4_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX3_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX2_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX1_PEND_MASK
                    |CSL_CPSW_RX_INTMASK_SET_RX0_PEND_MASK;

     //CPDMA TX and RX Enable
     ptrcpsw->TX_CONTROL = CSL_CPSW_TX_CONTROL_TX_EN_ENABLE;
     ptrcpsw->RX_CONTROL = CSL_CPSW_RX_CONTROL_RX_EN_ENABLE;

     //ALE control
     //Enable ALE and ALE Bypass
     ptrcpsw->ALE_CONTROL= CSL_CPSW_ALE_CONTROL_ENABLE_ALE_MASK
                    |CSL_CPSW_ALE_CONTROL_ALE_BYPASS_MASK;


     //ALE Port Control
     ptrcpsw->ALE_PORTCTL0 = CPSW_PORT_STATE_FORWARD;
     ptrcpsw->ALE_PORTCTL1 = CPSW_PORT_STATE_FORWARD;
     ptrcpsw->ALE_PORTCTL2 = CPSW_PORT_STATE_FORWARD;

     //MAC Control
     //Full Duplex mode GMII enable
     ptrcpsw->GMAC0_MACCONTROL=CSL_CPSW_GMAC0_MACCONTROL_CTL_EN_MASK
                    |CSL_CPSW_GMAC0_MACCONTROL_GMII_EN_MASK;
     ptrcpsw->GMAC1_MACCONTROL=CSL_CPSW_GMAC1_MACCONTROL_CTL_EN_MASK
                    |CSL_CPSW_GMAC1_MACCONTROL_GMII_EN_MASK;


     //P2 Transmit Header Priority to Switch priority mapping
     ptrcpsw->P2_TX_PRI_MAP=0x00;

     //Enable Tx interrupt for all the channels
     ptr3gss->TX_EN=0xFF;
     ptrsgmii0->CONTROL=CSL_SGMII_CONTROL_MASTER_MASK
                    |CSL_SGMII_CONTROL_FAST_LINK_TIMER_MASK
                    |CSL_SGMII_CONTROL_MR_AN_ENABLE_MASK;

     ptrsgmii0->MR_ADV_ABILITY = TEST_MR_ADV_ABILITY;

     ptrsgmii1->CONTROL=CSL_SGMII_CONTROL_MASTER_MASK
                    |CSL_SGMII_CONTROL_FAST_LINK_TIMER_MASK
                    |CSL_SGMII_CONTROL_MR_AN_ENABLE_MASK;

     ptrsgmii1->MR_ADV_ABILITY = TEST_MR_ADV_ABILITY;
}
```

3.5 Address Lookup Engine (ALE)

The address lookup engine (ALE) processes all of the received packets to determine if any packet(s) should be forwarded, and if so, to which port(s). Once it is determined that a packet should be forwarded, the ALE uses the received port number, destination address, source address, length/type, and VLAN information to determine how the packet should be forwarded. The ALE outputs the port mask to the switch fabric that indicates to which port(s) the packet should be forwarded. The ALE is enabled when the ENABLE_ALE bit in the ALE_CONTROL register is set. All packets are dropped when the ENABLE_ALE bit is cleared to zero.

In normal operation, the GMAC*n* modules are configured to issue an abort, instead of an end of packet (EOP), at the end of a packet that contains an error (runt, fragment, oversize, jabber, crc, alignment, code etc.) or at the end of a MAC control packet. However, when the GMAC*n*_MACCONTROL register configuration bit(s) RX_CEF_EN, RX_CSF_EN, or RX_CMF_EN are set, error frames, short frames, or MAC control frames have a normal EOP instead of an abort EOP. When the ALE receives a packet that contains errors (due to a set header error bit) or a MAC control frame and does not receive an abort, the packet will be forwarded only to the host port (port 2). No ALE learning occurs on packets with errors or MAC control frames.

The ALE may be configured to operate in bypass mode by setting the ALE_BYPASS bit in the ALE_CONTROL register. When in bypass mode, all GMAC*n* received packets are forwarded only to the host port (port 2). Packets from GMAC0 (port 0) are forwarded with the supervisory bit cleared. Packets from GMAC1 (port 1) are forwarded with the supervisory bit set. The supervisory bit allows packets from the two ports to be on separate RX DMA channels by configuring the 3pGSw CPDMA RX (Port 2 TX) switch priority to DMA channel mapping register (CPDMA_RX_CH_MAP). In bypass mode, the ALE processes host port transmit packets the same as in normal mode.

The ALE may be configured to operate in OUI deny mode by setting the ENABLE_OUI_DENY bit in the ALE_CONTROL register. When in OUI deny mode, a packet with a non-matching OUI source address will be dropped unless the destination address matches a multicast table entry with the SUPER bit set.

Broadcast packets will be dropped unless the broadcast address is entered into the table with the SUPER bit set. Unicast packets will be dropped unless the unicast address is in the table with the BLOCK and SECURE bits in the unicast address register (see Section 3.5.1.4) and both bits are set (unicast supervisory packet).

Multicast supervisory packets are designated by the SUPER bit in the table entry. Unicast supervisory packets are indicated when both BLOCK and SECURE bits are set. Supervisory packets are not dropped due to rate limiting, OUI, or VLAN processing.

3.5.1 Address Table

The ALE table contains 1024 entries. Each table entry represents a free entry, an address, a VLAN, an address/VLAN pair, or an OUI address. Software should ensure that there are not double address entries in the table. A double entry in the table would be indeterminate.

Source address learning occurs for packets with a unicast, multicast, or broadcast destination address and a unicast or multicast (including broadcast) source address. Multicast source addresses have the group bit (bit 40) cleared before ALE processing begins, which changes the multicast source address to a unicast source address. A multicast address of all ones is the broadcast address which may be added to the table. A learned unicast source address is added to the table with the following control bits:

- UNICAST_TYPE: 11
- BLOCK: 0
- ENTRY_TYPE: 0

If the source address is equal to the destination address, then the address is learned. If the address is not found, then the address is updated. If the address is found, then the packet is dropped.

3.5.1.1 Free

The free (unused) address entry is shown in Figure 7.

Figure 7. Free (Unused) Address

67 62	61 60	59 0
Reserved	ENTRY_TYPE	Reserved
	00	

3.5.1.1.1 Table Entry Type (ENTRY_TYPE)

Free entry type.

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.2 Multicast Address

The multicast address entry is shown in Figure 8.

Figure 8. Multicast Address

67	66 64	63 62	61 60	59 48
SUPER	PORT_MASK	MCAST_FWD_STATE	ENTRY_TYPE	Reserved
			01	

47 0
MULTICAST_ADDRESS

3.5.1.2.1 Supervisory Packet (SUPER)

When set, this field indicates that the packet with a matching multicast destination address is a supervisory packet.

- 0: Non-supervisory packet
- 1: Supervisory packet

3.5.1.2.2 Port Mask (PORT_MASK)

This three bit field is the port bit mask that is returned with a found multicast destination address. There may be multiple bits set indicating that the multicast packet may be forwarded to multiple ports (but not the receiving port).

3.5.1.2.3 Multicast Forward State (MCAST_FWD_STATE)

When set, this field indicates the port state(s) that the receive port on a destination address lookup requires in order for the multicast packet to be forwarded to the transmit port(s). A transmit port must be in the forwarding state in order to forward the packet.

If the transmit PORT_MASK has multiple set bits, then each forward decision is independent of the other transmit port(s) forward decision.

The forward state test returns a true value if both the Rx and TX ports are in the required state.

- 00: Forwarding
- 01: Blocking/forwarding/learning
- 10: Forwarding/learning
- 11: Forwarding

3.5.1.2.4 Table Entry Type (ENTRY_TYPE)

Address entry type. Unicast or multicast determined by address (40).

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.2.5 Packet Address (MULTICAST_ADDRESS)

This is the 48-bit packet MAC address. For an OUI address, only the upper 24-bits of the address are used in the source or destination address lookup. Otherwise, all 48-bits are used in the lookup.

3.5.1.3 VLAN/Multicast Address

The VLAN/Multicast address entry is shown in Figure 9.

Figure 9. VLAN/Multicast Address

67	66 64	63 62	61 60	59 48
SUPER	PORT_MASK	MCAST_FWD_STATE	ENTRY_TYPE	VLAN_ID
			11	

47 0
MULTICAST_ADDRESS

3.5.1.3.1 Supervisory Packet (SUPER)

When set, this field indicates that the packet with a matching multicast destination address is a supervisory packet.

- 0: Non-supervisory packet
- 1: Supervisory packet

3.5.1.3.2 Port Mask (PORT_MASK)

This three bit field is the port bit mask that is returned with a found multicast destination address. There may be multiple bits set indicating that the multicast packet may be forwarded to multiple ports (but not the receiving port).

3.5.1.3.3 Multicast Forward State (MCAST_FWD_STATE)

When set, this field indicates the port state(s) that the receive port on a destination address lookup requires order for the multicast packet to be forwarded to the transmit port(s). A transmit port must be in the forwarding state in order to forward the packet.

If the transmit PORT_MASK has multiple set bits, then each forward decision is independent of the other transmit port(s) forward decision.

The forward state test returns a true value if both the Rx and TX ports are in the required state.

- 00: Forwarding
- 01: Blocking/forwarding/learning
- 10: Forwarding/learning
- 11: Forwarding

3.5.1.3.4 Table Entry Type (ENTRY_TYPE)

VLAN address entry. Unicast or multicast determined by address (40).

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.3.5 VLAN ID (VLAN_ID)

The unique identifier for VLAN identification. This is the 12-bit VLAN ID.

3.5.1.3.6 Packet Address (MULTICAST_ADDRESS)

This is the 48-bit packet MAC address. For an OUI address, only the upper 24-bits of the address are used in the source or destination address lookup. Otherwise, all 48-bits are used in the lookup.

3.5.1.4 Unicast Address

The unicast address entry is shown in Figure 10.

Figure 10. Unicast Address

67 66	65 64	63	62	61 60	59 48
UNICAST_TYPE	PORT_NUMBER	BLOCK	SECURE	ENTRY_TYPE	Reserved
00 or X1				01	

47 0
UNICAST_ADDRESS

3.5.1.4.1 Unicast Type (UNICAST_TYPE)

This field is applicable and indicates the type of unicast address the table entry contains.

- 00: Unicast address that is not ageable.
- 01: Ageable unicast address that has not been touched.
- 10: OUI address. The lower 24-bits are not applicable (not ageable).
- 11: Ageable unicast address that has not been touched.

3.5.1.4.2 Port Number (PORT_NUMBER)

- 0: Received port number is not applicable.
- 1: Drop the packet if the received port is not the secure port for the source address and do not update the address (BLOCK must be zero).

3.5.1.4.3 Block (BLOCK)

The BLOCK bit indicates that a packet with a matching source or destination address should be dropped (block the address).

- 0: Address is not blocked.
- 1: Drop a packet with a matching source or destination address (SECURE must be zero).

If BLOCK and SECURE are both set, then they no longer mean BLOCK and SECURE. When both are set, the BLOCK and SECURE bits indicate that the packet is a unicast supervisory (SUPER) packet and they determine the unicast forward state test criteria. If both bits are set and the receive port is in the forwarding, blocking, or learning state, then the packet is forwarded. If both bits are not set and the receive port is in the forwarding state, then the packet is forwarded.

3.5.1.4.4 Secure (SECURE)

This bit indicates that a packet with a matching source address should be dropped if the received port number is not equal to the table entry.

3.5.1.4.5 Table Entry Type (ENTRY_TYPE)

Address entry. Unicast or multicast determined by address (40).

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.4.6 Packet Address (UNICAST_ADDRESS)

This is the 48-bit packet MAC address. For an OUI address, only the upper 24-bits of the address are used in the source or destination address lookup. Otherwise, all 48-bits are used in the lookup.

3.5.1.5 OUI Unicast Address

The OUI unicast address entry is shown in Figure 11.

Figure 11. OUI Unicast Address

67 66	65 62	61 60	59 48
UNICAST_TYPE	Reserved	ENTRY_TYPE	Reserved
10		01	

47 24	23 0
UNICAST_ADDRESS	Reserved

3.5.1.5.1 Unicast Type (UNICAST_TYPE)

This field is not applicable.

- 00: Unicast address that is not ageable.
- 01: Ageable unicast address that has not been touched.
- 10: OUI address. The lower 24-bits are not applicable (not ageable).

- 11: Ageable unicast address that has not been touched.

3.5.1.5.2 Table Entry Type (ENTRY_TYPE)

Address entry. Unicast or multicast determined by address (40).

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.5.3 Packet Address (UNICAST_ADDRESS)

This field is an OUI address and only the upper 24-bits of the address are used in the source or destination address lookup. Otherwise, all 48-bits are used in the lookup.

3.5.1.6 *VLAN/Unicast Address*

The VLAN/Unicast address entry is shown in Figure 12.

Figure 12. VLAN/Unicast Address

67 66	65 64	63	62	61 60	59 48
UNICAST_TYPE	PORT_NUMBER	BLOCK	SECURE	ENTRY_TYPE	VLAN_ID
00 or X1				11	

47 0
UNICAST_ADDRESS

3.5.1.6.1 Unicast Type (UNICAST_TYPE)

This field is applicable and indicates the type of unicast address the table entry contains.

- 00: Unicast address that is not ageable.
- 01: Ageable unicast address that has not been touched.
- 10: OUI address. The lower 24-bits are not applicable (not ageable).
- 11: Ageable unicast address that has not been touched.

3.5.1.6.2 Port Number (PORT_NUMBER)

- 0: Received port number is not applicable.
- 1: Drop the packet if the received port is not the secure port for the source address and do not update the address (BLOCK must be zero).

3.5.1.6.3 Block (BLOCK)

The BLOCK bit indicates that a packet with a matching source or destination address should be dropped (block the address).

- 0: Address is not blocked.
- 1: Drop a packet with a matching source or destination address (SECURE must be zero).

If BLOCK and SECURE are both set, then they no longer mean BLOCK and SECURE. When both are set, the BLOCK and SECURE bits indicate that the packet is a unicast supervisory packet (SUPER) and they determine the unicast forward state test criteria. If both bits are set and the receive port is in the forwarding, blocking, or learning state, then the packet is forwarded. If both bits are not set and the receive port is in the forwarding state, then the packet is forwarded.

3.5.1.6.4 Secure (SECURE)

This bit indicates that a packet with a matching source address should be dropped if the received port number is not equal to the table entry.

3.5.1.6.5 Table Entry Type (ENTRY_TYPE)

VLAN address entry. Unicast or multicast determined by address (40).

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.6.6 VLAN ID (VLAN_ID)

The unique identifier for VLAN identification. This is the 12-bit VLAN ID.

3.5.1.6.7 Packet Address (UNICAST_ADDRESS)

This is the 48-bit packet MAC address. For an OUI address, only the upper 24-bits of the address are used in the source or destination address lookup. Otherwise, all 48-bits are used in the lookup.

3.5.1.7 VLAN

The VLAN address entry is shown in Figure 13.

Figure 13. VLAN Address

67 62	61 60	59 48	47 32
Reserved	ENTRY_TYPE	VLAN_ID	Reserved
	10		

31 27	26 24	23 19	18 16
Reserved	FORCE_UNTAGGED_EGRESS	Reserved	REG_MCAST_FLOOD_MASK

15 11	10 8	7 3	2 0
Reserved	UNREG_MCAST_FLOOD_MASK	Reserved	VLAN_MEMBER_LIST

3.5.1.7.1 Table Entry Type (ENTRY_TYPE)

VLAN entry type.

- 00: Free entry
- 01: Address entry. Unicast or multicast determined by address (40).
- 10: VLAN entry
- 11: VLAN address entry. Unicast or multicast determined by address (40).

3.5.1.7.2 VLAN ID (VLAN_ID)

The unique identifier for VLAN identification. This is the 12-bit VLAN ID.

3.5.1.7.3 Force Untagged Packet Egress (FORCE_UNTAGGED_EGRESS)

This field causes the packet VLAN tag to be removed on egress (except on port 2).

3.5.1.7.4 Registered Multicast Flood Mask (REG_MCAST_FLOOD_MASK)

This field is a registered multicast flood mask.

3.5.1.7.5 Unregistered Multicast Flood Mask (UNREG_MCAST_FLOOD_MASK)

This field is an unregistered multicast flood mask .

3.5.1.7.6 VLAN Member List (VLAN_MEMBER_LIST)

This three bit field indicates which port(s) are a member of the associated VLAN.

There are four processes that an incoming received packet may go through to determine packet forwarding. The processes are ingress filtering, VLAN_Aware lookup, VLAN_Unaware lookup, and egress.

In the packet ingress process, there is a forward state test for unicast destination addresses and a forward state test for multicast addresses. The multicast forward state test indicates the port states required by the receiving port in order for the multicast packet to be forwarded to the transmit port(s). A transmit port must be in the forwarding state for the packet to be forwarded for transmission. The MCAST_FWD_STATE bit in the multicast address register indicates the required port state for the receiving port as indicated in Section 3.5.1.3. The unicast forward state test indicates the port state required for the receiving port in order to forward the unicast packet. The transmit port must be in the forwarding state in order to forward the packet. The BLOCK and SECURE bits determine the unicast forward state test criteria.

If both bits are set and the receive port is in the forwarding, blocking, or learning state, then the packet is forwarded. If both bits are not set and the receive port is in the forwarding state, then the packet is forwarded. The transmit port must be in the forwarding state. The forward state test used in the ingress process is determined by the destination address packet type (multicast or unicast).

In general, packets received with errors are aborted and dropped by the address lookup engine without learning or updating. The error condition and the abort is indicated by the GMAC to the ALE. Packets with errors may be passed (not aborted) by a GMAC port if the port has set RX_CMF_EN, RX_CEF_EN, or RX_CSF_EN bit(s) in the GMAC*n*_MACCONTROL register.

Error packets that are passed by the GMAC are considered to be bypass packets by the ALE and are sent only to the host. Error packets do not learn or update addresses regardless of whether they are aborted or sent to the host. The types of packets considered to be error packets are shown below:

- Packets with crc, code, or alignment errors.
- MAC control packets.
- Runt or fragment packets.
- Oversized or jabber packets.

The host does not transmit (into the switch) any error packet except for MAC control frames. The host may transmit a directed MAC control frame to any other port.

3.5.2 Learning Process

The learning process is applied to each receive packet that is not aborted. The learning process happens at the same time as the packet forwarding process.

3.5.3 Packet Priority Handling

Packets are received on three ports, two of which are GMAC ethernet ports and the third port is the CPPI host port. Received packets have a received packet priority of 0 to 7 with 7 being the highest priority. The received packet priority is the port priority for untagged packets. The received packet priority is also the actual packet priority for priority tagged and VLAN tagged packets. The received packet priority is mapped through the receive ports that are associated with the packet priority to the header packet priority mapping register to obtain the header packet priority (the CPDMA Rx and TX nomenclature is reversed from the GMAC nomenclature for legacy reasons). The header packet priority is mapped through the header priority to the switch priority mapping register to obtain the hardware switch priority (0 to 3 with 3 being the highest priority). The header packet priority is then used as the actual transmit packet priority if the VLAN information is to be sent on egress.

3.6 MDIO Module Operation

3.6.1 Initializing the MDIO Module

The following steps are performed by the application software or device driver to initialize the MDIO device:

1. Configure the PREAMBLE and CLKDIV bits in the MDIO control register (MDIOCONTROL).
2. Enable the MDIO module by setting the ENABLE bit in MDIOCONTROL.
3. The MDIO PHY alive status register (MDIOALIVE) can be read in polling fashion until a PHY connected to the system responded, and the MDIO PHY link status register (MDIOLINK) can determine whether this PHY already has a link.
4. Setup the appropriate PHY addresses in the MDIO user PHY select register (MDIOUSERPHYSEL*n*), and set the LINKINTENB bit to enable a link change event interrupt if desirable.
5. If an interrupt on general MDIO register access is desired, set the corresponding bit in the MDIO user command complete interrupt mask set register (MDIOUSERINTMASKSET) to use the MDIO user access register (MDIOUSERACCESS*n*). Since only one PHY is used in this device, the application software can use one MDIOUSERACCESS*n* to trigger a completion interrupt; the other MDIOUSERACCESS*n* is not setup.

3.6.2 Writing Data To a PHY Register

The MDIO module includes a user access register (MDIOUSERACCESS*n*) to directly access a specified PHY device. To write a PHY register, perform the following:

1. Check to ensure that the GO bit in the MDIO user access register (MDIOUSERACCESS*n*) is cleared.
2. Write to the GO, WRITE, REGADR, PHYADR, and DATA bits in MDIOUSERACCESS*n* corresponding to the PHY and PHY register you want to write.
3. The write operation to the PHY is scheduled and completed by the MDIO module. Completion of the write operation can be determined by polling the GO bit in MDIOUSERACCESS*n* for a 0.
4. Completion of the operation sets the corresponding USERINTRAW bit (0 or 1) in the MDIO user command complete interrupt register (MDIOUSERINTRAW) corresponding to USERACCESS*n* used. If interrupts have been enabled on this bit using the MDIO user command complete interrupt mask set register (MDIOUSERINTMASKSET), then the bit is also set in the MDIO user command complete interrupt register (MDIOUSERINTMASKED) and an interrupt is triggered on the CPU.

3.6.3 Reading Data From a PHY Register

The MDIO module includes a user access register (MDIOUSERACCESS*n*) to directly access a specified PHY device. To read a PHY register, perform the following:

1. Check to ensure that the GO bit in the MDIO user access register (MDIOUSERACCESS*n*) is cleared.
2. Write to the GO, REGADR, and PHYADR bits in MDIOUSERACCESS*n* corresponding to the PHY and PHY register you want to read.
3. The read data value is available in the DATA bits in MDIOUSERACCESS*n* after the module completes the read operation on the serial bus. Completion of the read operation can be determined by polling the GO and ACK bits in MDIOUSERACCESS*n*. After the GO bit has cleared, the ACK bit is set on a successful read.
4. Completion of the operation sets the corresponding USERINTRAW bit (0 or 1) in the MDIO user command complete interrupt register (MDIOUSERINTRAW) corresponding to USERACCESS*n* used. If interrupts have been enabled on this bit using the MDIO user command complete interrupt mask set register (MDIOUSERINTMASKSET), then the bit is also set in the MDIO user command complete interrupt register (MDIOUSERINTMASKED) and an interrupt is triggered on the CPU.

4 Chip Level Configuration Registers

This section describes the chip level configuration registers. Table 4 lists the chip level configuration registers.

These three registers are write protected and KEY_REG will need to be programmed. Please refer the KEY_REG in Digital Media Processor Data Manual for programming the KEY_REG.

Table 4. System Module Registers

Offset	Acronym	Register Description	Section
0204 90A8h	CFGPLL	PLL Configuration Register	Section 4.1
0204 90B0h / 0204 90B4h	CFGRX0/1	Receiver Configuration Register	Section 4.2
0204 90B8h / 0204 90BCh	CFGTX0/1	Transmitter Configuration Register	Section 4.3

4.1 PLL Configuration Register (CFGPLL)

PLL can be configured through the CFGPLL register (Offset: 0204 90A8h).

The PLL configuration register is shown in Figure 14 and described in Table 5.

Figure 14. PLL Configuration Register (CFGPLL)

31 16
Reserved

15 10	9 8	7 5	4 1	0
Reserved	LB	Reserved	MPY	ENPLL
R-0x0	R/W-0x0	R-0x0	R/W-0x1001	R/W-0x1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 5. PLL Configuration Register (CFGPLL) Field Descriptions

Bit	Field	Value	Description
31-10	Reserved	0	Reserved.
9-8	LB	0-3h	Should be 0. All other values are reserved.
7-5	Reserved	0	Reserved.
4-1	MPY		MPY is the multiplication factor that is used to generate frequency.
		0-8h	Reserved.
		9h	20×
		Ah	25×
		B-Fh	Reserved.
0	ENPLL		Enable PLL
		0	PLL is disabled.
		1	Pll is enabled. Allow 1μs + 200 REFCLKP/N cycles to stabilize.

4.2 Receiver Configuration Register (CFGRX0/1)

The receiver is configured through CFGRX0/1 registers. (Offsets 0204 90B0h / 0204 90B4h).

The receiver configuration register format is shown in Figure 15 and described in Table 6.

Figure 15. Receiver Configuration Register (CFGRX0/1)

31	24
Reserved	
R-0x0	

23	22	19	18	16
Reserved	EQ		CDR	
R-0x0	R/W-0x0		R/W-0x0	

15	14	13	12	11	10	8
LOS		ALIGN		Reserved	TERM	
R/W-0x0		R/W-0x0		R-0x0	R/W-0x001	

7	6	5	4	2	1	0
INVPAIR	RATE		BUSWIDTH		Reserved	ENRX
R/W-0x0	R/W-0x01		R/W-0x0		R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 6. Receiver Configuration Register (CFGRX0/1) Field Descriptions

Bit	Field	Value	Description	
31-23	Reserved	0	Reserved.	
22-19	EQ		The equalizer enables and configures the adaptive equalizer to compensate for loss in the transmission media.	
			Low Frequency Gain	**Zero Freq(at e_{27} (min))**
		0000	Maximum	—
		0001	Adaptive	Adaptive
		001x	Reserved	Reserved
		01xx	Reserved	Reserved
		1000	Adaptive	365 MHz
		1001	Adaptive	275 MHz
		1010	Adaptive	195 MHz
		1011	Adaptive	140 MHz
		1100	Adaptive	105 MHz
		1101	Adaptive	75 MHz
		1110	Adaptive	55 MHz
		1111	Adaptive	50 MHz

Table 6. Receiver Configuration Register (CFGRX0/1) Field Descriptions (continued)

Bit	Field	Value	Description
18-16	CDR		Clock/data recovery. Configures the clock/data recovery algorithm.
		000	First order, threshold of 1. Phase offset tracking up to ±488ppm. Suitable for use in asynchronous systems with low frequency offset.
		001	First order, threshold of 16. Phase offset tracking up to ±390ppm. Suitable for use in synchronous systems. Offers superior rejection of random jitter, but is less responsive to systematic variation such as sinusoidal jitter.
		010	Second order, high precision, threshold of 1. Highest precision frequency offset matching but relatively poor response to changes in frequency offset, and long lock time. Suitable for use in systems with fixed frequency offset.
		011	Second order, high precision, threshold of 16. Highest precision frequency offset matching but poorest response to changes in frequency offset, and longest lock time. Suitable for use in systems with fixed frequency offset and low systematic variation.
		100	Second order, low precision, threshold of 1. Best response to changes in frequency offset and fastest lock time, but lowest precision frequency offset matching. Suitable for use in systems with spread spectrum clocking.
		101	Second order, low precision, threshold of 16. Good response to changes in frequency offset and fast lock time, but low precision frequency offset matching. Suitable for use in systems with spread spectrum clocking.
		110	First order, threshold of 1 with fast lock. Phase offset tracking up to ±1953ppm in the presence of ..10101010.. training pattern, and ±488ppm otherwise.
		111	Second order, low precision with fast lock. As per setting 100, but with improved response to changes in frequency offset when not close to lock.
15-14	LOS		This field enables and disables the Loss of Signal detection.
		0	Disabled. Loss of signal detection disabled.
		01	Reserved.
		10	Enabled. Loss of signal detection enabled.
		11	Reserved.
13-12	ALIGN		Each receiver independently supports two forms of symbol alignment, selectable via the ALIGN field of CFGRX*n.*
		0	Alignment disabled. No symbol alignment will be performed while this setting is selected, or when switching to this selection from another selection.
		01	Comma alignment enabled. Symbol alignment will be performed whenever a misaligned comma symbol is received.
		10	Alignment jog. The symbol alignment will be adjusted by one bit position when this mode is selected.
		11	Reserved.
11	Reserved	0	Reserved.
10-8	TERM	0-7h	Input termination: The only valid value for this field is 001b. Common point set to 0.8 VDDT. This configuration is for AC coupled systems. The transmitter has no effect on the receiver common mode, which is set to optimize the input sensitivity of the receiver. Common mode termination is via a 50pF capacitor to VSSA.
7	INVPAIR		The polarity of RXP*n* and RXN*n* can be inverted by setting the INVPAIR bit of CFGRX*n.*
		0	Normal Polarity RXP*n* considered to be positive data and RXN*n* negative.
		1	Inverted Polarity RXP*n* considered to be negative data and RXN*n* positive.
6-5	RATE		Operating rate. Selects full, half, or quarter rate operation.
		0	Full rate. Two data samples taken per PLL output clock cycle.
		01	Half rate. One data sample taken per PLL output clock cycle.
		10	Quarter rate. One data sample taken every two PLL output clock cycles.
		11	Reserved.
4-2	BUSWIDTH	0-7h	Always write 0 to the bus width field to indicate a 10-bit wide parallel bus to the clock. All other values are reserved.
1	Reserved	0	Reserved.

Table 6. Receiver Configuration Register (CFGRX0/1) Field Descriptions (continued)

Bit	Field	Value	Description
0	ENRX		Enable receiver.
		0	Disables the receiver.
		1	Enables the receiver.

4.3 Transmitter Configuration Register (CFGTX0/1)

The transmitter is configured through CFGTX0/1 registers. (Offsets 0204 90B8h / 0204 90BCh) To enable a transmitter for serialization, set the ENTX bit of the associated CFGTX*n*.

The transmitter descriptor format is shown in Figure 16 and described in Table 7.

Figure 16. Transmitter Configuration Register (CFGTX0/1)

31-24
Reserved
R-0x0

23-17	16
Reserved	ENFTP
R-0x0	R/W-0x0

15-12	11-9	8
DE	SWING	CM
R/W-0x0	R/W-0x0	R/W-0x0

7	6-5	4-2	1	0
INVPAIR	RATE	BUSWIDTH	Reserved	ENTX
R/W-0x0	R/W-0x01	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 7. Transmitter Configuration Register (CFGTX0/1) Field Descriptions

Bit	Field	Value	Description	
31-17	Reserved	0	Reserved.	
16	ENFTP		Enables fixed phase relationship of transmit input clock with respect to transmit output clock. The only valid value for this field is 1.	
15-12	DE		The de-emphasis field provides a means to compensate for high frequency attenuation in the attached media.	
			Amplitude Reduction	
			Low Frequency Gain	**Zero Freq (at e_{27} (min))**
		0	0	0
		1h	4.76	-0.42
		2h	9.52	-0.87
		3h	14.28	-1.34
		4h	19.04	-1.83
		5h	23.8	-2.36
		6h	28.56	-2.92
		7h	33.32	-3.52
		8h	38.08	-4.16
		9h	42.85	-4.86
		Ah	47.61	-5.61
		Bh	52.38	-6.44
		Ch	57.14	-7.35
		Dh	61.9	-8.38
		Eh	66.66	-9.54
		Fh	71.42	-10.87

Table 7. Transmitter Configuration Register (CFGTX0/1) Field Descriptions (continued)

Bit	Field	Value	Description
11-9	SWING		The output swing of each transmitter can be independently set to one of a number of settings by the SWING bits of CFGTX.
			Amplitude (mV$_{\mathbf{dfpp}}$)
		0-3h	Reserved
		4h	750
		5h	1000
		6h	1250
		7h	1375
8	CM		The common mode of the differential signal output on TXP*n* and TXN*n.*
		0	Normal common mode. Common mode not adjusted.
		1	Raised common mode.
7	INVPAIR		The polarity of TXP*n* and TXN*n* can be inverted by setting the INVPAIR bit of CFGTX*n.*
		0	Normal Polarity TXP*n* considered to be positive data and TXN*n* negative.
		1	Inverted Polarity TXP*n* considered to be negative data and TXN*n* positive.
6-5	RATE		Operating rate. Selects full, half, or quarter rate operation.
		0	Full rate. Two data samples taken per PLL output clock cycle.
		01	Half rate. One data sample taken per PLL output clock cycle.
		10	Quarter rate. One data sample taken every two PLL output clock cycles.
		11	Reserved.
4-2	BUSWIDTH		Always write 0 to the bus width field to indicate a 10-bit wide parallel bus to the clock. All other values are reserved.
1	Reserved	0	Reserved.
0	ENTX		Enable transmitter
		0	Enables the transmitter.
		1	Disables the transmitter.

5 3-Port Gigabit Switch (CPSW) Registers

This section describes the memory-mapped registers for the 3-port gigabit switch (CPSW).

Note: Acronyms 3PSW, CPSW, CPSW_3G, and 3pGSw are interchangeable and all refer to the 3 port gigabit switch.

Table 8 lists the memory-mapped registers for the 3-port gigabit switch (CPSW). See the device-specific data manual for the memory address of these registers.

Table 8. 3-Port Gigabit Switch (CPSW) Registers

Offset	Acronym	Register Description	Section
0	CPSW_ID_VER	3pGSw ID Version Register	Section 5.1
4h	CPSW_CONTROL	3pGSw Switch Control Register	Section 5.2
8h	CPSW_SOFT_RESET	3pGSw Soft Reset Register	Section 5.3
Ch	CPSW_STAT_PORT_EN	3pGSw Statistics Port Enable Register	Section 5.4
10h	CPSW_PTYPE	3pGSw Transmit Priority Type Register	Section 5.5
14h	P0_MAX_BLKS	3pGSw Port 0 Maximum FIFO Blocks Register	Section 5.6
18h	P0_BLK_CNT	3pGSw Port 0 FIFO Block Usage Count Register (Read Only)	Section 5.7
1Ch	P0_FLOW_THRESH	3pGSw Port 0 Flow Control Threshold Register	Section 5.8
20h	P0_PORT_VLAN	3pGSw Port 0 VLAN Register	Section 5.9
24h	P0_TX_PRI_MAP	3pGSw Port 0 TX Header Priority to Switch Priority Mapping Register	Section 5.10
28h	GMAC0_GAP_THRESH	3pGSw GMAC0 Short Gap Threshold Register	Section 5.11
2Ch	GMAC0_SA_LO	3pGSw GMAC0 Source Address Low Register	Section 5.12
30h	GMAC0_SA_HI	3pGSw GMAC0 Source Address High Register	Section 5.13
34h	P1_MAX_BLKS	3pGSw Port 1 Maximum FIFO Blocks Register (DM648 only)	Section 5.14
38h	P1_BLK_CNT	3pGSw Port 1 FIFO Block Usage Count Register (Read Only) (DM648 only)	Section 5.15
3Ch	P1_FLOW_THRESH	3pGSw Port 1 Flow Control Threshold Register (DM648 only)	Section 5.16
40h	P1_PORT_VLAN	3pGSw Port 1 VLAN Register (DM648 only)	Section 5.17
44h	P1_TX_PRI_MAP	3pGSw Port 1 TX Header Priority to Switch Priority Mapping Register (DM648 only)	Section 5.18
48h	GMAC1_GAP_THRESH	3pGSw GMAC1 Short Gap Threshold Register (DM648 only)	Section 5.19
4Ch	GMAC1_SA_LO	3pGSw GMAC1 Source Address Low Register (DM648 only)	Section 5.20
50h	GMAC1_SA_HI	3pGSw GMAC1 Source Address High Register (DM648 only)	Section 5.21
54h	P2_MAX_BLKS	3pGSw Port 2 Maximum FIFO Blocks Register	Section 5.22
58h	P2_BLK_CNT	3pGSw Port 2 FIFO Block Usage Count Register (Read Only)	Section 5.23
5Ch	P2_FLOW_THRESH	3pGSw Port 2 Flow Control Threshold Register	Section 5.24
60h	P2_PORT_VLAN	3pGSw Port 2 VLAN Register	Section 5.25
64h	P2_TX_PRI_MAP	3pGSw Port 2 TX (CPDMA RX) Header Priority to Switch Priority Mapping Register	Section 5.26
68h	CPDMA_TX_PRI_MAP	3pGSw CPDMA TX (Port 2 Rx) Packet Priority to Header Priority Mapping Register	Section 5.27
6Ch	CPDMA_RX_CH_MAP	3pGSw CPDMA RX (Port 2 TX) Switch Priority to DMA Channel Mapping Register	Section 5.28
80h	GMAC0_IDVER	GMAC0 ID/Version Register	Section 5.29
84h	GMAC0_MACCONTROL	GMAC0 MAC Control Register	Section 5.30
88h	GMAC0_MACSTATUS	GMAC0 MAC Status Register	Section 5.31
8Ch	GMAC0_SOFT_RESET	GMAC0 Soft Reset Register	Section 5.32
90h	GMAC0_RX_MAXLEN	GMAC0 RX Maximum Length Register	Section 5.33
94h	GMAC0_BOFFTEST	GMAC0 Backoff Test Register	Section 5.34

Table 8. 3-Port Gigabit Switch (CPSW) Registers (continued)

Offset	Acronym	Register Description	Section
A0h	GMAC0_EMCONTROL	GMAC0 Emulation Control Register	Section 5.35
A4h	GMAC0_RX_PRI_MAP	GMAC0 Rx Packet Priority to Header Priority Mapping Register	Section 5.36
C0h	GMAC1_IDVER	GMAC1 ID/Version Register (DM648 only)	Section 5.37
C4h	GMAC1_MACCONTROL	GMAC1 MAC Control Register (DM648 only)	Section 5.38
C8h	GMAC1_MACSTATUS	GMAC1 MAC Status Register (DM648 only)	Section 5.39
CCh	GMAC1_SOFT_RESET	GMAC1 Soft Reset Register (DM648 only)	Section 5.40
D0h	GMAC1_RX_MAXLEN	GMAC1 RX Maximum Length Register (DM648 only)	Section 5.41
D4h	GMAC1_BOFFTEST	GMAC1 Backoff Test Register (DM648 only)	Section 5.42
E0h	GMAC1_EMCONTROL	GMAC1 Emulation Control Register (DM648 only)	Section 5.43
E4h	GMAC1_RX_PRI_MAP	GMAC1 Rx Packet Priority to Header Priority Mapping Register (DM648 only)	Section 5.44
100h	TX_IDVER	CPDMA_REGS TX Identification and Version Register	Section 5.45
104h	TX_CONTROL	CPDMA_REGS TX Control Register	Section 5.46
108h	TX_TEARDOWN	CPDMA_REGS TX Teardown Register	Section 5.47
110h	RX_IDVER	CPDMA_REGS RX Identification and Version Register	Section 5.48
114h	RX_CONTROL	CPDMA_REGS RX Control Register	Section 5.49
118h	RX_TEARDOWN	CPDMA_REGS RX Teardown Register	Section 5.50
11Ch	SOFT_RESET	CPDMA_REGS Soft Reset Register	Section 5.51
120h	DMACONTROL	CPDMA_REGS CPDMA Control Register	Section 5.52
124h	DMASTATUS	CPDMA_REGS CPDMA Status Register	Section 5.53
128h	RX_BUFFER_OFFSET	CPDMA_REGS Receive Buffer Offset Register	Section 5.54
12Ch	EMCONTROL	CPDMA_REGS Emulation Control Register	Section 5.55
180h	TX_INTSTAT_RAW	CPDMA_INT TX Interrupt Status Register (Raw Value)	Section 5.56
184h	TX_INTSTAT_MASKED	CPDMA_INT TX Interrupt Status Register (Masked Value)	Section 5.57
188h	TX_INTMASK_SET	CPDMA_INT TX Interrupt Mask Set Register	Section 5.58
18Ch	TX_INTMASK_CLEAR	CPDMA_INT TX Interrupt Mask Clear Register	Section 5.59
190h	CPDMA_IN_VECTOR	CPDMA_INT Input Vector Register (Read Only)	Section 5.60
194h	CPDMA_EOI_VECTOR	CPDMA_INT End Of Interrupt Vector Register	Section 5.61
1A0h	RX_INTSTAT_RAW	CPDMA_INT RX Interrupt Status Register (Raw Value)	Section 5.62
1A4h	RX_INTSTAT_MASKED	CPDMA_INT RX Interrupt Status Register (Masked Value)	Section 5.63
1A8h	RX_INTMASK_SET	CPDMA_INT RX Interrupt Mask Set Register	Section 5.64
1ACh	RX_INTMASK_CLEAR	CPDMA_INT RX Interrupt Mask Clear Register	Section 5.65
1B0h	DMA_INTSTAT_RAW	CPDMA_INT DMA Interrupt Status Register (Raw Value)	Section 5.66
1B4h	DMA_INTSTAT_MASKED	CPDMA_INT DMA Interrupt Status Register (Masked Value)	Section 5.67
1B8h	DMA_INTMASK_SET	CPDMA_INT DMA Interrupt Mask Set Register	Section 5.68
1BCh	DMA_INTMASK_CLEAR	CPDMA_INT DMA Interrupt Mask Clear Register	Section 5.69
1C0h	RX0_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 0	Section 5.70
1C4h	RX1_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 1	Section 5.70
1C8h	RX2_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 2	Section 5.70
1CCh	RX3_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 3	Section 5.70
1D0h	RX4_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 4	Section 5.70
1D4h	RX5_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 5	Section 5.70
1D8h	RX6_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 6	Section 5.70
1DCh	RX7_PENDTHRESH	CPDMA_INT Receive Threshold Pending Register Channel 7	Section 5.70
1E0h	RX0_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 0	Section 5.71
1E4h	RX1_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 1	Section 5.71

Table 8. 3-Port Gigabit Switch (CPSW) Registers (continued)

Offset	Acronym	Register Description	Section
1E8h	RX2_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 2	Section 5.71
1ECh	RX3_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 3	Section 5.71
1F0h	RX4_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 4	Section 5.71
1F4h	RX5_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 5	Section 5.71
1F8h	RX6_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 6	Section 5.71
1FCh	RX7_FREEBUFFER	CPDMA_INT Receive Free Buffer Register Channel 7	Section 5.71
200h	TX0_HDP	CPDMA_STATERAM TX Channel 0 Head Descriptor Pointer Register	Section 5.72
204h	TX1_HDP	CPDMA_STATERAM TX Channel 1 Head Descriptor Pointer Register	Section 5.72
208h	TX2_HDP	CPDMA_STATERAM TX Channel 2 Head Descriptor Pointer Register	Section 5.72
20Ch	TX3_HDP	CPDMA_STATERAM TX Channel 3 Head Descriptor Pointer Register	Section 5.72
210h	TX4_HDP	CPDMA_STATERAM TX Channel 4 Head Descriptor Pointer Register	Section 5.72
214h	TX5_HDP	CPDMA_STATERAM TX Channel 5 Head Descriptor Pointer Register	Section 5.72
218h	TX6_HDP	CPDMA_STATERAM TX Channel 6 Head Descriptor Pointer Register	Section 5.72
21Ch	TX7_HDP	CPDMA_STATERAM TX Channel 7 Head Descriptor Pointer Register	Section 5.72
220h	RX0_HDP	CPDMA_STATERAM RX Channel 0 Head Descriptor Pointer Register	Section 5.73
224h	RX1_HDP	CPDMA_STATERAM RX Channel 1 Head Descriptor Pointer Register	Section 5.73
228h	RX2_HDP	CPDMA_STATERAM RX Channel 2 Head Descriptor Pointer Register	Section 5.73
22Ch	RX3_HDP	CPDMA_STATERAM RX Channel 3 Head Descriptor Pointer Register	Section 5.73
230h	RX4_HDP	CPDMA_STATERAM RX Channel 4 Head Descriptor Pointer Register	Section 5.73
234h	RX5_HDP	CPDMA_STATERAM RX Channel 5 Head Descriptor Pointer Register	Section 5.73
238h	RX6_HDP	CPDMA_STATERAM RX Channel 6 Head Descriptor Pointer Register	Section 5.73
23Ch	RX7_HDP	CPDMA_STATERAM RX Channel 7 Head Descriptor Pointer Register	Section 5.73
240h	TX0_CP	CPDMA_STATERAM TX Channel 0 Completion Pointer Register	Section 5.74
244h	TX1_CP	CPDMA_STATERAM TX Channel 1 Completion Pointer Register	Section 5.74
248h	TX2_CP	CPDMA_STATERAM TX Channel 2 Completion Pointer Register	Section 5.74
24Ch	TX3_CP	CPDMA_STATERAM TX Channel 3 Completion Pointer Register	Section 5.74
250h	TX4_CP	CPDMA_STATERAM TX Channel 4 Completion Pointer Register	Section 5.74
254h	TX5_CP	CPDMA_STATERAM TX Channel 5 Completion Pointer Register	Section 5.74
258h	TX6_CP	CPDMA_STATERAM TX Channel 6 Completion Pointer Register	Section 5.74
25Ch	TX7_CP	CPDMA_STATERAM TX Channel 7 Completion Pointer Register	Section 5.74

Table 8. 3-Port Gigabit Switch (CPSW) Registers (continued)

Offset	Acronym	Register Description	Section
260h	RX0_CP	CPDMA_STATERAM RX Channel 0 Completion Pointer Register	Section 5.75
264h	RX1_CP	CPDMA_STATERAM RX Channel 1 Completion Pointer Register	Section 5.75
268h	RX2_CP	CPDMA_STATERAM RX Channel 2 Completion Pointer Register	Section 5.75
26Ch	RX3_CP	CPDMA_STATERAM RX Channel 3 Completion Pointer Register	Section 5.75
270h	RX4_CP	CPDMA_STATERAM RX Channel 4 Completion Pointer Register	Section 5.75
274h	RX5_CP	CPDMA_STATERAM RX Channel 5 Completion Pointer Register	Section 5.75
278h	RX6_CP	CPDMA_STATERAM RX Channel 6 Completion Pointer Register	Section 5.75
27Ch	RX7_CP	CPDMA_STATERAM RX Channel 7 Completion Pointer Register	Section 5.75
Statistics Interface Registers			
400h	RXGOODFRAMES	3pGSw_STATS Total Number of Good Frames Received Register	Section 5.76.1
404h	RXBROADCASTFRAMES	3pGSw_STATS Total Number of Good Broadcast Frames Received Register	Section 5.76.2
408h	RXMULTICASTFRAMES	3pGSw_STATS Total Number of Good Multicast Frames Received Register	Section 5.76.3
40Ch	RXPAUSEFRAMES	3pGSw_STATS PauseRxFrames Register	Section 5.76.4
410h	RXCRCERRORS	3pGSw_STATS Total Number of CRC Errors Frames Received Register	Section 5.76.5
414h	RXALIGNCODEERRORS	3pGSw_STATS Total Number of Alignment/Code Errors Received Register	Section 5.76.6
418h	RXOVERSIZEDFRAMES	3pGSw_STATS Total Number of Oversized Frames Received Register	Section 5.76.7
41Ch	RXJABBERFRAMES	3pGSw_STATS Total Number of Jabber Frames Received Register	Section 5.76.8
420h	RXUNDERSIZEDFRAMES	3pGSw_STATS Total Number of Undersized Frames Received Register	Section 5.76.9
424h	RXFRAGMENTS	3pGSw_STATS RxFragments Received Register	Section 5.76.10
430h	RXOCTETS	3pGSw_STATS Total Number of Received Bytes in Good Frames Register	Section 5.76.11
434h	TXGOODFRAMES	3pGSw_STATS GoodTXFrames Register	Section 5.76.12
438h	TXBROADCASTFRAMES	3pGSw_STATS BroadcastTXFrames Register Register	Section 5.76.13
43Ch	TXMULTICASTFRAMES	3pGSw_STATS MulticastTXFrames Register	Section 5.76.14
440h	TXPAUSEFRAMES	3pGSw_STATS PauseTXFrames Register	Section 5.76.15
444h	TXDEFERREDFRAMES	3pGSw_STATS Deferred Frames Register	Section 5.76.16
448h	TXCOLLISIONFRAMES	3pGSw_STATS Collisions Register	Section 5.76.17
44Ch	TXSINGLECOLLFRAMES	3pGSw_STATS Single CollisionTXFrames Register	Section 5.76.18
450h	TXMULTCOLLFRAMES	3pGSw_STATS Multiple CollisionTXFrames Register	Section 5.76.19
454h	TXEXCESSIVECOLLISIONS	3pGSw_STATS Excessive Collisions Register	Section 5.76.20
458h	TXLATECOLLISIONS	3pGSw_STATS Late Collisions Register	Section 5.76.21
45Ch	TXUNDERRUN	3pGSw_STATS Transmit Underrun Error Register	Section 5.76.22
460h	TXCARRIERSENSEERRORS	3pGSw_STATS Carrier Sense Errors Register	Section 5.76.23
464h	TXOCTETS	3pGSw_STATS TXOctets Register	Section 5.76.24
468h	OCTETFRAMES64	3pGSw_STATS 64OctetFrames Register	Section 5.76.25

Table 8. 3-Port Gigabit Switch (CPSW) Registers (continued)

Offset	Acronym	Register Description	Section
46Ch	OCTETFRAMES65T127	3pGSw_STATS 65-127OctetFrames Register	Section 5.76.26
470h	OCTETFRAMES128T255	3pGSw_STATS 128-255OctetFrames Register	Section 5.76.27
474h	OCTETFRAMES256T511	3pGSw_STATS 256-511OctetFrames Register	Section 5.76.28
478h	OCTETFRAMES512T1023	3pGSw_STATS 512-1023OctetFrames Register	Section 5.76.29
47Ch	OCTETFRAMES1024TUP	3pGSw_STATS 1023-1518OctetFrames Register	Section 5.76.30
480h	NETOCTETS	3pGSw_STATS NetOctets Register	Section 5.76.31
484h	RXSOFOVERRUNS	3pGSw_STATS Receive FIFO or DMA Start of Frame Overruns Register	Section 5.76.32
488h	RXMOFOVERRUNS	3pGSw_STATS Receive FIFO or DMA Mid of Frame Overruns Register	Section 5.76.33
48Ch	RXDMAOVERRUNS	3pGSw_STATS Receive DMA Start of Frame and Middle of Frame Overruns Register	Section 5.76.34
		Address Lookup Registers	
500h	ALE_IDVER	Address Lookup Engine ID/Version Register (DM648 only)	Section 5.77
508h	ALE_CONTROL	Address Lookup Engine Control Register (DM648 only)	Section 5.78
510h	ALE_PRESCALE	Address Lookup Engine Prescale Register (DM648 only)	Section 5.79
518h	ALE_UNKNOWN_VLAN	Address Lookup Engine UnKnown VLAN Register (DM648 only)	Section 5.80
520h	ALE_TBLCTL	Address Lookup Engine Table Control Register (DM648 only)	Section 5.81
534h	ALE_TBLW2	Address Lookup Engine Table Word 2 Register (DM648 only)	Section 5.82
538h	ALE_TBLW1	Address Lookup Engine Table Word 1 Register (DM648 only)	Section 5.83
53Ch	ALE_TBLW0	Address Lookup Engine Table Word 0 Register (DM648 only)	Section 5.84
540h	ALE_PORTCTL0	Address Lookup Engine Port 0 Control Register (DM648 only)	Section 5.85
544h	ALE_PORTCTL1	Address Lookup Engine Port 1 Control Register (DM648 only)	Section 5.86
548h	ALE_PORTCTL2	Address Lookup Engine Port 2 Control Register (DM648 only)	Section 5.87

5.1 3pGSw ID Version Register (CPSW_ID_VER)

The 3pGSw ID version register (CPSW_ID_VER) is shown in Figure 17 and described in Table 9.

Figure 17. 3pGSw ID Version Register (CPSW_ID_VER)

31 16
IDENT
R-0x19

15 11	10 8	7 0
RTL_VER	MAJOR_VER	MINOR_VER
R-0x0	R-0x1	R-0x3

LEGEND: R = Read only; -*n* = value after reset

Table 9. 3pGSw ID Version Register (CPSW_ID_VER) Field Descriptions

Bit	Field	Value	Description
31-16	IDENT	0-FFFFh	3pGSw identification value.
15-11	RTL_VER	0-1Fh	3pGSw RTL_version register.
10-8	MAJOR_VER	0-7h	3pGSw major version register.
7-0	MINOR_VER	0-FFh	3pGSw minor version register.

5.2 3pGSw Switch Control Register (CPSW_CONTROL)

The 3pGSw switch control register (CPSW_CONTROL) is shown in Figure 18 and described in Table 10.

Figure 18. 3pGSw Switch Control Register (CPSW_CONTROL)

31–16
Reserved
R-0x0

15–13	12	11	10	9	8
Reserved	P2_PASS_PRI_TAGGED	P1_PASS_PRI_TAGGED	P0_PASS_PRI_TAGGED	VLAN_AWARE	RX_VLAN_ENCAP
R-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

7	6	5	4	3	2	1	0
P2_P2RX_FLOW_EN	P2_P1TX_FLOW_EN	P2_P0TX_FLOW_EN	P1_P2TX_FLOW_EN	P1_P0TX_FLOW_EN	P0_P2TX_FLOW_EN	P0_P1TX_FLOW_EN	FIFO_LOOPBACK
R/W-0x1	R/W-0x1	R/W-0x1	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 10. 3pGSw Switch Control Register (CPSW_CONTROL) Field Descriptions

Bit	Field	Value	Description
31-13	Reserved	0	Reserved.
12	P2_PASS_PRI_TAGGED		Port 2 Pass Priority Tagged.
		0	Priority tagged packets have the zero VID replaced with the input port bit PORT_VID in the P2_PORT_VLAN register.
		1	Priority tagged packets are processed unchanged.
11	P1_PASS_PRI_TAGGED		Port 1 Pass Priority Tagged.
		0	Priority tagged packets have the zero VID replaced with the input port bit PORT_VID in the P1_PORT_VLAN register.
		1	Priority tagged packets are processed unchanged.
10	P0_PASS_PRI_TAGGED		Port 0 Pass Priority Tagged.
		0	Priority tagged packets have the zero VID replaced with the input port bit PORT_VID in the P0_PORT_VLAN register.
		1	Priority tagged packets are processed unchanged.
9	VLAN_AWARE		3pGSw VLAN Aware Mode.
		0	3pGSw is in the VLAN unaware mode.
		1	3pGSw is in the VLAN aware mode.
8	RX_VLAN_ENCAP		Port 2 VLAN Encapsulation.
		0	Port 2 receive packets (from the 3pGSw) are not VLAN encapsulated.
		1	Port 2 receive packets (from the 3pGSw) are VLAN encapsulated.
7	P2_P2RX_FLOW_EN		Port 2 transmit flow control enable from Port 2 receive FIFO.
		0	No port 2 transmit flow control due to receive FIFO availability.
		1	Issue port 2 transmit flow control when port 2 receive FIFO block usage is greater than or equal to the P2_P2RX_THRESH bit value in the P2_P2RX_THRESH register.
6	P2_P1TX_FLOW_EN		Port 2 transmit flow control enable from Port 1 transmit FIFO.
		0	No port 2 transmit flow control due to port 1 transmit FIFO availability.
		1	Issue port 2 transmit flow control when port 1 transmit FIFO block usage is greater than or equal to the P2_P1TX_THRESH bit value in the P2_P2RX_THRESH register.

Table 10. 3pGSw Switch Control Register (CPSW_CONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
5	P2_P0TX_FLOW_EN		Port 2 transmit flow control enable from Port 0 transmit FIFO.
		0	No port 2 transmit flow control due to port 0 transmit FIFO availability.
		1	Issue port 2 transmit flow control when port 0 transmit FIFO block usage is greater than or equal to the P2_P0TX_THRESH bit value in the P2_P2RX_THRESH register.
4	P1_P2TX_FLOW_EN		Port 1 receive flow control enable from Port 2 transmit FIFO.
		0	No port 1 receive flow control due to port 2 transmit FIFO availability.
		1	Issue port 1 receive flow control when port 2 transmit FIFO block usage is greater than or equal to the P1_P2TX_THRESH bit value in the P1_FLOW_THRESH register (the RX_FLOW_EN bit in the GMAC1_MACCONTROL register must also be set).
3	P1_P0TX_FLOW_EN		Port 1 receive flow control enable from Port 0 transmit FIFO.
		0	No port 1 receive flow control due to port 0 transmit FIFO availability.
		1	Issue port 1 receive flow control when port 0 transmit FIFO block usage is greater than or equal to the P1_P0TX_THRESH bit value in the P1_FLOW_THRESH register (the RX_FLOW_EN bit in the GMAC1_MACCONTROL register must also be set).
2	P0_P2TX_FLOW_EN		Port 0 receive flow control enable from Port 2 transmit FIFO.
		0	No port 0 receive flow control due to port 2 transmit FIFO availability.
		1	Issue port 0 receive flow control when port 2 transmit FIFO block usage is greater than or equal to the P0_P2TX_THRESH bit value in the P0_FLOW_THRESH register (the RX_FLOW_EN bit in the GMAC0_MACCONTROL register must also be set).
1	P0_P1TX_FLOW_EN		Port 0 receive flow control enable from Port 1 transmit FIFO.
		0	No port 0 receive flow control due to port 1 transmit FIFO availability.
		1	Issue port 0 receive flow control when port 1 transmit FIFO block usage is greater than or equal to the P0_P1TX_THRESH bit value in the P0_FLOW_THRESH register (the RX_FLOW_EN bit in the GMAC0_MACCONTROL register must also be set).
0	FIFO_LOOPBACK		FIFO Loopback Mode.
		0	Loopback is disabled.
		1	FIFO Loopback mode enabled. Each packet received is turned around and sent out on the same port's transmit path. Port 2 receive is fixed on channel zero. The RXSOFOVERRUNS statistic will increment for every packet sent in FIFO loopback mode.

5.3 3pGSw Soft Reset Register (CPSW_SOFT_RESET)

The 3pGSw soft reset register (CPSW_SOFT_RESET) is shown in Figure 19 and described in Table 11.

Figure 19. 3pGSw Soft Reset Register (CPSW_SOFT_RESET)

31	16
Reserved	
R-0x0	

15	1	0
Reserved		SOFT_RESET
R-0x0		R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 11. 3pGSw Soft Reset Register (CPSW_SOFT_RESET) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved. Read as zero.
0	SOFT_RESET		Software reset.
		0	If a zero is read then reset has occurred.
		1	Writing a one to this bit causes the 3pGSw logic to be reset. After writing a one to this bit, it may be polled to determine if the reset has occurred. If a one is read, the reset has not yet occurred. If a zero is read then reset has occurred.

5.4 3pGSw Statistics Port Enable Register (CPSW_STAT_PORT_EN)

The 3pGSw statistics port enable register (CPSW_STAT_PORT_EN) is shown in Figure 20 and described in Table 12.

Figure 20. 3pGSw Statistics Port Enable Register (CPSW_STAT_PORT_EN)

31	16
Reserved	
R-0x0	

15	3	2	1	0
Reserved		P2_STAT_EN	P1_STAT_EN	P0_STAT_EN
R-0x0		R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 12. 3pGSw Statistics Port Enable Register (CPSW_STAT_PORT_EN) Field Descriptions

Bit	Field	Value	Description
31-3	Reserved	0	Reserved. Read as zero
2	P2_STAT_EN		Port 2 Statistics Enable.
		0	Port 2 statistics are not enabled.
		1	Port 2 statistics are enabled. FIFO overruns (SOFOVERRUNS) are the only port 2 statistics that are enabled to be kept.
1	P1_STAT_EN		Port 1 (GMII 1 and Port 1 FIFO) Statistics Enable.
		0	Port 1 statistics are not enabled.
		1	Port 1 statistics are enabled.
0	P0_STAT_EN		Port 0 (GMII 0 and Port 0 FIFO) Statistics Enable.
		0	Port 0 statistics are not enabled.
		1	Port 0 statistics are enabled.

5.5 3pGSw Transmit Priority Type Register (CPSW_PTYPE)

The 3pGSw transmit priority type register (CPSW_PTYPE) is shown in Figure 21 and described in Table 13.

Figure 21. 3pGSw Transmit Priority Type Register (CPSW_PTYPE)

31–16
Reserved
R-0x0

15–11	10	9	8	7–5	4–0
Reserved	P2_PTYPE_ESC	P1_PTYPE_ESC	P0_PTYPE_ESC	Reserved	ESC_PRI_LD_VAL
R-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 13. 3pGSw Transmit Priority Type Register (CPSW_PTYPE) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved. Read as zero
10	P2_PTYPE_ESC		Port 2 priority type escalate.
		0	Port 2 priority type fixed.
		1	Port 2 priority type escalate.
9	P1_PTYPE_ESC		Port 1 priority type escalate.
		0	Port 1 priority type fixed.
		1	Port 1 priority type escalate.
8	P0_PTYPE_ESC		Port 0 priority type escalate.
		0	Port 0 priority type fixed.
		1	Port 0 priority type escalate.
7-5	Reserved	0	Reserved. Read as zero
4-0	ESC_PRI_LD_VAL	0-1Fh	Escalate priority load value. When a port is in escalate priority, this is the number of higher priority packets sent before the next lower priority is allowed to send a packet. Escalate priority allows lower priority packets to be sent at a fixed rate relative to the next higher priority.

5.6 3pGSw Port 0 Maximum FIFO Blocks Register (P0_MAX_BLKS)

The 3pGSw port 0 maximum FIFO blocks register (P0_MAX_BLKS) is shown in Figure 22 and described in Table 14.

Figure 22. 3pGSw Port 0 Maximum FIFO Blocks Register (P0_MAX_BLKS)

<table>
<tr><td>31</td><td>16</td></tr>
<tr><td colspan="2">Reserved</td></tr>
<tr><td colspan="2">R-0x0</td></tr>
</table>

<table>
<tr><td>15</td><td>9</td><td>8</td><td>4</td><td>3</td><td>0</td></tr>
<tr><td colspan="2">Reserved</td><td colspan="2">P0_TX_MAX_BLKS</td><td colspan="2">P0_RX_MAX_BLKS</td></tr>
<tr><td colspan="2">R-0x0</td><td colspan="2">R/W-0x11</td><td colspan="2">R/W-0x3</td></tr>
</table>

hLEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 14. 3pGSw Port 0 Maximum FIFO Blocks Register (P0_MAX_BLKS) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved. Read as zero.
8-4	P0_TX_MAX_BLKS	0-1Fh	Transmit FIFO Maximum Blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical transmit priority queues. 11h is the recommended value of P0_TX_MAX_BLKS unless the port is in FULLDUPLEX flow control mode. In flow control mode, the P0_RX_MAX_BLKS will need to increase in order to accept the required run out in FULLDUPLEX mode. This value will need to decrease by the amount of increase in P0_RX_MAX_BLKS. Eh is the minimum value TX_MAX_BLKS.
3-0	P0_RX_MAX_BLKS	0-Fh	Receive FIFO Maximum Blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical receive queue. This value must be greater than or equal to 3h. It should be increased In FULLDUPLEX flow control mode to 5h or 6h depending on the required run out space. The P0_TX_MAX_BLKS value must be decreased by the amount of increase in P0_RX_MAX_BLKS. 3h is the minimum value RX_MAX_BLKS and 6h is the maximum value.

5.7 3pGSw Port 0 FIFO Block Usage Count Register (Read Only) (P0_BLK_CNT)

The 3pGSw port 0 FIFO block usage count register (Read Only) (P0_BLK_CNT) is shown in Figure 23 and described in Table 15.

Figure 23. 3pGSw Port 0 FIFO Block Usage Count Register (Read Only) (P0_BLK_CNT)

<table>
<tr><td>31</td><td>16</td></tr>
<tr><td colspan="2">Reserved</td></tr>
<tr><td colspan="2">R-0x0</td></tr>
</table>

<table>
<tr><td>15</td><td>9</td><td>8</td><td>4</td><td>3</td><td>0</td></tr>
<tr><td colspan="2">Reserved</td><td colspan="2">TX_BLK_COUNT</td><td colspan="2">RX_BLK_COUNT</td></tr>
<tr><td colspan="2">R-0x0</td><td colspan="2">R-0x04</td><td colspan="2">R-0x01</td></tr>
</table>

LEGEND: R = Read only; -*n* = value after reset

Table 15. 3pGSw Port 0 FIFO Block Usage Count Register (Read Only) (P0_BLK_CNT) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved. Read as zero.
8-4	TX_BLK_COUNT	0-1Fh	Port 0 transmit block count usage. This value is the number of blocks allocated to the FIFO logical transmit queues.
3-0	RX_BLK_COUNT	0-Fh	Port 0 receive block count usage. This value is the number of blocks allocated to the FIFO logical receive queues.

5.8 3pGSw Port 0 Flow Control Threshold Register (P0_FLOW_THRESH)

The 3pGSw port 0 flow control threshold register (P0_FLOW_THRESH) is shown in Figure 24 and described in Table 16.

Figure 24. 3pGSw Port 0 Flow Control Threshold Register (P0_FLOW_THRESH)

<table>
<tr><td>31</td><td>16</td></tr>
<tr><td colspan="2">Reserved</td></tr>
<tr><td colspan="2">R-0x0</td></tr>
</table>

<table>
<tr><td>15</td><td>13</td><td>12</td><td>8</td><td>7</td><td>5</td><td>4</td><td>0</td></tr>
<tr><td colspan="2">Reserved</td><td colspan="2">P0_P2TX_THRESH</td><td colspan="2">Reserved</td><td colspan="2">P0_P1TX_THRESH</td></tr>
<tr><td colspan="2">R-0x0</td><td colspan="2">R/W-0x0B</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x0B</td></tr>
</table>

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 16. 3pGSw Port 0 Flow Control Threshold Register (P0_FLOW_THRESH) Field Descriptions

Bit	Field	Value	Description
31-13	Reserved	0	Reserved. Read as zero.
12-8	P0_P2TX_THRESH	0-1Fh	Port 2 transmit FIFO threshold value to trigger Port 0 receive flow control
7-5	Reserved	0	Reserved. Read as zero.
4-0	P0_P1TX_THRESH	0-1Fh	Port 1 transmit FIFO threshold value to trigger Port 0 receive flow control

5.9 3pGSw Port 0 VLAN Register (P0_PORT_VLAN)

The 3pGSw port 0 VLAN register (P0_PORT_VLAN) is shown in Figure 25 and described in Table 17.

Figure 25. 3pGSw Port 0 VLAN Register (P0_PORT_VLAN)

31 … 16
Reserved
R-0x0

15 … 13	12	11 … 0
PORT_PRI	PORT_CFI	PORT_VID
R/W-0x0	R/W-0x0	R-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 17. 3pGSw Port 0 VLAN Register (P0_PORT_VLAN) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15-13	PORT_PRI	0-7h	Port VLAN priority (7 is highest priority).
12	PORT_CFI	0-1	Port CFI bit.
11-0	PORT_VID	0-FFFh	Port VLAN ID.

5.10 3pGSw Port 0 TX Header Priority to Switch Priority Mapping Register (P0_TX_PRI_MAP)

The 3pGSw port 0 TX header priority to switch priority mapping register (P0_TX_PRI_MAP) is shown in Figure 26 and described in Table 18.

Figure 26. 3pGSw Port 0 TX Header Priority to Switch Priority Mapping Register (P0_TX_PRI_MAP)

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
Reserved	PRI7	Reserved	PRI6	Reserved	PRI5	Reserved	PRI4
R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
Reserved	PRI3	Reserved	PRI2	Reserved	PRI1	Reserved	PRI0
R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 18. 3pGSw Port 0 TX Header Priority to Switch Priority Mapping Register (P0_TX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31-30	Reserved	0	Reserved.
29-28	PRI7	0-3h	Priority 7. A packet header priority of 7h is given to this switch queue priority.
27-26	Reserved	0	Reserved.
25-24	PRI6	0-3h	Priority 6. A packet header priority of 6h is given to this switch queue priority.
23-22	Reserved	0	Reserved.
21-20	PRI5	0-3h	Priority 5. A packet header priority of 5h is given to this switch queue priority.
19-18	Reserved	0	Reserved.
17-16	PRI4	0-3h	Priority 4. A packet header priority of 4h is given to this switch queue priority.
15-14	Reserved	0	Reserved.
13-12	PRI3	0-3h	Priority 3. A packet header priority of 3h is given to this switch queue priority.
11-10	Reserved	0	Reserved.
9-8	PRI2	0-3h	Priority 2. A packet header priority of 2h is given to this switch queue priority.
7-6	Reserved	0	Reserved.
5-4	PRI1	0-3h	Priority 1. A packet header priority of 1h is given to this switch queue priority.
3-2	Reserved	0	Reserved.
1-0	PRI0	0-3h	Priority 0. A packet header priority of 0 is given to this switch queue priority.

5.11 3pGSw GMAC0 Short Gap Threshold Register (GMAC0_GAP_THRESH)

The 3pGSw GMAC0 short gap threshold register (GMAC0_GAP_THRESH) is shown in Figure 27 and described in Table 19.

Figure 27. 3pGSw GMAC0 Short Gap Threshold Register (GMAC0_GAP_THRESH)

31 16
Reserved
R-0x0

15 5	4 0
Reserved	P0_GAP_THRESH
R-0x0	R/W-0x0B

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 19. 3pGSw GMAC0 Short Gap Threshold Register (GMAC0_GAP_THRESH) Field Descriptions

Bit	Field	Value	Description
31-5	Reserved	0	Reserved.
4-0	P0_GAP_THRESH	0-1Fh	Port 0 short gap threshold.4.

5.12 3pGSw GMAC0 Source Address Low Register (GMAC0_SA_LO)

The 3pGSw GMAC0 source address low register (GMAC0_SA_LO) is shown in Figure 28 and described in Table 20.

Figure 28. 3pGSw GMAC0 Source Address Low Register (GMAC0_SA_LO)

31 16
Reserved
R-0x0

15 8	7 0
MACSRCADDR_0	MACSRCADDR_1
R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 20. 3pGSw GMAC0 Source Address Low Register (GMAC0_SA_LO) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-8	MACSRCADDR_0	0-FFh	Source address lower 8 bits (byte 0).
7-0	MACSRCADDR_1	0-FFh	Source address bits 15:8 (byte 1).

5.13 3pGSw GMAC0 Source Address High Register (GMAC0_SA_HI)

The 3pGSw GMAC0 source address high register (GMAC0_SA_HI) is shown in Figure 29 and described in Table 21.

Figure 29. 3pGSw GMAC0 Source Address High Register (GMAC0_SA_HI)

31 24	23 16
MACSRCADDR_2	MACSRCADDR_3
R/W-0x0	R/W-0x0

15 8	7 0
MACSRCADDR_4	MACSRCADDR_5
R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 21. 3pGSw GMAC0 Source Address High Register (GMAC0_SA_HI) Field Descriptions

Bit	Field	Value	Description
31-24	MACSRCADDR_2	0-FFh	Source address bits 23:16 (byte 2).
23-16	MACSRCADDR_3	0-FFh	Source address bits 31:24 (byte 3).
15-8	MACSRCADDR_4	0-FFh	Source address bits 39:32 (byte 4).
7-0	MACSRCADDR_5	0-FFh	Source address bits 47:40 (byte 5).

5.14 3pGSw Port 1 Maximum FIFO Blocks Register (DM648 only) (P1_MAX_BLKS)

The 3pGSw port 1 maximum FIFO blocks register (P1_MAX_BLKS) is shown in Figure 30 and described in Table 22.

Figure 30. 3pGSw Port 1 Maximum FIFO Blocks Register (DM648 only) (P1_MAX_BLKS)

31 ... 16
Reserved
R-0x0

15 ... 9	8 ... 4	3 ... 0
Reserved	TX_MAX_BLKS	RX_MAX_BLKS
R-0x0	R/W-0x11	R/W-0x3

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 22. 3pGSw Port 1 Maximum FIFO Blocks Register (DM648 only) (P1_MAX_BLKS) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved. Read as zero.
8-4	TX_MAX_BLKS	0-1Fh	Transmit FIFO maximum blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical transmit priority queues. 11h is the recommended value of TX_MAX_BLKS unless the port is in FULLDUPLEX flow control mode. In flow control mode, the RX_MAX_BLKS will need to increase in order to accept the required run out in FULLDUPLEX mode. This value will need to decrease by the amount of increase in RX_MAX_BLKS. Eh is the minimum value for the TX_MAX_BLKS bit.
3-0	RX_MAX_BLKS	0-Fh	Receive FIFO Maximum Blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical receive queue. This value must be greater than or equal to 3h. It should be increased In FULLDUPLEX flow control mode to 5h or 6h depending on the required runout space. The TX_MAX_BLKS value must be decreased by the amount of increase in RX_MAX_BLKS. 3h is the minimum value RX_MAX_BLKS and 6h is the maximum value.

5.15 3pGSw Port 1 FIFO Block Usage Count Register (Read Only) (DM648 only) (P1_BLK_CNT)

The 3pGSw port 1 FIFO block usage count register (Read Only) (P1_BLK_CNT) is shown in Figure 31 and described in Table 23.

Figure 31. 3pGSw Port 1 FIFO Block Usage Count Register (Read Only) (DM648 only) (P1_BLK_CNT)

31 16
Reserved
R-0x0

15 9	8 4	3 0
Reserved	TX_BLK_COUNT	RX_BLK_COUNT
R-0x0	R-0x04	R-0x01

LEGEND: R = Read only; -*n* = value after reset

Table 23. 3pGSw Port 1 FIFO Block Usage Count Register (Read Only) (DM648 only) (P1_BLK_CNT) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved.
8-4	TX_BLK_COUNT	0-1Fh	Port 1 transmit block count usage. This value is the number of blocks allocated to the FIFO logical transmit queues.
3-0	RX_BLK_COUNT	0-Fh	Port 1 receive block count usage. This value is the number of blocks allocated to the FIFO logical receive queues.

5.16 3pGSw Port 1 Flow Control Threshold Register (DM648 only) (P1_FLOW_THRESH)

The 3pGSw port 1 flow control threshold register (P1_FLOW_THRESH) is shown in Figure 32 and described in Table 24.

Figure 32. 3pGSw Port 1 Flow Control Threshold Register (DM648 only) (P1_FLOW_THRESH)

31 16
Reserved
R-0x0

15 13	12 8	7 5	4 0
Reserved	P1_P2TX_THRESH	Reserved	P1_P0TX_THRESH
R-0x0	R/W-0x0B	R-0x0	R/W-0x0B

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 24. 3pGSw Port 1 Flow Control Threshold Register (DM648 only) (P1_FLOW_THRESH) Field Descriptions

Bit	Field	Value	Description
31-13	Reserved	0	Reserved.
12-8	P1_P2TX_THRESH	0-1Fh	Port 2 transmit FIFO threshold value to trigger port 1 receive flow control.
7-5	Reserved	0	Reserved.
4-0	P1_P0TX_THRESH	0-1Fh	Port 0transmit FIFO threshold value to trigger port 1 receive flow control.

5.17 3pGSw Port 1 VLAN Register (DM648 only) (P1_PORT_VLAN)

The 3pGSw port 1 VLAN register (P1_PORT_VLAN) is shown in Figure 33 and described in Table 25.

Figure 33. 3pGSw Port 1 VLAN Register (DM648 only) (P1_PORT_VLAN)

31	16
Reserved	
R-0x0	

15 13	12	11 0
PORT_PRI	PORT_CFI	PORT_VID
R/W-0x0	R/W-0x0	R-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 25. 3pGSw Port 1 VLAN Register (DM648 only) (P1_PORT_VLAN) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15-13	PORT_PRI	0-7h	Port VLAN priority (7 is highest priority).
12	PORT_CFI	0-1	Port CFI bit.
11-0	PORT_VID	0-FFFh	Port VLAN ID.

5.18 3pGSw Port 1 TX Header Priority to Switch Priority Mapping Register (DM648 only) (P1_TX_PRI_MAP)

The 3pGSw port 1 TX header priority to switch priority mapping register (P1_TX_PRI_MAP) is shown in Figure 34 and described in Table 26.

Figure 34. 3pGSw Port 1 TX Header Priority to Switch Priority Mapping Register (DM648 only) (P1_TX_PRI_MAP)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="2">Reserved</td><td colspan="2">PRI7</td><td colspan="2">Reserved</td><td colspan="2">PRI6</td><td colspan="2">Reserved</td><td colspan="2">PRI5</td><td colspan="2">Reserved</td><td colspan="2">PRI4</td></tr>
<tr><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">Reserved</td><td colspan="2">PRI3</td><td colspan="2">Reserved</td><td colspan="2">PRI2</td><td colspan="2">Reserved</td><td colspan="2">PRI1</td><td colspan="2">Reserved</td><td colspan="2">PRI0</td></tr>
<tr><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td><td colspan="2">R-0x0</td><td colspan="2">R/W-0x3</td></tr>
</table>

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 26. 3pGSw Port 1 TX Header Priority to Switch Priority Mapping Register (DM648 only) (P1_TX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31-30	Reserved	0	Reserved.
29-28	PRI7	0-3h	Priority 7. A packet header priority of 7h is given to this switch queue priority.
27-26	Reserved	0	Reserved.
25-24	PRI6	0-3h	Priority 6. A packet header priority of 6h is given to this switch queue priority.
23-22	Reserved	0	Reserved.
21-20	PRI5	0-3h	Priority 5. A packet header priority of 5h is given to this switch queue priority.
19-18	Reserved	0	Reserved.
17-16	PRI4	0-3h	Priority 4. A packet header priority of 4h is given to this switch queue priority.
15-14	Reserved	0	Reserved.
13-12	PRI3	0-3h	Priority 3. A packet header priority of 3h is given to this switch queue priority.
11-10	Reserved	0	Reserved.
9-8	PRI2	0-3h	Priority 2. A packet header priority of 2h is given to this switch queue priority.
7-6	Reserved	0	Reserved.
5-4	PRI1	0-3h	Priority 1. A packet header priority of 1h is given to this switch queue priority.
3-2	Reserved	0	Reserved.
1-0	PRI0	0-3h	Priority 0. A packet header priority of 0 is given to this switch queue priority.

5.19 3pGSw GMAC1 Short Gap Threshold Register (DM648 only) (GMAC1_GAP_THRESH)

The 3pGSw GMAC1 short gap threshold register (GMAC1_GAP_THRESH) is shown in Figure 35 and described in Table 27.

Figure 35. 3pGSw GMAC1 Short Gap Threshold Register (DM648 only) (GMAC1_GAP_THRESH)

31 16
Reserved
R-0x0

15 5	4 0
Reserved	P1_GAP_THRESH
R-0x0	R/W-0x0B

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 27. 3pGSw GMAC1 Short Gap Threshold Register (DM648 only) (GMAC1_GAP_THRESH) Field Descriptions

Bit	Field	Value	Description
31-5	Reserved	0	Reserved.
4-0	P1_GAP_THRESH	0-1Fh	Port 1 short gap threshold.

5.20 3pGSw GMAC1 Source Address Low Register (DM648 only) (GMAC1_SA_LO)

The 3pGSw GMAC1 source address low register (GMAC1_SA_LO) is shown in Figure 36 and described in Table 28.

Figure 36. 3pGSw GMAC1 Source Address Low Register (DM648 only) (GMAC1_SA_LO)

31 16
Reserved
R-0x0

15 8	7 0
MACSRCADDR_0	MACSRCADDR_1
R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 28. 3pGSw GMAC1 Source Address Low Register (DM648 only) (GMAC1_SA_LO) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-8	MACSRCADDR_0	0-FFh	Source address lower 8 bits (byte 0).
7-0	MACSRCADDR_1	0-FFh	Source address bits 15:8 (byte 1).

5.21 3pGSw GMAC1 Source Address High Register (DM648 only) (GMAC1_SA_HI)

The 3pGSw GMAC1 source address high register (GMAC1_SA_HI) is shown in Figure 37 and described in Table 29.

Figure 37. 3pGSw GMAC1 Source Address High Register (DM648 only) (GMAC1_SA_HI)

31 24	23 16
MACSRCADDR_2	MACSRCADDR_3
R/W-0x0	R/W-0x0

15 8	7 0
MACSRCADDR_4	MACSRCADDR_5
R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 29. 3pGSw GMAC1 Source Address High Register (DM648 only) (GMAC1_SA_HI) Field Descriptions

Bit	Field	Value	Description
31-24	MACSRCADDR_2	0-FFh	Source address bits 23:16 (byte 2).
23-16	MACSRCADDR_3	0-FFh	Source address bits 31:24 (byte 3).
15-8	MACSRCADDR_4	0-FFh	Source address bits 39:32 (byte 4).
7-0	MACSRCADDR_5	0-FFh	Source address bits 47:40 (byte 5).

5.22 3pGSw Port 2 Maximum FIFO Blocks Register (P2_MAX_BLKS)

The 3pGSw port 2 maximum FIFO blocks register (P2_MAX_BLKS) is shown in Figure 38 and described in Table 30.

Figure 38. 3pGSw Port 2 Maximum FIFO Blocks Register (P2_MAX_BLKS)

31 16
Reserved
R-0x0

15 9	8 4	3 0
Reserved	TX_MAX_BLKS	RX_MAX_BLKS
R-0x0	R/W-0x11	R/W-0x3

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 30. 3pGSw Port 2 Maximum FIFO Blocks Register (P2_MAX_BLKS) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved. Read as zero.
8-4	TX_MAX_BLKS	0-1Fh	Transmit FIFO maximum blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical transmit priority queues. 10h is the recommended value of TX_MAX_BLKS. Port 2 should remain in flow control mode. Eh is the minimum value TX_MAX_BLKS.
3-0	RX_MAX_BLKS	0-Fh	Receive FIFO maximum blocks. This value is the maximum number of 1k memory blocks that may be allocated to the FIFO's logical receive queue. 4h is the recommended value. 3h is the minimum value RX_MAX_BLKS and 6h is the maximum value.

5.23 3pGSw Port 2 FIFO Block Usage Count Register (Read Only) (P2_BLK_CNT)

The 3pGSw port 2 FIFO block usage count register (Read Only) (P2_BLK_CNT) is shown in Figure 39 and described in Table 31.

Figure 39. 3pGSw Port 2 FIFO Block Usage Count Register (Read Only) (P2_BLK_CNT)

31 - 16
Reserved
R-0x0

15 - 9	8 - 4	3 - 0
Reserved	TX_BLK_COUNT	RX_BLK_COUNT
R-0x0	R-0x04	R-0x01

LEGEND: R = Read only; -*n* = value after reset

Table 31. 3pGSw Port 2 FIFO Block Usage Count Register (Read Only) (P2_BLK_CNT) Field Descriptions

Bit	Field	Value	Description
31-9	Reserved	0	Reserved.
8-4	TX_BLK_COUNT	0-1Fh	Port 2 transmit block count usage. This value is the number of blocks allocated to the FIFO logical transmit queues.
3-0	RX_BLK_COUNT	0-Fh	Port 2 receive block count usage. This value is the number of blocks allocated to the FIFO logical receive queues.

5.24 3pGSw Port 2 Flow Control Threshold Register (P2_FLOW_THRESH)

The 3pGSw port 2 flow control threshold register (P2_FLOW_THRESH) is shown in Figure 40 and described in Table 32.

Figure 40. 3pGSw Port 2 Flow Control Threshold Register (P2_FLOW_THRESH)

31 - 20	19 - 16
Reserved	P2_P2RX_THRESH
R-0x0	R/W-0x03

15 - 13	12 - 8	7 - 5	4 - 0
Reserved	P2_P1TX_THRESH	Reserved	P2_P0TX_THRESH
R-0x0	R/W-0x10	R-0x0	R/W-0x10

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 32. 3pGSw Port 2 Flow Control Threshold Register (P2_FLOW_THRESH) Field Descriptions

Bit	Field	Value	Description
31-20	Reserved	0	Reserved.
19-16	P2_P2RX_THRESH	0-Fh	Port 2 receive FIFO threshold value to trigger port 2 transmit flow control. 3h is the recommended value.
15-13	Reserved	0	Reserved.
12-8	P2_P1TX_THRESH	0-1Fh	Port 1 transmit FIFO threshold value to trigger port 2 transmit flow control. Bh is the recommended value.
7-5	Reserved	0	Reserved.
4-0	P2_P0TX_THRESH	0-1Fh	Port 0 transmit FIFO threshold value to trigger port 2 transmit flow control. Bh is the recommended value.

5.25 3pGSw Port 2 VLAN Register (P2_PORT_VLAN)

The 3pGSw port 2 VLAN register (P2_PORT_VLAN) is shown in Figure 41 and described in Table 33.

Figure 41. 3pGSw Port 2 VLAN Register (P2_PORT_VLAN)

31 … 16
Reserved
R-0x0

15 … 13	12	11 … 0
PORT_PRI	PORT_CFI	PORT_VID
R/W-0x0	R/W-0x0	R-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 33. 3pGSw Port 2 VLAN Register (P2_PORT_VLAN) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15-13	PORT_PRI	0-7h	Port VLAN priority (7 is highest priority).
12	PORT_CFI	0-1	Port CFI bit.
11-0	PORT_VID	0-FFFh	Port VLAN ID.

5.26 3pGSw Port 2 TX (CPDMA RX) Header Priority to Switch Priority Mapping Register (P2_TX_PRI_MAP)

The 3pGSw port 2 TX (CPDMA RX) header priority to switch priority mapping register (P2_TX_PRI_MAP) is shown in Figure 42 and described in Table 34.

Figure 42. 3pGSw Port 2 TX (CPDMA RX) Header Priority to Switch Priority Mapping Register (P2_TX_PRI_MAP)

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
Reserved	PRI7	Reserved	PRI6	Reserved	PRI5	Reserved	PRI4
R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
Reserved	PRI3	Reserved	PRI2	Reserved	PRI1	Reserved	PRI0
R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3	R-0x0	R/W-0x3

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 34. 3pGSw Port 2 TX (CPDMA RX) Header Priority to Switch Priority Mapping Register (P2_TX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31-30	Reserved	0	Reserved.
29-28	PRI7	0-3h	Priority 7. A packet header priority of 7h is given to this switch queue priority.
27-26	Reserved	0	Reserved.
25-24	PRI6	0-3h	Priority 6. A packet header priority of 6h is given to this switch queue priority.
23-22	Reserved	0	Reserved.
21-20	PRI5	0-3h	Priority 5. A packet header priority of 5h is given to this switch queue priority.
19-18	Reserved	0	Reserved.
17-16	PRI4	0-3h	Priority 4. A packet header priority of 4h is given to this switch queue priority.
15-14	Reserved	0	Reserved.
13-12	PRI3	0-3h	Priority 3. A packet header priority of 3h is given to this switch queue priority.
11-10	Reserved	0	Reserved.
9-8	PRI2	0-3h	Priority 2. A packet header priority of 2h is given to this switch queue priority.
7-6	Reserved	0	Reserved.
5-4	PRI1	0-3h	Priority 1. A packet header priority of 1h is given to this switch queue priority.
3-2	Reserved	0	Reserved.
1-0	PRI0	0-3h	Priority 0. A packet header priority of 0 is given to this switch queue priority.

5.27 3pGSw CPDMA TX (Port 2 Rx) Packet Priority to Header Priority Mapping Register (CPDMA_TX_PRI_MAP)

The 3pGSw CPDMA TX (Port 2 Rx) packet priority to header priority mapping register (CPDMA_TX_PRI_MAP) is shown in Figure 43 and described in Table 35.

Figure 43. 3pGSw CPDMA TX (Port 2 Rx) Packet Priority to Header Priority Mapping Register (CPDMA_TX_PRI_MAP)

31	30 28	27	26 24	23	22 20	19	18 16
Reserved	PRI7	Reserved	PRI6	Reserved	PRI5	Reserved	PRI4
R-0x0	R/W-0x7	R-0x0	R/W-0x6	R-0x0	R/W-0x5	R-0x0	R/W-0x4

15	14 12	11	10 8	7	6 4	3	2 0
Reserved	PRI3	Reserved	PRI2	Reserved	PRI1	Reserved	PRI0
R-0x0	R/W-0x3	R-0x0	R/W-0x2	R-0x0	R/W-0x1	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 35. 3pGSw CPDMA TX (Port 2 Rx) Packet Priority to Header Priority Mapping Register (CPDMA_TX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-28	PRI7	0-7h	Priority 7. A packet priority of 7h is given to this header packet priority.
27	Reserved	0	Reserved.
26-24	PRI6	0-7h	Priority 6. A packet priority of 6h is given to this header queue priority.
23	Reserved	0	Reserved.
22-20	PRI5	0-7h	Priority 5. A packet priority of 5h is given to this header queue priority.
19	Reserved	0	Reserved.
18-16	PRI4	0-7h	Priority 4. A packet priority of 4h is given to this header queue priority.
15	Reserved	0	Reserved.
14-12	PRI3	0-7h	Priority 3. A packet priority of 3h is given to this header queue priority.
11	Reserved	0	Reserved.
10-8	PRI2	0-7h	Priority 2. A packet priority of 2h is given to this header queue priority.
7	Reserved	0	Reserved.
6-4	PRI1	0-7h	Priority 1. A packet priority of 1h is given to this header queue priority.
3	Reserved	0	Reserved.
2-0	PRI0	0-7h	Priority 0. A packet priority of 0 is given to this header queue priority.

5.28 3pGSw CPDMA RX (Port 2 TX) Switch Priority to DMA Channel Mapping Register (CPDMA_RX_CH_MAP)

The 3pGSw CPDMA RX (Port 2 TX) switch priority to DMA channel mapping register (CPDMA_RX_CH_MAP) is shown in Figure 44 and described in Table 36.

Figure 44. 3pGSw CPDMA RX (Port 2 TX) Switch Priority to DMA Channel Mapping Register (CPDMA_RX_CH_MAP)

31	30 28	27	26 24	23	22 20	19	18 16
Reserved	SU_CH3	Reserved	SU_CH2	Reserved	SU_CH1	Reserved	SU_CH0
R-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0

15	14 12	11	10 8	7	6 4	3	2 0
Reserved	CH3	Reserved	CH2	Reserved	CH1	Reserved	CH0
R-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 36. 3pGSw CPDMA RX (Port 2 TX) Switch Priority to DMA Channel Mapping Register (CPDMA_RX_CH_MAP) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-28	SU_CH3	0-7h	Channel 3. Supervisory packet of switch priority 3h is mapped to this CPDMA RX channel.
27	Reserved	0	Reserved.
26-24	SU_CH2	0-7h	Channel 2. Supervisory packet of switch priority 2h is mapped to this CPDMA RX channel.
23	Reserved	0	Reserved.
22-20	SU_CH1	0-7h	Channel 1. Supervisory packet of switch priority 1h is mapped to this CPDMA RX channel.
19	Reserved	0	Reserved.
18-16	SU_CH0	0-7h	Channel 0. Supervisory packet of switch priority 0 is mapped to this CPDMA RX channel.
15	Reserved	0	Reserved.
14-12	CH3	0-7h	Channel 3. Switch priority of 3h is mapped to this CPDMA Rx channel.
11	Reserved	0	Reserved.
10-8	CH2	0-7h	Channel 2. Switch priority of 2h is mapped to this CPDMA Rx channel.
7	Reserved	0	Reserved.
6-4	CH1	0-7h	Channel 1. Switch priority of 1h is mapped to this CPDMA Rx channel.
3	Reserved	0	Reserved.
2-0	CH0	0-7h	Channel 0. Switch priority of 0 is mapped to this CPDMA Rx channel.

5.29 GMAC0 ID/Version Register (GMAC0_IDVER)

The GMAC0 ID/Version register (GMAC0_IDVER) is shown in Figure 45 and described in Table 37.

Figure 45. GMAC0 ID/Version Register (GMAC0_IDVER)

31 16
RX_IDENT
R-0x17

15 11	10 8	7 0
RX_Z	RX_X	RX_Y
R-0x00	R-0x1	R-0x0A

LEGEND: R = Read only; -*n* = value after reset

Table 37. GMAC0 ID/Version Register (GMAC0_IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	RX_IDENT	0-FFFFh	RX identification value.
15-11	RX_Z	0-1Fh	RX Z value.
10-8	RX_X	0-7h	Rx X value.
7-0	RX_Y	0-FFh	Rx Y value.

5.30 GMAC0 MAC Control Register (GMAC0_MACCONTROL)

The GMAC0 MAC control register (GMAC0_MACCONTROL) is shown in Figure 46 and described in Table 38.

Figure 46. GMAC0 MAC Control Register (GMAC0_MACCONTROL)

31 25	24
Reserved	RX_CMF_EN
R-0x0	R/W-0x00

23	22	21 19	18	17	16
RX_CSF_EN	RX_CEF_EN	Reserved	CTL_EN	GIG_FORCE	IFCTL_B
R/W-0x00	R/W-0x00	R-0x0	R/W-0x00	R/W-0x00	R/W-0x00

15	14 12	11	10	9 8
IFCTL_A	Reserved	CMD_IDLE	TX_SHORT_GAP_EN	Reserved
R/W-0x00	R-0x0	R/W-0x00	R/W-0x00	R-0x0

7	6	5	4	3	2	1	0
GIG	TX_PACE	GMII_EN	TX_FLOW_EN	RX_FLOW_EN	MTEST	LOOPBACK	FULLDUPLEX
R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 38. GMAC0 MAC Control Register (GMAC0_MACCONTROL) Field Descriptions

Bit	Field	Value	Description
31-25	Reserved	0	Reserved.
24	RX_CMF_EN		RX copy MAC control frames enable. Enables MAC control frames to be transferred to memory. MAC control frames are normally acted upon (if enabled), but not copied to memory. MAC control frames that are pause frames will be acted upon if enabled in the GMAC*n*_MACCONTROL register, regardless of the value of RX_CMF_EN. Frames transferred to memory due to RX_CMF_EN will have the control bit set in their EOP buffer descriptor.
		0	MAC control frames are filtered (but acted upon if enabled).
		1	MAC control frames are transferred to memory.
23	RX_CSF_EN		RX copy short frames enable. Enables frames or fragments shorter than 64 bytes to be copied to memory. Frames transferred to memory due to RX_CSF_EN will have the fragment or undersized bit set in their EOP buffer descriptor. Fragments are short frames that contain CRC/align/code errors and undersized are short frames without errors.
		0	Short frames are filtered.
		1	Short frames are transferred to memory.
22	RX_CEF_EN		RX copy error frames enable. Enables frames containing errors to be transferred to memory. The appropriate error bit will be set in the frame EOP buffer descriptor. Frames containing errors will be filtered when RX_CEF_EN is not set.
		0	Frames containing errors are filtered.
		1	Frames containing errors are transferred to memory.
21-19	Reserved	0	Reserved.
18	CTL_EN		Control enable. Enables the fullduplex and gigabit mode to be selected from the FULLDUPLEX_IN and GIG_IN input signals and not from the FULLDUPLEX and GIG bits in this register. The FULLDUPLEX_MODE bit reflects the actual fullduplex mode selected.
		0	
		1	

Table 38. GMAC0 MAC Control Register (GMAC0_MACCONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
17	GIG_FORCE		Gigabit mode force. This bit is used to force the GMAC into gigabit mode.
		0	
		1	
16	IFCTL_B		Interface control B. Intended as a general purpose output bit to be used to control external gaskets associated with the GMII (GMII to RGMII).
		0	
		1	
15	IFCTL_A		Interface control A. Intended as a general purpose output bit to be used to control external gaskets associated with the GMII (GMII to RGMII).
		0	
		1	
14-12	Reserved	0	Reserved.
11	CMD_IDLE		Command idle.
		0	Idle not commanded.
		1	Idle commanded (read idle in MACStatus).
10	TX_SHORT_GAP_EN		Transmit short gap enable
		0	Transmit with a short IPG is disabled.
		1	Transmit with a short IPG (when TX_SHORT_GAP input is asserted) is enabled.
9-8	Reserved	0	Reserved.
7	GIG		Gigabit mode.
		0	10/100 mode.
		1	Gigabit mode (full duplex only).
6	TX_PACE		Transmit pacing
		0	Transmit pacing disabled.
		1	Transmit pacing enabled.
5	GMII_EN		GMII enable.
		0	GMII RX and TX held in reset.
		1	GMII RX and TX released from reset.
4	TX_FLOW_EN		Transmit flow control enable. Determines if incoming pause frames are acted upon in full-duplex mode. Incoming pause frames are not acted upon in half-duplex mode regardless of this bit setting. The RX_MBP_Enable bits determine whether or not received pause frames are transferred to memory.
		0	Transmit flow control disabled. Full-duplex mode. Incoming pause frames are not acted upon.
		1	Transmit flow control enabled. Full-duplex mode. Incoming pause frames are acted upon.
3	RX_FLOW_EN		Receive flow control enable.
		0	Receive flow control disabled Half-duplex mode. No flow control generated collisions are sent. Full-duplex mode. No outgoing pause frames are sent.
		1	Receive flow control enabled Half-duplex mode. Collisions are initiated when receive flow control is triggered. Full-duplex mode. Outgoing pause frames are sent when receive flow control is triggered.
2	MTEST		Manufacturing test mode. This bit must be set to allow writes to the Backoff_Test and PauseTimer registers.
		0	
		1	
1	LOOPBACK		Loop back mode. Loopback mode forces internal fullduplex mode regardless of whether the fullduplex bit is set or not. The loopback bit should be changed only when GMII_en is de-asserted.
		0	Not looped back.
		1	Loop back mode enabled.

Table 38. GMAC0 MAC Control Register (GMAC0_MACCONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
0	FULLDUPLEX		Full duplex mode. Gigabit mode forces fullduplex mode regardless of whether the fullduplex bit is set or not. The FULLDUPLEX_OUT output is the value of this register bit.
		0	Half duplex mode.
		1	Full duplex mode.

5.31 *GMAC0 MAC Status Register (GMAC0_MACSTATUS)*

The MAC0 MAC status register (GMAC0_MACSTATUS) is shown in Figure 47 and described in Table 39.

Figure 47. GMAC0 MAC Status Register (GMAC0_MACSTATUS)

31	30-16
IDLE	Reserved
R-0x01	R-0x0

15-5	4	3	2	1	0
Reserved	EXT_GIG	EXT_FULLDUPLEX	Reserved	RX_FLOW_ACT	TX_FLOW_ACT
R-0x0	R-0x00	R-0x00	R-0x0	R-0x00	R-0x00

LEGEND: R = Read only; -*n* = value after reset

Table 39. GMAC0 MAC Status Register (GMAC0_MACSTATUS) Field Descriptions

Bit	Field	Value	Description
31	IDLE		GMAC idle. The GMAC is in the idle state (valid after an idle command).
		1	The GMAC is in the idle state.
		0	The GMAC is not in the idle state.
30-5	Reserved	0	Reserved.
4	EXT_GIG	0-1	External GIG. This is the value of the EXT_GIG input bit.
3	EXT_FULLDUPLEX	0-1	External fullduplex. This is the value of the EXT_FULLDUPLEX input bit.
2	Reserved	0	Reserved.
1	RX_FLOW_ACT	0-1	Receive flow control active. When asserted, indicates that receive flow control is enabled and triggered.
0	TX_FLOW_ACT	0-1	Transmit flow control active. When asserted, this bit indicates that the pause time period is being observed for a received pause frame. No new transmissions will begin while this bit is asserted except for the transmission of pause frames. Any transmission in progress when this bit is asserted will complete.

5.32 GMAC0 Soft Reset Register (GMAC0_SOFT_RESET)

The GMAC0 soft reset register (GMAC0_SOFT_RESET) is shown in Figure 48 and described in Table 40.

Figure 48. GMAC0 Soft Reset Register (GMAC0_SOFT_RESET)

31	16
Reserved	
R-0x0	

15	1	0
Reserved		SOFT_RESET
R-0x0		R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 40. GMAC0 Soft Reset Register (GMAC0_SOFT_RESET) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	SOFT_RESET		Software reset.
		0	If a zero is read then reset has occurred.
		1	If a one is read, the reset has not yet occurred.
			Writing a one to this bit causes the GMAC logic to be reset. After writing a one to this bit, it may be polled to determine if the reset has occurred.

5.33 GMAC0 RX Maximum Length Register (GMAC0_RX_MAXLEN)

The GMAC0 RX maximum length register (GMAC0_RX_MAXLEN) is shown in Figure 49 and described in Table 41.

Figure 49. GMAC0 RX Maximum Length Register (GMAC0_RX_MAXLEN)

31	16
Reserved	
R-0x0	

15	11	10	1	0
Reserved		RX_MAXLEN		
R-0x0		R/W-0x5EE		

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 41. GMAC0 RX Maximum Length Register (GMAC0_RX_MAXLEN) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved.
10-0	RX_MAXLEN	0-7FFh	Receive maximum frame length This field determines the maximum length of a received frame. The reset value is 1518 (dec). Frames with byte counts greater than RX_MAXLEN are long frames. Long frames with no errors are oversized frames. Long frames with CRC, code, or alignment error are jabber frames. The maximum value is 2047.

5.34 GMAC0 Backoff Test Register (GMAC0_BOFFTEST)

The GMAC0 backoff test register (GMAC0_BOFFTEST) is shown in Figure 50 and described in Table 42.

Figure 50. GMAC0 Backoff Test Register (GMAC0_BOFFTEST)

31	30 26	25 16
Reserved	PACEVAL	RNDNUM
R-0x0	R/W-0x00	R/W-0x00

15 12	11 10	9 0
COLL_COUNT	Reserved	TX_BACKOFF
R-0x00	R-0x0	R-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 42. GMAC0 Backoff Test Register (GMAC0_BOFFTEST) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-26	PACEVAL	0-1Fh	Pacing register current value. A non-zero value in this field indicates that transmit pacing is active. A transmit frame collision or deferral causes paceval to loaded with decimal 31, good frame transmissions (with no collisions or deferrals) cause PACEVAL to be decremented down to zero. When PACEVAL is nonzero, the transmitter delays 4 IPGs between new frame transmissions after each successfully transmitted frame that had no deferrals or collisions. Transmit pacing helps reduce "capture" effects improving overall network bandwidth.
25-16	RNDNUM	0-3FFh	Backoff random number generator. This field allows the backoff random number generator to be read (or written in test mode only). This field can be written only when MTEST has previously been set. Reading this field returns the generator's current value. The value is reset to zero and begins counting on the clock after the de-assertion of reset.
15-12	COLL_COUNT	0-Fh	Collision count. The number of collisions the current frame has experienced.
11-10	Reserved	0	Reserved.
9-0	TX_BACKOFF	0-3FFh	Backoff count. This field allows the current value of the backoff counter to be observed for test purposes. This field is loaded automatically according to the backoff algorithm, and is decremented by one for each slot time after the collision.

5.35 GMAC0 Emulation Control Register (GMAC0_EMCONTROL)

The GMAC0 emulation control register (GMAC0_EMCONTROL) is shown in Figure 51 and described in Table 43.

Figure 51. GMAC0 Emulation Control Register (GMAC0_EMCONTROL)

31 16
Reserved
R-0x0

15 2	1	0
Reserved	SOFT	FREE
R-0x0	R/W-0x00	R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 43. GMAC0 Emulation Control Register (GMAC0_EMCONTROL) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	SOFT	0-1	Emulation soft bit.
0	FREE	0-1	Emulation free bit.

5.36 GMAC0 Rx Packet Priority to Header Priority Mapping Register (GMAC0_RX_PRI_MAP)

The GMAC0 Rx packet priority to header priority mapping register (GMAC0_RX_PRI_MAP) is shown in Figure 52 and described in Table 44.

Figure 52. GMAC0 Rx Packet Priority to Header Priority Mapping Register (GMAC0_RX_PRI_MAP)

31	30 28	27	26 24	23	22 20	19	18 16
Reserved	PRI7	Reserved	PRI6	Reserved	PRI5	Reserved	PRI4
R-0x0	R/W-0x7	R-0x0	R/W-0x6	R-0x0	R/W-0x5	R-0x0	R/W-0x4

15	14 12	11	10 8	7	6 4	3	2 0
Reserved	PRI3	Reserved	PRI2	Reserved	PRI1	Reserved	PRI0
R-0x0	R/W-0x3	R-0x0	R/W-0x2	R-0x0	R/W-0x1	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 44. GMAC0 Rx Packet Priority to Header Priority Mapping Register (GMAC0_RX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-28	PRI7	0-7h	Priority 7. A packet priority of 7h is mapped (changed) to this value.
27	Reserved	0	Reserved.
26-24	PRI6	0-7h	Priority 6. A packet priority of 6h is mapped (changed) to this value.
23	Reserved	0	Reserved.
22-20	PRI5	0-7h	Priority 5. A packet priority of 5h is mapped (changed) to this value.
19	Reserved	0	Reserved.
18-16	PRI4	0-7h	Priority 4. A packet priority of 4h is mapped (changed) to this value.
15	Reserved	0	Reserved.
14-12	PRI3	0-7h	Priority 3. A packet priority of 3h is mapped (changed) to this value.
11	Reserved	0	Reserved.
10-8	PRI2	0-7h	Priority 2. A packet priority of 2h is mapped (changed) to this value.
7	Reserved	0	Reserved.
6-4	PRI1	0-7h	Priority 1. A packet priority of 1h is mapped (changed) to this value.
3	Reserved	0	Reserved.
2-0	PRI0	0-7h	Priority 0. A packet priority of 0 is mapped (changed) to this value.

5.37 GMAC1 ID/Version Register (DM648 only) (GMAC1_IDVER)

The GMAC1 ID/Version register (GMAC1_IDVER) is shown in Figure 53 and described in Table 45.

Figure 53. GMAC1 ID/Version Register (DM648 only) (GMAC1_IDVER)

31	16
RX_IDENT	
R-0x17	

15	11	10	8	7	0
RX_Z		RX_X		RX_Y	
R-0x00		R-0x1		R-0x0A	

LEGEND: R = Read only; -*n* = value after reset

Table 45. GMAC1 ID/Version Register (DM648 only) (GMAC1_IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	RX_IDENT	0-FFFFh	Receive identification value.
15-11	RX_Z	0-1Fh	Receive Z value.
10-8	RX_X	0-7h	Receive X value.
7-0	RX_Y	0-FFh	Receive Y value.

5.38 GMAC1 MAC Control Register (DM648 only) (GMAC1_MACCONTROL)

The GMAC1 MAC control register (GMAC1_MACCONTROL) is shown in Figure 54 and described in Table 46.

Figure 54. GMAC1 MAC Control Register (DM648 only) (GMAC1_MACCONTROL)

31	25	24
Reserved		RX_CMF_EN
R-0x0		R/W-0x00

23	22	21	19	18	17	16
RX_CSF_EN	RX_CEF_EN	Reserved		CTL_EN	GIG_FORCE	IFCTL_B
R/W-0x00	R/W-0x00	R-0x0		R/W-0x00	R/W-0x00	R/W-0x00

15	14	12	11	10	9	8
IFCTL_A	Reserved		CMD_IDLE	TX_SHORT_GAP_EN	Reserved	
R/W-0x00	R-0x0		R/W-0x00	R/W-0x00	R-0x0	

7	6	5	4	3	2	1	0
GIG	TX_PACE	GMII_EN	TX_FLOW_EN	RX_FLOW_EN	MTEST	LOOPBACK	FULLDUPLEX
R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00	R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 46. GMAC1 MAC Control Register (DM648 only) (GMAC1_MACCONTROL) Field Descriptions

Bit	Field	Value	Description
31-25	Reserved	0	Reserved.
24	RX_CMF_EN		RX copy MAC control frames enable. Enables MAC control frames to be transferred to memory. MAC control frames are normally acted upon (if enabled), but not copied to memory. MAC control frames that are pause frames will be acted upon if enabled in the GMAC*n*_MACCONTROL register, regardless of the value of RX_CMF_EN. Frames transferred to memory due to RX_CMF_EN will have the control bit set in their EOP buffer descriptor.
		0	MAC control frames are filtered (but acted upon if enabled).
		1	MAC control frames are transferred to memory.
23	RX_CSF_EN		RX copy short frames enable. Enables frames or fragments shorter than 64 bytes to be copied to memory. Frames transferred to memory due to RX_CSF_EN will have the fragment or undersized bit set in their EOP buffer descriptor. Fragments are short frames that contain CRC/align/code errors and undersized are short frames without errors.
		0	Short frames are filtered.
		1	Short frames are transferred to memory.
22	RX_CEF_EN		RX copy error frames enable. Enables frames containing errors to be transferred to memory. The appropriate error bit will be set in the frame EOP buffer descriptor. Frames containing errors will be filtered when RX_CEF _EN is not set.
		0	Frames containing errors are filtered.
		1	Frames containing errors are transferred to memory.
21-19	Reserved	0	Reserved.
18	CTL_EN		Control enable. Enables the fullduplex and gigabit mode to be selected from the FULLDUPLEX_IN and GIG_IN input signals and not from the fullduplex and gig bits in this register. The FULLDUPLEX_MODE bit reflects the actual fullduplex mode selected.
		0	
		1	

Table 46. GMAC1 MAC Control Register (DM648 only) (GMAC1_MACCONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
17	GIG_FORCE		Gigabit mode force. This bit is used to force the GMAC into gigabit mode.
		0	
		1	
16	IFCTL_B		Interface control B. Intended as a general purpose output bit to be used to control external gaskets associated with the GMII (GMII to RGMII etc).
		0	
		1	
15	IFCTL_A		Interface control A. Intended as a general purpose output bit to be used to control external gaskets associated with the GMII (GMII to RGMII etc).
		0	
		1	
14-12	Reserved	0	Reserved.
11	CMD_IDLE		Command idle.
		0	Idle not commanded.
		1	Idle commanded (read idle in MACStatus).
10	TX_SHORT_GAP_EN		Transmit short gap enable
		0	Transmit with a short IPG is disabled.
		1	Transmit with a short IPG (when TX_SHORT_GAP input is asserted) is enabled.
9-8	Reserved	0	Reserved.
7	GIG		Gigabit mode.
		0	10/100 mode.
		1	Gigabit mode (full duplex only).
6	TX_PACE		Transmit pacing
		0	Transmit pacing disabled.
		1	Transmit pacing enabled.
5	GMII_EN		GMII enable.
		0	GMII RX and TX held in reset.
		1	GMII RX and TX released from reset.
4	TX_FLOW_EN		Transmit flow control enable. Determines if incoming pause frames are acted upon in full-duplex mode. Incoming pause frames are not acted upon in half-duplex mode regardless of this bit setting. The RX_MBP_Enable bits determine whether or not received pause frames are transferred to memory.
		0	Transmit flow control disabled. Full-duplex mode. Incoming pause frames are not acted upon.
		1	Transmit flow control enabled. Full-duplex mode. Incoming pause frames are acted upon.
3	RX_FLOW_EN		Receive flow control enable.
		0	Receive flow control disabled Half-duplex mode. No flow control generated collisions are sent. Full-duplex mode. No outgoing pause frames are sent.
		1	Receive flow control enabled Half-duplex mode. Collisions are initiated when receive flow control is triggered. Full-duplex mode. Outgoing pause frames are sent when receive flow control is triggered.
2	MTEST		Manufacturing test mode. This bit must be set to allow writes to the Backoff_Test and PauseTimer registers.
		0	
		1	
1	LOOPBACK		Loop back mode. Loopback mode forces internal fullduplex mode regardless of whether the fullduplex bit is set or not. The loopback bit should be changed only when GMII_en is de-asserted.
		0	Not looped back.
		1	Loop back mode enabled.

Table 46. GMAC1 MAC Control Register (DM648 only) (GMAC1_MACCONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
0	FULLDUPLEX		Full duplex mode. Gigabit mode forces fullduplex mode regardless of whether the fullduplex bit is set or not. The FULLDUPLEX_OUT output is the value of this register bit.
		0	Half duplex mode.
		1	Full duplex mode.

5.39 GMAC1 MAC Status Register (DM648 only) (GMAC1_MACSTATUS)

The GMAC1 MAC status register (GMAC1_MACSTATUS) is shown in Figure 55 and described in Table 47.

Figure 55. GMAC1 MAC Status Register (DM648 only) (GMAC1_MACSTATUS)

31	16
IDLE	Reserved
R-0x01	R-0x0

15 5	4	3	2	1	0
Reserved	EXT_GIG	EXT_FULLDUPLEX	Reserved	RX_FLOW_ACT	TX_FLOW_ACT
R-0x0	R-0x00	R-0x00	R-0x0	R-0x00	R-0x00

LEGEND: R = Read only; -*n* = value after reset

Table 47. GMAC1 MAC Status Register (DM648 only) (GMAC1_MACSTATUS) Field Descriptions

Bit	Field	Value	Description
31	IDLE		GMAC idle. The GMAC is in the idle state (valid after an idle command).
		1	The GMAC is in the idle state.
		0	The GMAC is not in the idle state.
30-5	Reserved	0	Reserved.
4	EXT_GIG	0-1	External GIG. This is the value of the EXT_GIG input bit.
3	EXT_FULLDUPLEX	0-1	External fullduplex. This is the value of the EXT_FULLDUPLEX input bit.
2	Reserved	0	Reserved.
1	RX_FLOW_ACT	0-1	Receive flow control active. When asserted, indicates that receive flow control is enabled and triggered.
0	TX_FLOW_ACT	0-1	Transmit Flow control active. When asserted, this bit indicates that the pause time period is being observed for a received pause frame. No new transmissions will begin while this bit is asserted except for the transmission of pause frames. Any transmission in progress when this bit is asserted will complete.

5.40 GMAC1 Soft Reset Register (DM648 only) (GMAC1_SOFT_RESET)

The GMAC1 soft reset register (GMAC1_SOFT_RESET) is shown in Figure 56 and described in Table 48.

Figure 56. GMAC1 Soft Reset Register (DM648 only) (GMAC1_SOFT_RESET)

31 16
Reserved
R-0x0

15 1	0
Reserved	SOFT_RESET
R-0x0	R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 48. GMAC1 Soft Reset Register (DM648 only) (GMAC1_SOFT_RESET) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	SOFT_RESET		Software reset.
		0	If a zero is read then reset has occurred.
		1	If a one is read, the reset has not yet occurred.
			Writing a one to this bit causes the GMAC logic to be reset. After writing a one to this bit, it may be polled to determine if the reset has occurred.

5.41 GMAC1 RX Maximum Length Register (DM648 only) (GMAC1_RX_MAXLEN)

The GMAC1 RX maximum length register (GMAC1_RX_MAXLEN) is shown in Figure 57 and described in Table 49.

Figure 57. GMAC1 RX Maximum Length Register (DM648 only) (GMAC1_RX_MAXLEN)

31 16
Reserved
R-0x0

15 11	10 0
Reserved	RX_MAXLEN
R-0x0	R/W-0x5EE

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 49. GMAC1 RX Maximum Length Register (DM648 only) (GMAC1_RX_MAXLEN) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved.
10-0	RX_MAXLEN	0-7FFh	Receive maximum frame length This field determines the maximum length of a received frame. The reset value is 1518 (dec). Frames with byte counts greater than RX_MAXLEN are long frames. Long frames with no errors are oversized frames. Long frames with CRC, code, or alignment error are jabber frames. The maximum value is 2047.

5.42 GMAC1 Backoff Test Register (DM648 only) (GMAC1_BOFFTEST)

The GMAC1 backoff test register (GMAC1_BOFFTEST) is shown in Figure 58 and described in Table 50.

Figure 58. GMAC1 Backoff Test Register (DM648 only) (GMAC1_BOFFTEST)

31	30 26	25 16
Reserved	PACEVAL	RNDNUM
R-0x0	R/W-0x00	R/W-0x00

15 12	11 10	9 0
COLL_COUNT	Reserved	TX_BACKOFF
R-0x00	R-0x0	R-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 50. GMAC1 Backoff Test Register (DM648 only) (GMAC1_BOFFTEST) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-26	PACEVAL	0-1Fh	Pacing register current value. A non-zero value in this field indicates that transmit pacing is active. A transmit frame collision or deferral causes paceval to loaded with decimal 31, good frame transmissions (with no collisions or deferrals) cause paceval to be decremented down to zero. When paceval is nonzero, the transmitter delays 4 IPGs between new frame transmissions after each successfully transmitted frame that had no deferrals or collisions. Transmit pacing helps reduce "capture" effects improving overall network bandwidth.
25-16	RNDNUM	0-3FFh	Backoff random number generator. This field allows the backoff random number generator to be read (or written in test mode only). This field can be written only when mtest has previously been set. Reading this field returns the generator's current value. The value is reset to zero and begins counting on the clock after the de-assertion of reset.
15-12	COLL_COUNT	0-Fh	Collision count. The number of collisions the current frame has experienced.
11-10	Reserved	0	Reserved.
9-0	TX_BACKOFF	0-3FFh	Backoff count. This field allows the current value of the backoff counter to be observed for test purposes. This field is loaded automatically according to the backoff algorithm, and is decremented by one for each slot time after the collision.

5.43 GMAC1 Emulation Control Register (DM648 only) (GMAC1_EMCONTROL)

The GMAC1 emulation control register (GMAC1_EMCONTROL) is shown in Figure 59 and described in Table 51.

Figure 59. GMAC1 Emulation Control Register (DM648 only) (GMAC1_EMCONTROL)

31-16
Reserved
R-0x0

15-2	1	0
Reserved	SOFT	FREE
R-0x0	R/W-0x00	R/W-0x00

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 51. GMAC1 Emulation Control Register (DM648 only) (GMAC1_EMCONTROL) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	SOFT	0-1	Emulation soft bit.
0	FREE	0-1	Emulation free bit.

5.44 GMAC1 Rx Packet Priority to Header Priority Mapping Register (DM648 only) (GMAC1_RX_PRI_MAP)

The GMAC1 Rx packet priority to header priority mapping register (GMAC1_RX_PRI_MAP) is shown in Figure 60 and described in Table 52.

Figure 60. GMAC1 Rx Packet Priority to Header Priority Mapping Register (DM648 only) (GMAC1_RX_PRI_MAP)

31	30 28	27	26 24	23	22 20	19	18 16
Reserved	PRI7	Reserved	PRI6	Reserved	PRI5	Reserved	PRI4
R-0x0	R/W-0x7	R-0x0	R/W-0x6	R-0x0	R/W-0x5	R-0x0	R/W-0x4

15	14 12	11	10 8	7	6 4	3	2 0
Reserved	PRI3	Reserved	PRI2	Reserved	PRI1	Reserved	PRI0
R-0x0	R/W-0x3	R-0x0	R/W-0x2	R-0x0	R/W-0x1	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 52. GMAC1 Rx Packet Priority to Header Priority Mapping Register (DM648 only) (GMAC1_RX_PRI_MAP) Field Descriptions

Bit	Field	Value	Description
31	Reserved	0	Reserved.
30-28	PRI7	0-7h	Priority 7. A packet priority of 7h is mapped (changed) to this value.
27	Reserved	0	Reserved.
26-24	PRI6	0-7h	Priority 6. A packet priority of 6h is mapped (changed) to this value.
23	Reserved	0	Reserved.
22-20	PRI5	0-7h	Priority 5. A packet priority of 5h is mapped (changed) to this value.
19	Reserved	0	Reserved.
18-16	PRI4	0-7h	Priority 4. A packet priority of 4h is mapped (changed) to this value.
15	Reserved	0	Reserved.
14-12	PRI3	0-7h	Priority 3. A packet priority of 3h is mapped (changed) to this value.
11	Reserved	0	Reserved.
10-8	PRI2	0-7h	Priority 2. A packet priority of 2h is mapped (changed) to this value.
7	Reserved	0	Reserved.
6-4	PRI1	0-7h	Priority 1. A packet priority of 1h is mapped (changed) to this value.
3	Reserved	0	Reserved.
2-0	PRI0	0-7h	Priority 0. A packet priority of 0 is mapped (changed) to this value.

5.45 CPDMA_REGS TX Identification and Version Register (TX_IDVER)

The CPDMA_REGS TX identification and version register (TX_IDVER) is shown in Figure 61 and described in Table 53.

Figure 61. CPDMA_REGS TX Identification and Version Register (TX_IDVER)

31 16
TX_IDENT
R-0x18

15 8	7 0
TX_MAJOR_VER	TX_MINOR_VER
R-0x1	R-0x06

LEGEND: R = Read only; -*n* = value after reset

Table 53. CPDMA_REGS TX Identification and Version Register (TX_IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	TX_IDENT	0-FFFFh	TX Identification value.
15-8	TX_MAJOR_VER	0-FFh	TX major version value. The value read is the major version number.
7-0	TX_MINOR_VER	0-FFh	TX minor version value. The value read is the minor version number.

5.46 CPDMA_REGS TX Control Register (TX_CONTROL)

The CPDMA_REGS TX control register (TX_CONTROL) is shown in Figure 62 and described in Table 54.

Figure 62. CPDMA_REGS TX Control Register (TX_CONTROL)

31 16
Reserved
R-0x0

15 1	0
Reserved	TX_EN
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 54. CPDMA_REGS TX Control Register (TX_CONTROL) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	TX_EN		Transmit enable.
		0	Transmitter disabled.
		1	Transmitter enabled.

5.47 CPDMA_REGS TX Teardown Register (TX_TEARDOWN)

The CPDMA_REGS TX teardown register (TX_TEARDOWN) is shown in Figure 63 and described in Table 55.

Figure 63. CPDMA_REGS TX Teardown Register (TX_TEARDOWN)

31	30 16
TX_TDN_READY	Reserved
R-0x0	R-0x0

15 3	2 0
Reserved	TX_TDN_CH
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 55. CPDMA_REGS TX Teardown Register (TX_TEARDOWN) Field Descriptions

Bit	Field	Value	Description
31	TX_TDN_READY		TX teardown ready. Read as zero, but is always assumed to be one.
30-3	Reserved	0	Reserved. Read as zero.
2-0	TX_TDN_CH	0-7h	TX teardown channel. Transmit channel teardown is commanded by writing the encoded value of the transmit channel to be torn down. The teardown register is read as zero. 000 – teardown transmit channel 0 … 111 – teardown transmit channel 7.

5.48 CPDMA_REGS RX Identification and Version Register (RX_IDVER)

The CPDMA_REGS RX identification and version register (RX_IDVER) is shown in Figure 64 and described in Table 56.

Figure 64. CPDMA_REGS RX Identification and Version Register (RX_IDVER)

31 16
RX_IDENT
R-0x18

15 8	7 0
RX_MAJOR_VER	RX_MINOR_VER
R-0x1	R-0x06

LEGEND: R = Read only; -*n* = value after reset

Table 56. CPDMA_REGS RX Identification and Version Register (RX_IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	RX_IDENT	0-FFFFh	RX identification value.
15-8	RX_MAJOR_VER	0-FFh	RX major version value. The value read is the major version number.
7-0	RX_MINOR_VER	0-FFh	RX minor version value. The value read is the minor version number.

5.49 CPDMA_REGS RX Control Register (RX_CONTROL)

The CPDMA_REGS RX control register (RX_CONTROL) is shown in Figure 65 and described in Table 57.

Figure 65. CPDMA_REGS RX Control Register (RX_CONTROL)

31–16
Reserved
R-0x0

15–1	0
Reserved	RX_EN
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 57. CPDMA_REGS RX Control Register (RX_CONTROL) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	RX_EN		Receive enable.
		0	Receiver disabled.
		1	Receiver enabled.

5.50 CPDMA_REGS RX Teardown Register (RX_TEARDOWN)

The CPDMA_REGS RX teardown register (RX_TEARDOWN) is shown in Figure 66 and described in Table 58.

Figure 66. CPDMA_REGS RX Teardown Register (RX_TEARDOWN)

31	30–16
RX_TDN_READY	Reserved
R-0x0	R-0x0

15–3	2–0
Reserved	RX_TDN_CH
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 58. CPDMA_REGS RX Teardown Register (RX_TEARDOWN) Field Descriptions

Bit	Field	Value	Description
31	RX_TDN_READY	0-1	Rx teardown ready. Read as zero, but is always assumed to be one.
30-3	Reserved	0	Reserved. Read as zero.
2-0	RX_TDN_CH	0-7h	Rx teardown channel. Receive channel teardown is commanded by writing the encoded value of the receive channel to be torn down. The teardown register is read as zero. 000 – teardown receive channel 0 … 111 – teardown receive channel 7.

5.51 CPDMA_REGS Soft Reset Register (SOFT_RESET)

The CPDMA_REGS soft reset register (SOFT_RESET) is shown in Figure 67 and described in Table 59.

Figure 67. CPDMA_REGS Soft Reset Register (SOFT_RESET)

31	16
Reserved	
R-0x0	

15	1	0
Reserved		SOFT_RESET
R-0x0		R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 59. CPDMA_REGS Soft Reset Register (Soft_Reset) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	SOFT_RESET		Software reset.
		0	If a zero is read then reset has occurred.
		1	If a one is read, the reset has not yet occurred.
			Writing a one to this bit causes the CPDMA logic to be reset. Software reset occurs when the RX and TX DMA Controllers are in an idle state. After writing a one to this bit, it may be polled to determine if the reset has occurred.

5.52 CPDMA_REGS CPDMA Control Register (DMACONTROL)

The CPDMA_REGS CPDMA control register (DMACONTROL) is shown in Figure 68 and described in Table 60.

Figure 68. CPDMA_REGS CPDMA Control Register (DMACONTROL)

31	16
Reserved	
R-0x0	

15	5	4	3	2	1	0
Reserved		RX_CEF	CMD_IDLE	RX_OFFLEN_BLOCK	RX_OWNERSHIP	TX_PTYPE
R-0x0		R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 60. CPDMA_REGS CPDMA Control Register (DMACONTROL) Field Descriptions

Bit	Field	Value	Description
31-5	Reserved	0	Reserved.
4	RX_CEF		RX copy error frames enable. Enables DMA overrun frames to be transferred to memory (up to the point of overrun). The overrun error bit will be set in the frame EOP buffer descriptor. Overrun frame data will be filtered when RX_CEF is not set. Frames coming from the receive FIFO with other error bits set are not effected by this bit.
		0	Frames containing overrun errors are filtered.
		1	Frames containing overrun errors are transferred to memory.
3	CMD_IDLE		Command idle.
		0	Idle commanded (read idle in DMAStatus).
		1	Idle not commanded.
2	RX_OFFLEN_BLOCK		Receive Offset/Length word write block.
		0	Do not block the DMA writes to the receive buffer descriptor offset/buffer length word. The offset/buffer length word is written as specified in CPPI 3.0.
		1	Block all CPDMA DMA controller writes to the receive buffer descriptor offset/buffer length words during CPPI packet processing. When this bit is set, the CPDMA will never write the third word to any receive buffer descriptor.
1	RX_OWNERSHIP		Receive ownership write bit value.
		0	The CPDMA writes the RX_OWNERSHIP bit to zero at the end of packet processing as specified in CPPI 3.0.
		1	The CPDMA writes the RX_OWNERSHIP bit to one at the end of packet processing. Users who do not use the ownership mechanism can use this mode to preclude the necessity of software having to set this bit each time the buffer descriptor is used.
0	TX_PTYPE		Transmit queue priority type.
		0	The queue uses a round robin scheme to select the next channel for transmission.
		1	The queue uses a fixed (channel 7 highest priority) priority scheme to select the next channel for transmission.

5.53 CPDMA_REGS CPDMA Status Register (DMASTATUS)

The CPDMA_REGS CPDMA status register (DMASTATUS) is shown in Figure 69 and described in Table 61.

Figure 69. CPDMA_REGS CPDMA Status Register (DMASTATUS)

31	30 24	23 20	19	18 16
IDLE	Reserved	TX_HOST_ERR_CODE	Reserved	TX_ERR_CH
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

15 12	11	10 8	7 0
RX_HOST_ERR_CODE	Reserved	RX_ERROR_CH	Reserved
R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 61. CPDMA_REGS CPDMA Status Register (DMASTATUS) Field Descriptions

Bit	Field	Value	Description
31	IDLE	0-1	Idle Status Bit. Indicates when set that the CPDMA is not transferring a packet on transmit or receive.
30-24	Reserved	0	Reserved.
23-20	TX_HOST_ERR_CODE		TX host error code. This field is set to indicate CPDMA detected TX DMA related host errors. The host should read this field after a HOST_ERR_INT to determine the error. Host error Interrupts require hardware reset in order to recover. A zero packet length is an error, but it is not detected.
		0	No error.
		0001	SOP error.
		0010	OWNERSHIP bit not set in SOP buffer.
		0011	Zero next buffer descriptor pointer without EOP.
		0100	Zero buffer pointer.
		0101	Zero buffer length.
		0110	Packet length error (sum of buffers < packet length).
		0111	Reserved.
		1111	Reserved.
19	Reserved	0	Reserved.
18-16	TX_ERR_CH	0-7h	TX host error channel. This field indicates which TX channel (if applicable) the host error occurred on. This field is cleared to zero on a host read. 000 – The host error occurred on TX channel 0 … 111 – The host error occurred on TX channel 7
15-12	RX_HOST_ERR_CODE		RX host error code. This field is set to indicate CPDMA detected RX DMA related host errors. The host should read this field after a HOST_ERR_INT to determine the error. Host error Interrupts require hardware reset in order to recover.
		0000	No error.
		0001	Reserved.
		0010	OWNERSHIP bit not set in input buffer.
		0011	Reserved.
		0100	Zero Buffer Pointer.
		0101	Zero buffer length on non-SOP descriptor.
		0110	SOP buffer length not greater than offset.
		0111	
		1111	Reserved.
11	Reserved	0	Reserved.

Table 61. CPDMA_REGS CPDMA Status Register (DMASTATUS) Field Descriptions (continued)

Bit	Field	Value	Description
10-8	RX_ERROR_CH	0-7h	RX host error channel. This field indicates which RX channel the host error occurred on. This field is cleared to zero on a host read. 000 – The host error occurred on RX channel 0 … 111 – The host error occurred on RX channel 7
7-0	Reserved	0	Reserved.

5.54 CPDMA_REGS Receive Buffer Offset Register (RX_BUFFER_OFFSET)

The CPDMA_REGS receive buffer offset register (RX_BUFFER_OFFSET) is shown in Figure 70 and described in Table 62.

Figure 70. CPDMA_REGS Receive Buffer Offset Register (RX_BUFFER_OFFSET)

31	16
Reserved	
R-0x0	

15	0
RX_BUFFER_OFFSET	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 62. CPDMA_REGS Receive Buffer Offset Register (RX_BUFFER_OFFSET) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15-0	RX_BUFFER_OFFSET	0-FFFFh	Receive buffer offset value. The RX_BUFFER_OFFSET will be written by the port into each frame SOP buffer descriptor BUFFER_OFFSET field. The frame data will begin after the RX_BUFFER_OFFSET value of bytes. A value of 0 indicates that there are no unused bytes at the beginning of the data and that valid data begins on the first byte of the buffer. A value of 000Fh (decimal 15) indicates that the first 15 bytes of the buffer are to be ignored by the port and that valid buffer data starts on byte 16 of the buffer. This value is used for all channels.

5.55 CPDMA_REGS Emulation Control Register (EMCONTROL)

The CPDMA_REGS emulation control register (EMCONTROL) is shown in Figure 71 and described in Table 63.

Figure 71. CPDMA_REGS Emulation Control Register (EMCONTROL)

31–16
Reserved
R-0x0

15–2	1	0
Reserved	SOFT	FREE
R-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 63. CPDMA_REGS Emulation Control Register (EMCONTROL) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	SOFT	0-1	Emulation Soft Bit.
0	FREE	0-1	Emulation free Bit.

5.56 CPDMA_INT TX Interrupt Status Register (Raw Value) (TX_INTSTAT_RAW)

The CPDMA_INT TX interrupt status register (Raw Value) (TX_INTSTAT_RAW) is shown in Figure 72 and described in Table 64.

Figure 72. CPDMA_INT TX Interrupt Status Register (Raw Value) (TX_INTSTAT_RAW)

31-16
Reserved
R-0x0

15	14	13	12	11	10	9	8
TX7_THRESH_PEND	TX6_THRESH_PEND	TX5_THRESH_PEND	TX4_THRESH_PEND	TX3_THRESH_PEND	TX2_THRESH_PEND	TX1_THRESH_PEND	TX0_THRESH_PEND
R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1

7	6	5	4	3	2	1	0
TX7_PEND	TX6_PEND	TX5_PEND	TX4_PEND	TX3_PEND	TX2_PEND	TX1_PEND	TX0_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 64. CPDMA_INT TX Interrupt Status Register (Raw Value) (TX_INTSTAT_RAW) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	TX7_THRESH_PEND	0-1	TX7_THRESH_PEND raw interrupt read (before mask).
14	TX6_THRESH_PEND	0-1	TX6_THRESH_PEND raw interrupt read (before mask).
13	TX5_THRESH_PEND	0-1	TX5_THRESH_PEND raw interrupt read (before mask).
12	TX4_THRESH_PEND	0-1	TX4_THRESH_PEND raw interrupt read (before mask).
11	TX3_THRESH_PEND	0-1	TX3_THRESH_PEND raw interrupt read (before mask).
10	TX2_THRESH_PEND	0-1	TX2_THRESH_PEND raw interrupt read (before mask).
9	TX1_THRESH_PEND	0-1	TX1_THRESH_PEND raw interrupt read (before mask).
8	TX0_THRESH_PEND	0-1	TX0_THRESH_PEND raw interrupt read (before mask).
7	TX7_PEND	0-1	TX7_PEND raw interrupt read (before mask).
6	TX6_PEND	0-1	TX6_PEND raw interrupt read (before mask).
5	TX5_PEND	0-1	TX5_PEND raw interrupt read (before mask).
4	TX4_PEND	0-1	TX4_PEND raw interrupt read (before mask).
3	TX3_PEND	0-1	TX3_PEND raw interrupt read (before mask).
2	TX2_PEND	0-1	TX2_PEND raw interrupt read (before mask).
1	TX1_PEND	0-1	TX1_PEND raw interrupt read (before mask).
0	TX0_PEND	0-1	TX0_PEND raw interrupt read (before mask).

5.57 CPDMA_INT TX Interrupt Status Register (Masked Value) (TX_INTSTAT_MASKED)

The TX interrupt status register (Masked Value) (TX_INTSTAT_MASKED) is shown in Figure 73 and described in Table 65.

Figure 73. CPDMA_INT TX Interrupt Status Register (Masked Value) (TX_INTSTAT_MASKED)

31-16
Reserved
R-0x0

15	14	13	12	11	10	9	8
TX7_THRESH_PEND	TX6_THRESH_PEND	TX5_THRESH_PEND	TX4_THRESH_PEND	TX3_THRESH_PEND	TX2_THRESH_PEND	TX1_THRESH_PEND	TX0_THRESH_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

7	6	5	4	3	2	1	0
TX7_PEND	TX6_PEND	TX5_PEND	TX4_PEND	TX3_PEND	TX2_PEND	TX1_PEND	TX0_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 65. CPDMA_INT TX Interrupt Status Register (Masked Value) (TX_INTSTAT_MASKED) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	TX7_THRESH_PEND	0-1	TX7_THRESH_PEND masked interrupt read.
14	TX6_THRESH_PEND	0-1	TX6_THRESH_PEND masked interrupt read.
13	TX5_THRESH_PEND	0-1	TX5_THRESH_PEND masked interrupt read.
12	TX4_THRESH_PEND	0-1	TX4_THRESH_PEND masked interrupt read.
11	TX3_THRESH_PEND	0-1	TX3_THRESH_PEND masked interrupt read.
10	TX2_THRESH_PEND	0-1	TX2_THRESH_PEND masked interrupt read.
9	TX1_THRESH_PEND	0-1	TX1_THRESH_PEND masked interrupt read.
8	TX0_THRESH_PEND	0-1	TX0_THRESH_PEND masked interrupt read.
7	TX7_PEND	0-1	TX7_PEND masked interrupt read.
6	TX6_PEND	0-1	TX6_PEND masked interrupt read.
5	TX5_PEND	0-1	TX5_PEND masked interrupt read.
4	TX4_PEND	0-1	TX4_PEND masked interrupt read.
3	TX3_PEND	0-1	TX3_PEND masked interrupt read.
2	TX2_PEND	0-1	TX2_PEND masked interrupt read.
1	TX1_PEND	0-1	TX1_PEND masked interrupt read.
0	TX0_PEND	0-1	TX0_PEND masked interrupt read.

5.58 CPDMA_INT TX Interrupt Mask Set Register (TX_INTMASK_SET)

The CPDMA_INT TX interrupt mask set register (TX_INTMASK_SET) is shown in Figure 74 and described in Table 66.

Figure 74. CPDMA_INT TX Interrupt Mask Set Register (TX_INTMASK_SET)

31 ... 16
Reserved
R-0x0

15	14	13	12	11	10	9	8
TX7_THRESH_PEND_MASK	TX6_THRESH_PEND_MASK	TX5_THRESH_PEND_MASK	TX4_THRESH_PEND_MASK	TX3_THRESH_PEND_MASK	TX2_THRESH_PEND_MASK	TX1_THRESH_PEND_MASK	TX0_THRESH_PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

7	6	5	4	3	2	1	0
TX7_PEND_MASK	TX6_PEND_MASK	TX5_PEND_MASK	TX4_PEND_MASK	TX3_PEND_MASK	TX2_PEND_MASK	TX1_PEND_MASK	TX0_PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

LEGEND: WSR = Write to Set and Read; R = Read only; -*n* = value after reset

Table 66. CPDMA_INT TX Interrupt Mask Set Register (TX_INTMASK_SET) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	TX7_THRESH_PEND_MASK	0-1	TX Channel 7 threshold pending interrupt mask. Write one to enable interrupt.
14	TX6_THRESH_PEND_MASK	0-1	TX Channel 6 threshold pending interrupt mask. Write one to enable interrupt.
13	TX5_THRESH_PEND_MASK	0-1	TX Channel 5 threshold pending interrupt mask. Write one to enable interrupt.
12	TX4_THRESH_PEND_MASK	0-1	TX Channel 4 threshold pending interrupt mask. Write one to enable interrupt.
11	TX3_THRESH_PEND_MASK	0-1	TX Channel 3 threshold pending interrupt mask. Write one to enable interrupt.
10	TX2_THRESH_PEND_MASK	0-1	TX Channel 2 threshold pending interrupt mask. Write one to enable interrupt.
9	TX1_THRESH_PEND_MASK	0-1	TX Channel 1 threshold pending interrupt mask. Write one to enable interrupt.
8	TX0_THRESH_PEND_MASK	0-1	TX Channel 0 threshold pending interrupt mask. Write one to enable interrupt.
7	TX7_PEND_MASK	0-1	TX Channel 7 pending interrupt mask. Write one to enable interrupt.
6	TX6_PEND_MASK	0-1	TX Channel 6 pending interrupt mask. Write one to enable interrupt.
5	TX5_PEND_MASK	0-1	TX Channel 5 pending interrupt mask. Write one to enable interrupt.
4	TX4_PEND_MASK	0-1	TX Channel 4 pending interrupt mask. Write one to enable interrupt.
3	TX3_PEND_MASK	0-1	TX Channel 3 pending interrupt mask. Write one to enable interrupt.
2	TX2_PEND_MASK	0-1	TX Channel 2 pending interrupt mask. Write one to enable interrupt.
1	TX1_PEND_MASK	0-1	TX Channel 1 pending interrupt mask. Write one to enable interrupt.
0	TX0_PEND_MASK	0-1	TX Channel 0 pending interrupt mask. Write one to enable interrupt.

5.59 CPDMA_INT TX Interrupt Mask Clear Register (TX_INTMASK_CLEAR)

The CPDMA_INT TX interrupt mask clear register (TX_INTMASK_CLEAR) is shown in Figure 75 and described in Table 67.

Figure 75. CPDMA_INT TX Interrupt Mask Clear Register (TX_INTMASK_CLEAR)

31–16
Reserved
R-0x0

15	14	13	12	11	10	9	8
TX7_THRESH_ PEND_MASK	TX6_THRESH_ PEND_MASK	TX5_THRESH_ PEND_MASK	TX4_THRESH_ PEND_MASK	TX3_THRESH_ PEND_MASK	TX2_THRESH_ PEND_MASK	TX1_THRESH_ PEND_MASK	TX0_THRESH_ PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

7	6	5	4	3	2	1	0
TX7_PEND_MASK	TX6_PEND_MASK	TX5_PEND_MASK	TX4_PEND_MASK	TX3_PEND_MASK	TX2_PEND_MASK	TX1_PEND_MASK	TX0_PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

LEGEND: WSR = Write to Set and Read; R = Read only; -*n* = value after reset

Table 67. CPDMA_INT TX Interrupt Mask Clear Register (TX_INTMASK_CLEAR) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	TX7_THRESH_PEND_MASK	0-1	TX Channel 7 threshold pending interrupt mask. Write one to disable interrupt.
14	TX6_THRESH_PEND_MASK	0-1	TX Channel 6 threshold pending interrupt mask. Write one to disable interrupt.
13	TX5_THRESH_PEND_MASK	0-1	TX Channel 5 threshold pending interrupt mask. Write one to disable interrupt.
12	TX4_THRESH_PEND_MASK	0-1	TX Channel 4 threshold pending interrupt mask. Write one to disable interrupt.
11	TX3_THRESH_PEND_MASK	0-1	TX Channel 3 threshold pending interrupt mask. Write one to disable interrupt.
10	TX2_THRESH_PEND_MASK	0-1	TX Channel 2 threshold pending interrupt mask. Write one to disable interrupt.
9	TX1_THRESH_PEND_MASK	0-1	TX Channel 1 threshold pending interrupt mask. Write one to disable interrupt.
8	TX0_THRESH_PEND_MASK	0-1	TX Channel 0 threshold pending interrupt mask. Write one to disable interrupt.
7	TX7_PEND_MASK	0-1	TX Channel 7 pending interrupt mask. Write one to disable interrupt.
6	TX6_PEND_MASK	0-1	TX Channel 6 pending interrupt mask. Write one to disable interrupt.
5	TX5_PEND_MASK	0-1	TX Channel 5 pending interrupt mask. Write one to disable interrupt.
4	TX4_PEND_MASK	0-1	TX Channel 4 pending interrupt mask. Write one to disable interrupt.
3	TX3_PEND_MASK	0-1	TX Channel 3 pending interrupt mask. Write one to disable interrupt.
2	TX2_PEND_MASK	0-1	TX Channel 2 pending interrupt mask. Write one to disable interrupt.
1	TX1_PEND_MASK	0-1	TX Channel 1 pending interrupt mask. Write one to disable interrupt.
0	TX0_PEND_MASK	0-1	TX Channel 0 pending interrupt mask. Write one to disable interrupt.

5.60 CPDMA_INT Input Vector Register (Read Only) (CPDMA_IN_VECTOR)

The CPDMA_INT input vector register (Read Only) (CPDMA_IN_VECTOR) is shown in Figure 76 and described in Table 68.

Figure 76. CPDMA_INT Input Vector Register (Read Only) (CPDMA_IN_VECTOR)

31	0
DMA_IN_VECTOR	
R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 68. CPDMA_INT Input Vector Register (Read Only) (CPDMA_IN_VECTOR) Field Descriptions

Bit	Field	Value	Description
31-0	DMA_IN_VECTOR	0-FFFF FFFFh	DMA input vector.

5.61 CPDMA_INT End Of Interrupt Vector Register (CPDMA_EOI_VECTOR)

The CPDMA_INT end of interrupt vector register (CPDMA_EOI_VECTOR) is shown in Figure 77 and described in Table 69.

Figure 77. CPDMA_INT End Of Interrupt Vector Register (CPDMA_EOI_VECTOR)

31	16
Reserved	
R-0x0	

15	4	0
Reserved	DMA_EOI_Vector	
R-0x0	R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 69. CPDMA_INT End Of Interrupt Vector Register (CPDMA_EOI_VECTOR) Field Descriptions

Bit	Field	Value	Description
31-5	Reserved	0	Reserved
4-0	DMA_EOI_Vector	0-1Fh	DMA end of interrupt vector.

5.62 CPDMA_INT RX Interrupt Status Register (Raw Value) (RX_INTSTAT_RAW)

The CPDMA_INT RX interrupt status register (Raw Value) (RX_INTSTAT_RAW) is shown in Figure 78 and described in Table 70.

Figure 78. CPDMA_INT RX Interrupt Status Register (Raw Value) (RX_INTSTAT_RAW)

31	16
Reserved	
R-0x0	

15	14	13	12	11	10	9	8
RX7_THRESH_PEND	RX6_THRESH_PEND	RX5_THRESH_PEND	RX4_THRESH_PEND	RX3_THRESH_PEND	RX2_THRESH_PEND	RX1_THRESH_PEND	RX0_THRESH_PEND
R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1	R-0x1

7	6	5	4	3	2	1	0
RX7_PEND	RX6_PEND	RX5_PEND	RX4_PEND	RX3_PEND	RX2_PEND	RX1_PEND	RX0_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 70. CPDMA_INT RX Interrupt Status Register (Raw Value) (RX_INTSTAT_RAW) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	RX7_THRESH_PEND	0-1	RX7_THRESH_PEND raw interrupt read (before mask).
14	RX6_THRESH_PEND	0-1	RX6_THRESH_PEND raw interrupt read (before mask).
13	RX5_THRESH_PEND	0-1	RX5_THRESH_PEND raw interrupt read (before mask).
12	RX4_THRESH_PEND	0-1	RX4_THRESH_PEND raw interrupt read (before mask).
11	RX3_THRESH_PEND	0-1	RX3_THRESH_PEND raw interrupt read (before mask).
10	RX2_THRESH_PEND	0-1	RX2_THRESH_PEND raw interrupt read (before mask).
9	RX1_THRESH_PEND	0-1	RX1_THRESH_PEND raw interrupt read (before mask).
8	RX0_THRESH_PEND	0-1	RX0_THRESH_PEND raw interrupt read (before mask).
7	RX7_PEND	0-1	RX7_PEND raw interrupt read (before mask).
6	RX6_PEND	0-1	RX6_PEND raw interrupt read (before mask).
5	RX5_PEND	0-1	RX5_PEND raw interrupt read (before mask).
4	RX4_PEND	0-1	RX4_PEND raw interrupt read (before mask).
3	RX3_PEND	0-1	RX3_PEND raw interrupt read (before mask).
2	RX2_PEND	0-1	RX2_PEND raw interrupt read (before mask).
1	RX1_PEND	0-1	RX1_PEND raw interrupt read (before mask).
0	RX0_PEND	0-1	RX0_PEND raw interrupt read (before mask).

5.63 CPDMA_INT RX Interrupt Status Register (Masked Value) (RX_INTSTAT_MASKED)

The CPDMA_INT RX interrupt status register (Masked Value) (RX_INTSTAT_MASKED) is shown in Figure 79 and described in Table 71.

Figure 79. CPDMA_INT RX Interrupt Status Register (Masked Value) (RX_INTSTAT_MASKED)

31-16
Reserved
R-0x0

15	14	13	12	11	10	9	8
RX7_THRESH_PEND	RX6_THRESH_PEND	RX5_THRESH_PEND	RX4_THRESH_PEND	RX3_THRESH_PEND	RX2_THRESH_PEND	RX1_THRESH_PEND	RX0_THRESH_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

7	6	5	4	3	2	1	0
RX7_PEND	RX6_PEND	RX5_PEND	RX4_PEND	RX3_PEND	RX2_PEND	RX1_PEND	RX0_PEND
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 71. CPDMA_INT RX Interrupt Status Register (Masked Value) (RX_INTSTAT_MASKED) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	RX7_THRESH_PEND	0-1	RX7_THRESH_PEND masked interrupt read.
14	RX6_THRESH_PEND	0-1	RX6_THRESH_PEND masked interrupt read.
13	RX5_THRESH_PEND	0-1	RX5_THRESH_PEND masked interrupt read.
12	RX4_THRESH_PEND	0-1	RX4_THRESH_PEND masked interrupt read.
11	RX3_THRESH_PEND	0-1	RX3_THRESH_PEND masked interrupt read.
10	RX2_THRESH_PEND	0-1	RX2_THRESH_PEND masked interrupt read.
9	RX1_THRESH_PEND	0-1	RX1_THRESH_PEND masked interrupt read.
8	RX0_THRESH_PEND	0-1	RX0_THRESH_PEND masked interrupt read.
7	RX7_PEND	0-1	RX7_PEND masked interrupt read.
6	RX6_PEND	0-1	RX6_PEND masked interrupt read.
5	RX5_PEND	0-1	RX5_PEND masked interrupt read.
4	RX4_PEND	0-1	RX4_PEND masked interrupt read.
3	RX3_PEND	0-1	RX3_PEND masked interrupt read.
2	RX2_PEND	0-1	RX2_PEND masked interrupt read.
1	RX1_PEND	0-1	RX1_PEND masked interrupt read.
0	RX0_PEND	0-1	RX0_PEND masked interrupt read.

5.64 CPDMA_INT RX Interrupt Mask Set Register (RX_INTMASK_SET)

The CPDMA_INT RX interrupt mask set register (RX_INTMASK_SET) is shown in Figure 80 and described in Table 72.

Figure 80. CPDMA_INT RX Interrupt Mask Set Register (RX_INTMASK_SET)

31–16
Reserved
R-0x0

15	14	13	12	11	10	9	8
RX7_THRESH_ PEND_MASK	RX6_THRESH_ PEND_MASK	RX5_THRESH_ PEND_MASK	RX4_THRESH_ PEND_MASK	RX3_THRESH_ PEND_MASK	RX2_THRESH_ PEND_MASK	RX1_THRESH_ PEND_MASK	RX0_THRESH_ PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

7	6	5	4	3	2	1	0
RX7_PEND_MASK	RX6_PEND_MASK	RX5_PEND_MASK	RX4_PEND_MASK	RX3_PEND_MASK	RX2_PEND_MASK	RX1_PEND_MASK	RX0_PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

LEGEND: WSR = Write to Set and Read; R = Read only; -*n* = value after reset

Table 72. CPDMA_INT RX Interrupt Mask Set Register (RX_INTMASK_SET) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	RX7_THRESH_PEND_MASK	0-1	RX Channel 7 threshold pending interrupt mask. Write one to enable interrupt.
14	RX6_THRESH_PEND_MASK	0-1	RX Channel 6 threshold pending interrupt mask. Write one to enable interrupt.
13	RX5_THRESH_PEND_MASK	0-1	RX Channel 5 threshold pending interrupt mask. Write one to enable interrupt.
12	RX4_THRESH_PEND_MASK	0-1	RX Channel 4 threshold pending interrupt mask. Write one to enable interrupt.
11	RX3_THRESH_PEND_MASK	0-1	RX Channel 3 threshold pending interrupt mask. Write one to enable interrupt.
10	RX2_THRESH_PEND_MASK	0-1	RX Channel 2 threshold pending interrupt mask. Write one to enable interrupt.
9	RX1_THRESH_PEND_MASK	0-1	RX Channel 1 threshold pending interrupt mask. Write one to enable interrupt.
8	RX0_THRESH_PEND_MASK	0-1	RX Channel 0 threshold pending interrupt mask. Write one to enable interrupt.
7	RX7_PEND_MASK	0-1	RX Channel 7 pending interrupt mask. Write one to enable interrupt.
6	RX6_PEND_MASK	0-1	RX Channel 6 pending interrupt mask. Write one to enable interrupt.
5	RX5_PEND_MASK	0-1	RX Channel 5 pending interrupt mask. Write one to enable interrupt.
4	RX4_PEND_MASK	0-1	RX Channel 4 pending interrupt mask. Write one to enable interrupt.
3	RX3_PEND_MASK	0-1	RX Channel 3 pending interrupt mask. Write one to enable interrupt.
2	RX2_PEND_MASK	0-1	RX Channel 2 pending interrupt mask. Write one to enable interrupt.
1	RX1_PEND_MASK	0-1	RX Channel 1 pending interrupt mask. Write one to enable interrupt.
0	RX0_PEND_MASK	0-1	RX Channel 0 pending interrupt mask. Write one to enable interrupt.

5.65 CPDMA_INT RX Interrupt Mask Clear Register (RX_INTMASK_CLEAR)

The CPDMA_INT RX interrupt mask clear register (RX_INTMASK_CLEAR) is shown in Figure 81 and described in Table 73.

Figure 81. CPDMA_INT RX Interrupt Mask Clear Register (RX_INTMASK_CLEAR)

31–16
Reserved
R-0x0

15	14	13	12	11	10	9	8
RX7_THRESH_ PEND_MASK	RX6_THRESH_ PEND_MASK	RX5_THRESH_ PEND_MASK	RX4_THRESH_ PEND_MASK	RX3_THRESH_ PEND_MASK	RX2_THRESH_ PEND_MASK	RX1_THRESH_ PEND_MASK	RX0_THRESH_ PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

7	6	5	4	3	2	1	0
RX7_PEND_MASK	RX6_PEND_MASK	RX5_PEND_MASK	RX4_PEND_MASK	RX3_PEND_MASK	RX2_PEND_MASK	RX1_PEND_MASK	RX0_PEND_MASK
WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0	WSR-0x0

LEGEND: WSR = Write to Set and Read; R = Read only; -*n* = value after reset

Table 73. CPDMA_INT RX Interrupt Mask Clear Register (RX_INTMASK_CLEAR) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15	RX7_THRESH_PEND_MASK	0-1	RX Channel 7 threshold pending interrupt mask. Write one to disable interrupt.
14	RX6_THRESH_PEND_MASK	0-1	RX Channel 6 threshold pending interrupt mask. Write one to disable interrupt.
13	RX5_THRESH_PEND_MASK	0-1	RX Channel 5 threshold pending interrupt mask. Write one to disable interrupt.
12	RX4_THRESH_PEND_MASK	0-1	RX Channel 4 threshold pending interrupt mask. Write one to disable interrupt.
11	RX3_THRESH_PEND_MASK	0-1	RX Channel 3 threshold pending interrupt mask. Write one to disable interrupt.
10	RX2_THRESH_PEND_MASK	0-1	RX Channel 2 threshold pending interrupt mask. Write one to disable interrupt.
9	RX1_THRESH_PEND_MASK	0-1	RX Channel 1 threshold pending interrupt mask. Write one to disable interrupt.
8	RX0_THRESH_PEND_MASK	0-1	RX Channel 0 threshold pending interrupt mask. Write one to disable interrupt.
7	RX7_PEND_MASK	0-1	RX Channel 7 pending interrupt mask. Write one to disable interrupt.
6	RX6_PEND_MASK	0-1	RX Channel 6 pending interrupt mask. Write one to disable interrupt.
5	RX5_PEND_MASK	0-1	RX Channel 5 pending interrupt mask. Write one to disable interrupt.
4	RX4_PEND_MASK	0-1	RX Channel 4 pending interrupt mask. Write one to disable interrupt.
3	RX3_PEND_MASK	0-1	RX Channel 3 pending interrupt mask. Write one to disable interrupt.
2	RX2_PEND_MASK	0-1	RX Channel 2 pending interrupt mask. Write one to disable interrupt.
1	RX1_PEND_MASK	0-1	RX Channel 1 pending interrupt mask. Write one to disable interrupt.
0	RX0_PEND_MASK	0-1	RX Channel 0 pending interrupt mask. Write one to disable interrupt.

5.66 CPDMA_INT DMA Interrupt Status Register (Raw Value) (DMA_INTSTAT_RAW)

The CPDMA_INT DMA interrupt status register (Raw Value) (DMA_INTSTAT_RAW) is shown in Figure 82 and described in Table 74.

Figure 82. CPDMA_INT DMA Interrupt Status Register (Raw Value) (DMA_INTSTAT_RAW)

31 - 16
Reserved
R-0x0

15 - 2	1	0
Reserved	HOST_PEND	STAT_PEND
R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 74. CPDMA_INT DMA Interrupt Status Register (Raw Value) (DMA_INTSTAT_RAW) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	HOST_PEND	0-1	Host pending interrupt. Raw interrupt read (before mask).
0	STAT_PEND	0-1	Statistics pending interrupt. Raw interrupt read (before mask).

5.67 CPDMA_INT DMA Interrupt Status Register (Masked Value) (DMA_INTSTAT_MASKED)

The CPDMA_INT DMA interrupt status register (Masked Value) (DMA_INTSTAT_MASKED) is shown in Figure 83 and described in Table 75.

Figure 83. CPDMA_INT DMA Interrupt Status Register (Masked Value) (DMA_INTSTAT_MASKED)

31 - 16
Reserved
R-0x0

15 - 2	1	0
Reserved	HOST_PEND	STAT_PEND
R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 75. CPDMA_INT DMA Interrupt Status Register (Masked Value) (DMA_INTSTAT_MASKED) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	HOST_PEND	0-1	Host pending interrupt. Masked interrupt read.
0	STAT_PEND	0-1	Statistics pending interrupt. Masked interrupt read.

5.68 CPDMA_INT DMA Interrupt Mask Set Register (DMA_INTMASK_SET)

The CPDMA_INT DMA interrupt mask set register (DMA_INTMASK_SET) is shown in Figure 84 and described in Table 76.

Figure 84. CPDMA_INT DMA Interrupt Mask Set Register (DMA_INTMASK_SET)

31	16
Reserved	
R-0x0	

15	2	1	0
Reserved		HOST_ERR_INT_MASK	STAT_INT_MASK
R-0x0		WSR-0x0	WSR-0x0

LEGEND: WSR = Write to Set and Read; R = Read only; -*n* = value after reset

Table 76. CPDMA_INT DMA Interrupt Mask Set Register (DMA_INTMASK_SET) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	HOST_ERR_INT_MASK	0-1	Host error interrupt mask. Write one to enable interrupt.
0	STAT_INT_MASK	0-1	Statistics interrupt mask. Write one to enable interrupt.

5.69 CPDMA_INT DMA Interrupt Mask Clear Register (DMA_INTMASK_CLEAR)

The CPDMA_INT DMA interrupt mask clear register (DMA_INTMASK_CLEAR) is shown in Figure 85 and described in Table 77.

Figure 85. CPDMA_INT DMA Interrupt Mask Clear Register (DMA_INTMASK_CLEAR)

31	16
Reserved	
R-0x0	

15	2	1	0
Reserved		HOST_ERR_INT_MASK	STAT_INT_MASK
R-0x0		WCR-0x0	WCR-0x0

LEGEND: WCR = Write to Clear and read; R = Read only; -*n* = value after reset

Table 77. CPDMA_INT DMA Interrupt Mask Clear Register (DMA_INTMASK_CLEAR) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	HOST_ERR_INT_MASK	0-1	Host error interrupt mask. Write one to disable interrupt.
0	STAT_INT_MASK	0-1	Statistics interrupt mask. Write one to disable interrupt.

5.70 CPDMA_INT Receive Threshold Pending Register Channels 0-7 (RXn_PENDTHRESH)

The CPDMA_INT receive threshold pending register channels 0-7 (RXn_PENDTHRESH) is shown in Figure 86 and described in Table 78.

Figure 86. CPDMA_INT Receive Threshold Pending Register Channels 0-7 (RX*n*_PENDTHRESH)

31-16
Reserved
R-0x0

15-8	7-0
Reserved	RX*n*_PENDTHRESH
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 78. CPDMA_INT Receive Threshold Pending Register Channels 0-7 (RX*n*_PENDTHRESH) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved
7-0	RX*n*_PENDTHRESH	0-FFh	Receive flow threshold. This field contains the threshold value for issuing receive threshold pending interrupts (when enabled).

5.71 CPDMA_INT Receive Free Buffer Register Channels 0-7 (RXn_FREEBUFFER)

The CPDMA_INT receive free buffer register channels 0-7 (RX*n*_FREEBUFFER) is shown in Figure 87 and described in Table 79.

Figure 87. CPDMA_INT Receive Free Buffer Register Channels 0-7 (RX*n*_FREEBUFFER)

31-16
Reserved
R-0x0

15-0
RX*n*_FREEBUFFER
WI-0x0

LEGEND: WI = Write to Increment; R = Read only; -*n* = value after reset

Table 79. CPDMA_INT Receive Free Buffer Register Channels 0-7 (RX*n*_FREEBUFFER) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved.
15-0	RX*n*_FREEBUFFER	0-FFFFh	Receive free buffer count. This field contains the count of free buffers available. The RX*n*_PENDTHRESH value is compared with this field to determine if the receive threshold pending interrupt should be asserted (if enabled). This is a write to increment field. This field rolls over to zero on overflow. If receive threshold pending interrupts are used, the host must initialize this field to the number of available buffers (one register per channel). The port decrements (by the number of buffers in the received frame) the associated channel register for each received frame. This is a write to increment field. The host must write this field with the number of buffers that have been freed due to host processing.

5.72 CPDMA_STATERAM TX Channels 0-7 Head Descriptor Pointer Register (TXn_HDP)

The CPDMA_STATERAM TX channels 0-7 head descriptor pointer register (TX*n*_HDP) is shown in Figure 88 and described in Table 80.

Figure 88. CPDMA_STATERAM TX Channels 0-7 Head Descriptor Pointer Register (TX*n*_HDP)

31	0
TX*n*_HDP	
R/W-Unknown	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 80. CPDMA_STATERAM TX Channels 0-7 Head Descriptor Pointer Register (TX*n*_HDP) Field Descriptions

Bit	Field	Value	Description
31-0	TX*n*_HDP	0-FFFF FFFFh	Transmit channels 0-7 DMA head descriptor pointer. Writing a transmit DMA Buffer Descriptor address to a head pointer location initiates TX DMA operations in the queue for the selected channel. Writing to these locations when they are non-zero is an error (except at reset). Host software must initialize these locations to zero on reset.

5.73 CPDMA_STATERAM RX Channels 0-7 Head Descriptor Pointer Register (RXn_HDP)

The CPDMA_STATERAM RX channels 0-7 head descriptor pointer register (RX*n*_HDP) is shown in Figure 89 and described in Table 81.

Figure 89. CPDMA_STATERAM RX Channels 0-7 Head Descriptor Pointer Register (RX*n*_HDP)

31	0
RX*n*_HDP	
R/W-Unknown	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 81. CPDMA_STATERAM RX Channels 0-7 Head Descriptor Pointer Register (RX*n*_HDP) Field Descriptions

Bit	Field	Value	Description
31-0	RX*n*_HDP	0-FFFF FFFFh	Receive DMA head descriptor pointer. Writing a RX DMA Buffer Descriptor address to this location allows RX DMA operations in the selected channel when a channel frame is received. Writing to these locations when they are non-zero is an error (except at reset). Host software must initialize these locations to zero on reset.

5.74 CPDMA_STATERAM TX Channels 0-7 Completion Pointer Register (TXn_CP)

The CPDMA_STATERAM TX channels 0-7 completion pointer register (TX*n*_CP) is shown in Figure 90 and described in Table 82.

Figure 90. CPDMA_STATERAM TX Channels 0-7 Completion Pointer Register (TX*n*_CP)

31	0
TX*n*_CP	
R/W-Unknown	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 82. CPDMA_STATERAM TX Channels 0-7 Completion Pointer Register (TX*n*_CP) Field Descriptions

Bit	Field	Value	Description
31-0	TX*n*_CP	0-FFFF FFFFh	Transmit completion pointer register. This register is written by the host with the buffer descriptor address for the last buffer processed by the host during interrupt processing. The port uses the value written to determine if the interrupt should be de-asserted.

5.75 CPDMA_STATERAM RX Channels 0-7 Completion Pointer Register (RXn_CP)

The CPDMA_STATERAM RX channels 0-7 completion pointer register (RX*n*_CP) is shown in Figure 91 and described in Table 83.

Figure 91. CPDMA_STATERAM RX Channels 0-7 Completion Pointer Register (RX*n*_CP)

31	0
RX*n*_CP	
R/W-Unknown	

LEGEND: R/W = Read/Write; -*n* = value after reset

Table 83. CPDMA_STATERAM RX Channels 0-7 Completion Pointer Register (RX*n*_CP) Field Descriptions

Bit	Field	Value	Description
31-0	RX*n*_CP	0-FFFF FFFFh	Receive completion pointer register. This register is written by the host with the buffer descriptor address for the last buffer processed by the host during interrupt processing. The port uses the value written to determine if the interrupt should be de-asserted.

5.76 Statistics Interface Registers

The statistics interrupt (STAT_PEND) will be issued if enabled when any statistics value is greater than or equal to 8000 0000h. The statistics interrupt is removed by writing to decrement any statistics value greater than 8000 0000h. The statistics are mapped into internal memory space and are 32-bits wide. All statistics rollover from FFFF FFFFh to 0000 0000h.

The maximum and minimum transmit (TX) frame size is software controllable.

The statistics register is shown in Figure 92.

Figure 92. Statistics Register

31 … 0
COUNT
R/W-0x0

LEGEND: R/W = Read/Write; -*n* = value after reset

5.76.1 3pGSw_STATS Total Number of Good Frames Received Register (RXGOODFRAMES)

Receive frames are the total number of good frames received on the port. A good frame is defined as having all of the following:

- Any data or MAC control frame which is destined for address FFFF FFFF FFFFh
- From 64 up to and including RX_MAXLEN bytes in length
- No CRC error, alignment error, or code error

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.2 3pGSw_STATS Total Number of Good Broadcast Frames Received Register (RXBROADCASTFRAMES)

Broadcast frames are the total number of good broadcast frames received on the port. A good broadcast frame is defined as having all of the following:

- Any data or MAC control frame which is destined for address FFFF FFFF FFFFh
- From 64 up to and including RX_MAXLEN bytes in length
- No CRC error, alignment error, or code error

  See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.3 3pGSw_STATS Total Number of Good Multicast Frames Received Register (RXMULTICASTFRAMES)

Multicast frames are the total number of good multicast frames received on the port. A good multicast frame is defined as having all of the following:

- Any data or MAC control frame which is destined for any multicast address other than FFFF FFFF FFFFh
- From 64 up to and including RX_MAXLEN bytes in length
- No CRC error, alignment error, or code error

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.4 3pGSw_STATS PauseRxFrames Register (RXPAUSEFRAMES)

Pause receive frames are the total number of IEEE 802.3X pause frames received by the port (whether acted upon or not). A good pause frame is defined as having all of the following:

- Any unicast, broadcast, or multicast address
- must contain the length/type field value 88.08 (hex) and the opcode 0001h
- From 64 up to and including RX_MAXLEN bytes in length
- No CRC error, alignment error, or code error
- Pause frames are enabled on the port (TX_FLOW_EN bit in the GMAC*n* MAC control register (GMAC*n*_MACCONTROL) = 1)

The port may be in either half or full-duplex mode. See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.5 3pGSw_STATS Total Number of CRC Errors Frames Received Register (RXCRCERRORS)

Receive CRC errors are the total number of frames received on the port that experienced a CRC error. Receive CRC errors are:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- From 64 up to and including RX_MAXLEN bytes in length
- No code/align error
- Has a CRC error

Overruns have no effect upon this statistic. A CRC error is defined as:

- A frame containing an even number of nibbles
- Fails the Frame Check Sequence test

5.76.6 3pGSw_STATS Total Number of Alignment/Code Errors Received Register (RXALIGNCODEERRORS)

Receive alignment/code errors are the total number of frames received on the port that experience an alignment error or code error. Receive alignment/code error frames are:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode and
- From 64 up to and including RX_MAXLEN bytes in length
- Has either an alignment error or a code error.

Over-runs have no effect upon this statistic. An alignment error is defined as:

- A frame containing an odd number of nibbles
- Fails the Frame Check Sequence test if the final nibble is ignored

A code error is defined to be a frame which has been discarded because the port's MRXER pin driven with a one for at least one bit-time's duration at any point during the frame's reception.

Note: RFC 1757 etherStatsCRCAlignErrors Ref. 1.5 can be calculated by summing RXALIGNCODEERRORS and RXCRCERRORS.

5.76.7 3pGSw_STATS Total Number of Oversized Frames Received Register (RXOVERSIZEDFRAMES)

Oversize receive frames are the total number of oversized frames received on the port. A good oversized frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- Greater than RX_MAXLEN in bytes
- No CRC error, alignment error, or code error

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.8 3pGSw_STATS Total Number of Jabber Frames Received Register (RXJABBERFRAMES)

Receive jabbers are the total number of jabber frames received on the port. A good jabber frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- Greater than RX_MAXLEN bytes long
- Has a CRC error, an alignment error, or a code error

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.9 3pGSw_STATS Total Number of Undersized Frames Received Register (RXUNDERSIZEDFRAMES)

Undersize (short) receive frames are the total number of undersized frames received on the port. A good undersized frame is defined as having all of the following:

- Any data frame which matches a unicast, broadcast, or multicast address, or is matched due to promiscuous mode
- Less than 64 bytes long
- No CRC error, alignment error, or code error

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.10 3pGSw_STATS RxFragments Received Register (RXFRAGMENTS)

Receive fragments are the total number of frame fragments received on the port. A good frame fragment is defined as having all of the following:

- Any data frame (address matching does not matter)
- Less than 64 bytes long
- Has a CRC error, an alignment error, or a code error
- Not the result of a collision caused by half duplex, collision based flow control

See the Section 5.76.6 and Section 5.76.5 statistic descriptions for definitions of alignment, code, and CRC errors. Overruns have no effect upon this statistic.

5.76.11 3pGSw_STATS Total Number of Received Bytes in Good Frames Register (RXOCTETS)

Receive octets are the total number of bytes in all good frames received on the port. A good frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- From 64 up to and including RX_MAXLEN bytes in length
- No CRC error, alignment error, or code error

5.76.12 3pGSw_STATS GoodTXFrames Register (TXGOODFRAMES)

Transmit frames are the total number of good frames transmitted on the port. A good frame is defined as having all of the following:

- Any data or MAC control frame which is destined for any unicast, broadcast, or multicast address
- Any length
- No late or excessive collisions, no carrier loss, and no underrun

5.76.13 3pGSw_STATS BroadcastTXFrames Register (TXBROADCASTFRAMES)

Broadcast transmit frames are the total number of good broadcast frames transmitted on the port. A good broadcast frame is defined as having all of the following:

- Any data or MAC control frame destined for address FFFF FFFF FFFFh
- Any length
- No late or excessive collisions, no carrier loss, and no underrun

5.76.14 3pGSw_STATS MulticastTXFrames Register (TXMULTICASTFRAMES)

Multicast transmit frames are the total number of good multicast frames transmitted on the port. A good multicast frame is defined as having all of the following:

- Any data or MAC control frame destined for any multicast address other than FFFF FFFF FFFFh
- Any length
- No late or excessive collisions, no carrier loss, and no underrun

5.76.15 3pGSw_STATS PauseTXFrames Register (TXPAUSEFRAMES)

This statistic indicates the number of IEEE 802.3X pause frames transmitted by the port. Pause frames cannot underrun or contain a CRC error because they are created in the transmitting MAC. These error conditions have no effect upon the statistic. Pause frames sent by software will not be included in this count.

Since pause frames are only transmitted in full duplex carrier loss and collisions have no effect upon this statistic, transmitted pause frames are always 64 byte multicast frames so they will appear in the transmit multicast frame and 64 octect frame statistics.

5.76.16 3pGSw_STATS Deferred Frames Register (TXDEFERREDFRAMES)

Deferred transmit frames are the total number of frames transmitted on the port that first experience deferment. A good deferred transmit frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- No carrier loss and no underrun
- No collisions before being successfully transmitted
- Find the medium busy when transmission is first attempted forcing the frames to wait

CRC errors have no effect upon this statistic. See RFC1623 Ref. 2.6 dot3StatsDefferredTransmissions.

5.76.17 3pGSw_STATS Collisions Register (TXCOLLISIONFRAMES)

This statistic records the total number of times that the port experiences a collision. Collisions occur under two circumstances as described below.

1. When a transmit data or MAC control frame:
   - Destined for any unicast, broadcast, or multicast address
   - Any size
   - No carrier loss and no underrun
   - Experiences a collision

   A jam sequence is sent for every non-late collision, so this statistic will increment on each occasion if a frame experiences multiple collisions (and increments on late collisions). CRC errors have no effect upon this statistic.
2. When the port is in half-duplex mode, flow control is active, and a frame reception begins.

5.76.18 3pGSw_STATS SingleCollisionTXFrames Register (TXSINGLECOLLFRAMES)

Single collision transmit frames are the total number of frames transmitted on the port that experience exactly one collision. A single collision transmit frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- No carrier loss and no underrun
- Experience one collision before a successful transmission
- The collision is not late

CRC errors have no effect upon this statistic.

5.76.19 3pGSw_STATS MultipleCollisionTXFrames Register (TXMULTCOLLFRAMES)

Multiple collision transmit frames are the total number of frames transmitted on the port that experience multiple collisions. A multiple collision transmit frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- No carrier loss and no underrun
- Experience 2 to 15 collisions before being successfully transmitted
- None of the collisions are late

CRC errors have no effect upon this statistic.

5.76.20 3pGSw_STATS ExcessiveCollisions Register (TXEXCESSIVECOLLISIONS)

Excessive collisions are the total number of frames in which transmission is abandoned due to excessive collisions. An excessive collision frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- No carrier loss and no underrun
- Experience 16 collisions before abandoning all attempts at transmitting the frame.
- None of the collisions are late.

CRC errors have no effect upon this statistic.

5.76.21 3pGSw_STATS LateCollisions Register (TXLATECOLLISIONS)

Late collisions are the total number of frames on the port in which transmission is abandoned because the frames experience a late collision. A late collision frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- Experience a collision later than 512 bit-times into the transmission.
  Note: There may be up to 15 previous (non-late) collisions which had previously required the transmission to be re-attempted.

The Late Collisions statistic dominates over the single, multiple, and excessive collisions statistics. If a late collision occurs, the frame will not be counted in any of these other three statistics. CRC errors, carrier loss, and underrun have no effect upon this statistic.

5.76.22 3pGSw_STATS Transmit Underrun Error Register (TXUNDERRUN)

There should be no transmitted frames that experience underrun.

5.76.23 3pGSw_STATS CarrierSenseErrors Register (TXCARRIERSENSEERRORS)

Carrier sense errors are the total number of frames on the port that experience carrier loss. A carrier sense error frame is defined as having all of the following:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address
- Any size
- Carrier sense condition is lost or never asserted when transmitting the frame (the frame is not re-transmitted). This statistic only transmits frames. Carrier sense errors ignore received frames.

Transmit frames with carrier sense errors are sent until completion and are not aborted. CRC errors and underrun have no effect upon this statistic.

5.76.24 3pGSw_STATS TXOctets Register (TXOCTETS)

Transmit octets are the total number of bytes in all good frames transmitted on the port. A good frame is defined as having all of the following:

- Any data or MAC control frame which is destined for any unicast, broadcast, or multicast address
- Any size
- No late or excessive collisions, no carrier loss, and no underrun.

5.76.25 3pGSw_STATS 64OctetFrames Register (OCTETFRAMES64)

Receive and transmit 64 octet frames are the total number of 64-byte frames received and transmitted on the port. Receive and transmit 64 octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- Exactly 64 bytes long. (If the frame being transmitted experiences carrier loss that results in a 64-byte frame being transmitted, then the frame will be recorded in this statistic).

CRC errors, code/align errors, and overruns do not affect the recording of frames in this statistic.

5.76.26 3pGSw_STATS 65-127OctetFrames Register (OCTETFRAMES65T127)

Receive and transmit 65-127 octet frames are the total number of frames from 65 to 127 bytes long that are received and transmitted on the port. Receive and transmit 65-127 octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- From 65 to 127 bytes long

CRC errors, code/align errors, underruns, and overruns do not affect the recording of frames in this statistic.

5.76.27 3pGSw_STATS 128-255OctetFrames Register (OCTETFRAMES128T255)

Receive and transmit 128-255 octet frames are the total number of frames from 128 to 255 bytes long that are received and transmitted on the port. Receive and transmit 128-255 octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- From 128 to 255 bytes long

CRC errors, code/align errors, underruns, and overruns do not affect the recording of frames in this statistic.

5.76.28 3pGSw_STATS 256-511OctetFrames Register (OCTETFRAMES256T511)

Receive and transmit 256-511 octet frames are the total number of frames from 256 to 511 bytes long that are received and transmitted on the port. Receive and transmit 256-511 octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- From 256 to 511 bytes long

CRC errors, code/align errors, underruns, and overruns do not affect the recording of frames in this statistic.

5.76.29 3pGSw_STATS 512-1023OctetFrames Register (OCTETFRAMES512T1023)

Receive and transmit 512-1023 octet frames are the total number of frames from 512 to 1023 bytes long that are received and transmitted on the port. Receive and transmit 512-1023 octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- From 512 to 1023 bytes long

CRC errors, code/align errors, and overruns do not affect the recording of frames in this statistic.

5.76.30 3pGSw_STATS 1023-1518OctetFrames Register (OCTETFRAMES1024TUP)

Receive and transmit 1024 and up octet frames are the total number of frames from 1024 to RX_MAXLEN bytes long for receive or from 1024 bytes and longer for transmit on the port. Receive and transmit 1024 and up octet frames are defined as having all of the following:

- Any data or MAC control frame which are destined for any unicast, broadcast, or multicast address
- No late collisions, excessive collisions, or carrier sense errors
- From 1024 to RX_MAXLEN bytes long on receive, or are longer than or equal to 1024 byte size on transmit.

CRC errors, code/align errors, underruns, and overruns do not affect the recording of frames in this statistic.

5.76.31 3pGSw_STATS NetOctets Register (NETOCTETS)

Net octets are the total number of bytes of frame data received and transmitted on the port. Frames counted include:

- Any data or MAC control frame destined for any unicast, broadcast, or multicast address (address match does not matter)
- Any size (including <64 byte and > RX_MAXLEN byte frames).

Also counted in this statistic are:

- Every byte transmitted before a carrier-loss was experienced,
- Every byte transmitted before each collision was experienced (i.e. multiple retries are counted each time),
- Every byte received if the port is in half-duplex mode until a jam sequence is transmitted to initiate flow control. (The jam sequence was not counted to prevent double-counting).

Error conditions such as alignment errors, CRC errors, code errors, overruns, and underruns do not affect the recording of bytes by this statistic. The objective of this statistic is to give a reasonable indication of Ethernet utilization.

5.76.32 3pGSw_STATS Receive FIFO or DMA Start of Frame Overruns Register (RXSOFOVERRUNS)

Receive start of frame overruns are the total number of frames received on the port that have a CPDMA start of frame (SOF) overrun or are dropped due to FIFO resource limitations. A SOF overrun frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- Any length (including < 64 bytes and > RX_MAXLEN bytes)
- The CPDMA has a start of frame overrun or the packet is dropped due to FIFO resource limitations.

5.76.33 3pGSw_STATS Receive FIFO or DMA Mid of Frame Overruns Register (RXMOFOVERRUNS)

Receive middle of frame overruns are the total number of frames received on the port that have a CPDMA middle of frame (MOF) overrun. A MOF overrun frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- Any length (including <64 bytes and > RX_MAXLEN bytes)
- The CPDMA has a middle of frame overrun.

5.76.34 3pGSw_STATS Receive DMA Start of Frame and Middle of Frame Overruns Register (RXDMAOVERRUNS)

Receive DMA overruns are the total number of frames received on the port that have either a DMA start of frame (SOF) overrun or a DMA MOF overrun. A receive DMA overrun frame is defined as having all of the following:

- Any data or MAC control frame which matches a unicast, broadcast, or multicast address or is matched due to promiscuous mode
- Any length (including <64 bytes and > RX_MAXLEN bytes)
- The GMAC*n* is unable to receive it because it does not have the DMA buffer resources to receive it (zero head descriptor pointer at the start or during the middle of the frame reception).

CRC errors, alignment errors, and code errors have no effect upon this statistic.

5.77 Address Lookup Engine ID/Version Register (DM648 only) (ALE_IDVER)

The address lookup engine ID/Version register (ALE_IDVER) is shown in Figure 93 and described in Table 84.

Figure 93. Address Lookup Engine ID/Version Register (DM648 only) (ALE_IDVER)

31	16
ALE_IDENT	
R-0x29	

15	8	7	0
ALE_MAJ_VER		ALE_MINOR_VER	
R-0x01		R-0x01	

LEGEND: R = Read only; -*n* = value after reset

Table 84. Address Lookup Engine ID/Version Register (DM648 only) (ALE_IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	ALE_IDENT	0-FFFFh	ALE identification value.
15-8	ALE_MAJ_VER	0-FFh	ALE major version value.
7-0	ALE_MINOR_VER	0-FFh	ALE minor version value.

5.78 Address Lookup Engine Control Register (DM648 only) (ALE_CONTROL)

The address lookup engine control register (ALE_CONTROL) is shown in Figure 94 and described in Table 85.

Figure 94. Address Lookup Engine Control Register (DM648 only) (ALE_CONTROL)

31	30	29	28 16
ENABLE_ALE	CLEAR_TABLE	AGE_OUT_NOW	Reserved
R/W-0x0	R/W-0x0	R/W-0x0	R-0x0

15 8
Reserved
R-0x0

7	6	5	4	3	2	1	0
LEARN_NO_VID	EN_VID0_MODE	ENABLE_OUI_DENY	ALE_BYPASS	RATE_LIMIT_TX	ALE_VLAN_AWARE	ENABLE_AUTH_ MODE	ENABLE_RATE_ LIMIT
R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 85. Address Lookup Engine Control Register (DM648 only) (ALE_CONTROL) Field Descriptions

Bit	Field	Value	Description
31	ENABLE_ALE		Enable ALE.
		0	Drop all packets. The GMAC*n* is not in the idle state.
		1	Enable ALE packet processing. The GMAC*n* is in the idle state.
30	CLEAR_TABLE	0-1	Clear ALE address table. Setting this bit causes the ALE hardware to write all table bit values to zero. Software must perform a clear table operation as part of the ALE setup/configuration process. Setting this bit causes all ALE accesses to be held up for 64 clocks while the clear is performed. Access to all ALE registers will be blocked (wait states) until the 64 clocks have completed. This bit cannot be read as one because the read is blocked until the clear table is completed at which time this bit is cleared to zero.
29	AGE_OUT_NOW	0-1	Age out address table now. Setting this bit causes the ALE hardware to remove (free up) any ageable table entry that does not have a set touch bit. This bit is cleared when the age out process has completed. This bit may be read. The age out process takes 4096 clocks in the best case (no ale packet processing during ageout) and 66550 clocks at the absolute worst case.
28-8	Reserved	0	Reserved.
7	LEARN_NO_VID		Learn no VID.
		0	VID is learned with the source address.
		1	VID is not learned with the source address (source address is not tied to VID).
6	EN_VID0_MODE		Enable VLAN ID = 0 mode.
		0	Process the packet with VID = PORT_VID in the P*n*_PORT_VLAN register.
		1	Process the packet with VID = 0.
5	ENABLE_OUI_DENY	0-1	Enable OUI deny mode. When set, this bit indicates that a packet with a non OUI table entry matching the source address will be dropped to the host unless the destination address matches a multicast table entry with the SUPER bit set.
4	ALE_BYPASS		ALE bypass. This bit causes all packets to be forwarded only to the host port. Packets from GMAC1 will be designated as supervisory so that they may be sent to a different CPDMA channel than the packets from GMAC0.
		0	ALE is in normal mode.
		1	ALE is in bypass mode.

Table 85. Address Lookup Engine Control Register (DM648 only) (ALE_CONTROL) Field Descriptions (continued)

Bit	Field	Value	Description
3	RATE_LIMIT_TX		Rate limit transmit mode.
		0	Broadcast and multicast rate limit counters are received port based.
		1	Broadcast and multicast rate limit counters are transmit port based.
2	ALE_VLAN_AWARE		ALE VLAN aware. Determines what is done if VLAN is not found.
		0	Flood if VLAN is not found.
		1	Drop packet if VLAN is not found.
1	ENABLE_AUTH_MODE		Enable MAC authorization mode. MAC authorization mode requires that all table entries be made by the host software. There are no learned addresses in the authorization mode and the packet will be dropped if the source address is not found (and the destination address is not a multicast address with the SUPER table entry bit set).
		0	The ALE is not in MAC authorization mode.
		1	The ALE is in MAC authorization mode.
0	ENABLE_RATE_LIMIT		Enable broadcast and multicast rate limit.
		0	Broadcast/Multicast rates not limited.
		1	Broadcast/Multicast packet reception limited to the port control register rate limit fields.

5.79 Address Lookup Engine Prescale Register (DM648 only) (ALE_PRESCALE)

The address lookup engine prescale register (ALE_PRESCALE) is shown in Figure 95 and described in Table 86.

Figure 95. Address Lookup Engine Prescale Register (DM648 only) (ALE_PRESCALE)

31 20	19 16
Reserved	ALE_PRESCALE
R-0x0	R/W-0x0

15 0
ALE_PRESCALE
R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 86. Address Lookup Engine Prescale Register (DM648 only) (ALE_PRESCALE) Field Descriptions

Bit	Field	Value	Description
31-20	Reserved	0	Reserved.
19-0	ALE_PRESCALE	0-F FFFFh	ALE prescale register. The input clock is divided by this value for use in the multicast/broadcast rate limiters. The minimum operating value is 10h. The prescaler is off when the value is zero.

5.80 Address Lookup Engine UnKnown VLAN Register (DM648 only) (ALE_UNKNOWN_VLAN)

The address lookup engine unKnown VLAN register (ALE_UNKNOWN_VLAN) is shown in Figure 96 and described in Table 87.

Figure 96. Address Lookup Engine UnKnown VLAN Register (DM648 only) (ALE_UNKNOWN_VLAN)

31 27	26 24	23 19	18 16
Reserved	UNKNOWN_FORCE_UNTAGGED_EGRESS	Reserved	UNKNOWN_REG_MCAST_FLOOD_MASK
R-0x0	R/W-0x0	R-0x0	R/W-0x0

15 11	10 8	7 3	2 0
Reserved	UNKNOWN_MCAST_FLOOD_MASK	Reserved	UNKNOWN_VLAN_MEMBER_LIST
R-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 87. Address Lookup Engine UnKnown VLAN Register (DM648 only) (ALE_UNKNOWN_VLAN) Field Descriptions

Bit	Field	Value	Description
31-27	Reserved	0	Reserved.
26-24	UNKNOWN_FORCE_UNTAGGED_EGRESS	0-7h	Unknown VLAN force untagged egress (except on port 2).
23-19	Reserved	0	Reserved.
18-16	UNKNOWN_REG_MCAST_FLOOD_MASK	0-7h	Unknown VLAN registered multicast flood mask.
15-11	Reserved	0	Reserved.
10-8	UNKNOWN_MCAST_FLOOD_MASK	0-7h	Unknown VLAN multicast flood mask.
7-3	Reserved	0	Reserved.

Table 87. Address Lookup Engine UnKnown VLAN Register (DM648 only) (ALE_UNKNOWN_VLAN) Field Descriptions (continued)

Bit	Field	Value	Description
2-0	UNKNOWN_VLAN_MEMBER_LIST	0-7h	Unknown VLAN member list.

5.81 Address Lookup Engine Table Control Register (DM648 only) (ALE_TBLCTL)

The address lookup engine table control register (ALE_TBLCTL) is shown in Figure 97 and described in Table 88.

Figure 97. Address Lookup Engine Table Control Register (DM648 only) (ALE_TBLCTL)

31	30-16
WRITE_RDZ	Reserved
R/W-0x0	R-0x0

15-10	9-0
Reserved	ENTRY_POINTER
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 88. Address Lookup Engine Table Control Register (DM648 only) (ALE_TBLCTL) Field Descriptions

Bit	Field	Value	Description
31	WRITE_RDZ		Write bit. This bit is always read as zero. Writing a 1 to this bit causes the three table word register values to be written to the ENTRY_POINTER location in the address table. Writing a 0 to this bit causes the three table word register values to be loaded from the ENTRY_POINTER location in the address table so that they may be subsequently read. A read of any ALE address location will be stalled until the read or write has completed.
		0	The GMAC*n* is not in the idle state.
		1	The GMAC*n* is in the idle state.
30-10	Reserved	0	Reserved.
9-0	ENTRY_POINTER	0-3FFh	Table entry pointer. The ENTRY_POINTER contains the table entry value that will be read/written with accesses to the table word registers.

5.82 Address Lookup Engine Table Word 2 Register (DM648 only) (ALE_TBLW2)

The address lookup engine table word 2 register (ALE_TBLW2) is shown in Figure 98 and described in Table 89.

Figure 98. Address Lookup Engine Table Word 2 Register (DM648 only) (ALE_TBLW2)

31-16
Reserved
R-0x0

15-4	3-0
Reserved	ENTRY[67:64]
R-0x0	R/W-0xX

LEGEND: R/W = Read/Write; X = Unidentified; R = Read only; -*n* = value after reset

Table 89. Address Lookup Engine Table Word 2 Register (DM648 only) (ALE_TBLW2) Field Descriptions

Bit	Field	Value	Description
31-4	Reserved	0	Reserved.
3-0	ENTRY[67:64]	0-Fh	Table entry bits 67:64.

5.83 Address Lookup Engine Table Word 1 Register (DM648 only) (ALE_TBLW1)

The address lookup engine table word 1 register (ALE_TBLW1) is shown in Figure 99 and described in Table 90.

Figure 99. Address Lookup Engine Table Word 1 Register (DM648 only) (ALE_TBLW1)

31	0
ENTRY[63:32]	
R/W-X	

LEGEND: R/W = Read/Write; X = Unidentified; -*n* = value after reset

Table 90. Address Lookup Engine Table Word 1 Register (DM648 only) (ALE_TBLW1) Field Descriptions

Bit	Field	Value	Description
31-0	ENTRY[63:32]	0-FFFF FFFFh	Table entry bits 63:32.

5.84 Address Lookup Engine Table Word 0 Register (DM648 only) (ALE_TBLW0)

The address lookup engine table word 0 register (ALE_TBLW0) is shown in Figure 100 and described in Table 91.

Figure 100. Address Lookup Engine Table Word 0 Register (DM648 only) (ALE_TBLW0)

31	0
ENTRY[31:0]	
R/W-0xX	

LEGEND: R/W = Read/Write; X = Unidentified; -*n* = value after reset

Table 91. Address Lookup Engine Table Word 0 Register (DM648 only) (ALE_TBLW0) Field Descriptions

Bit	Field	Value	Description
31-0	ENTRY[31:0]	0-FFFF FFFFh	Table entry bits 31:0.

5.85 Address Lookup Engine Port 0 Control Register (DM648 only) (ALE_PORTCTL0)

The address lookup engine port 0 control register (ALE_PORTCTL0) is shown in Figure 101 and described in Table 92.

Figure 101. Address Lookup Engine Port 0 Control Register (DM648 only) (ALE_PORTCTL0)

31 24	23 16
BCAST_LIMIT	MCAST_LIMIT
R/W-0x0	R/W-0x0

15   8
Reserved
R-0x0

7		5	4	3	2	1
	Reserved		NO_LEARN	VID_INGRESS_CHECK	DROP_UNTAGGED	PORT_STATE
	R-0x0		R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 92. Address Lookup Engine Port 0 Control Register (DM648 only) (ALE_PORTCTL0) Field Descriptions

Bit	Field	Value	Description
31-24	BCAST_LIMIT		Broadcast packet rate limit. Each prescale pulse loads this field into the port broadcast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Broadcast rate limiting is enabled by a non-zero value in this field.
		0	The GMAC is not in the idle state.
		1	The GMAC is in the idle state.
23-16	MCAST_LIMIT	0-1	Multicast packet rate limit. Each prescale pulse loads this field into the port multicast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Multicast rate limiting is enabled by a non-zero value in this field.
15-5	Reserved	0	Reserved.
4	NO_LEARN	0-1	No learn mode. When set, the port is disabled from learning or updating source addresses.
3	VID_INGRESS_CHECK	0-1	VLAN ID ingress check. If VLAN is not found, then drop the packet. Packets with an unknown (default) VLAN will be dropped.
2	DROP_UNTAGGED	0-1	Drop untagged packet. Drop non-VLAN tagged ingress packets.
1-0	PORT_STATE		Port state.
		0	Disabled.
		1	Blocked.
		2	Learn.
		3	Forward.

5.86 Address Lookup Engine Port 1 Control Register (DM648 only) (ALE_PORTCTL1)

The address lookup engine port 1 control register (ALE_PORTCTL1) is shown in Figure 102 and described in Table 93.

Figure 102. Address Lookup Engine Port 1 Control Register (DM648 only) (ALE_PORTCTL1)

31 24	23 16
BCAST_LIMIT	MCAST_LIMIT
R/W-0x0	R/W-0x0

15 8
Reserved
R-0x0

7		5	4	3	2	1
	Reserved		NO_LEARN	VID_INGRESS_CHECK	DROP_UNTAGGED	PORT_STATE
	R-0x0		R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 93. Address Lookup Engine Port 1 Control Register (DM648 only) (ALE_PORTCTL1) Field Descriptions

Bit	Field	Value	Description
31-24	BCAST_LIMIT		Broadcast packet rate limit. Each prescale pulse loads this field into the port broadcast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Broadcast rate limiting is enabled by a non-zero value in this field.
		0	The GMAC is not in the idle state.
		1	The GMAC is in the idle state.
23-16	MCAST_LIMIT	0-FFh	Multicast packet rate limit. Each prescale pulse loads this field into the port multicast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Multicast rate limiting is enabled by a non-zero value in this field.
15-5	Reserved	0	Reserved.
4	NO_LEARN	0-1	No learn mode. When set, the port is disabled from learning or updating source addresses.
3	VID_INGRESS_CHECK	0-1	VLAN ID ingress check. If VLAN is not found, then drop the packet. Packets with an unknown (default) VLAN will be dropped.
2	DROP_UNTAGGED	0-1	Drop untagged packet. Drop non-VLAN tagged ingress packets.
1-0	PORT_STATE		Port state.
		0	Disabled.
		1	Blocked.
		2	Learn.
		3	Forward.

5.87 Address Lookup Engine Port 2 Control Register (DM648 only) (ALE_PORTCTL2)

The address lookup engine port 2 control register (ALE_PORTCTL2) is shown in Figure 103 and described in Table 94.

Figure 103. Address Lookup Engine Port 2 Control Register (DM648 only) (ALE_PORTCTL2)

31 - 24	23 - 16
BCAST_LIMIT	MCAST_LIMIT
R/W-0x0	R/W-0x0

15 - 8
Reserved
R-0x0

7		5	4	3	2	1
	Reserved		NO_LEARN	VID_INGRESS_CHECK	DROP_UNTAGGED	PORT_STATE
	R-0x0		R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 94. Address Lookup Engine Port 2 Control Register (DM648 only) (ALE_PORTCTL2) Field Descriptions

Bit	Field	Value	Description
31-24	BCAST_LIMIT		Broadcast packet rate limit. Each prescale pulse loads this field into the port broadcast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Broadcast rate limiting is enabled by a non-zero value in this field.
		0	The GMAC is not in the idle state.
		1	The GMAC is in the idle state.
23-16	MCAST_LIMIT	0-FFh	Multicast packet rate limit. Each prescale pulse loads this field into the port multicast rate limit counter. The port counters are decremented with each packet received or transmitted depending on whether the mode is transmit or receive. If the counters decrement to zero, then further packets are rate limited until the next prescale pulse. Multicast rate limiting is enabled by a non-zero value in this field.
15-5	Reserved	0	Reserved.
4	NO_LEARN	0-1	No learn mode. When set, the port is disabled from learning or updating source addresses.
3	VID_INGRESS_CHECK	0-1	VLAN ID ingress check. If VLAN not found, then drop the packet. Packets with an unknown (default) VLAN will be dropped.
2	DROP_UNTAGGED	0-1	Drop untagged packet. Drop non-VLAN tagged ingress packets.
1-0	PORT_STATE		Port state.
		0	Disabled.
		1	Blocked.
		2	Learn.
		3	Forward.

6 3-Port Gigabit Switch Subsystem Registers (CPSW_3GSS) Registers

This section describes the memory-mapped registers for the 3GSS (CPSW_3GSS).

Table 95 lists the memory-mapped registers for the 3GSS (CPSW_3GSS). See the device-specific data manual for the memory address of these registers.

Table 95. 3-port Gigabit Switch Subsystem Registers (CPSW_3GSS) Registers

Offset	Acronym	Register Description	Section
0	IDVER	ID Version Register	Section 6.1
4h	SOFT_RESET	Soft Reset Register	Section 6.2
8h	EM_CONTROL	Emulation Control Register	Section 6.3
Ch	INT_CONTROL	Interrupt Control Register	Section 6.4
10h	RX_THRESH_EN	Receive Threshold Enable Register	Section 6.5
14h	RX_EN	Receive Interrupt Enable Register	Section 6.6
18h	TX_EN	Transmit Interrupt Enable Register	Section 6.7
1Ch	MISC_EN	Miscellaneous Interrupt Enable Register	Section 6.8
20h	RX_THRESH_STAT	Receive Threshold Status Register	Section 6.9
24h	RX_STAT	Receive Status Register	Section 6.10
28h	TX_STAT	Transmit Status Register	Section 6.11
2Ch	MISC_STAT	Miscellaneous Interrupt Status Register	Section 6.12
30h	RX_IMAX	Receive Interrupts per Millisecond Register	Section 6.13
34h	TX_IMAX	Transmit Interrupts per Millisecond Register	Section 6.14

6.1 ID Version Register (IDVER)

The ID version register (IDVER) is shown in Figure 104 and described in Table 96.

Figure 104. ID Version Register (IDVER)

31 16
CPGMACSS_S_IDENT
R-0x001A

15 11	10 8	7 0
RTL_VER	MAJOR_VER	MINOR_VER
R-0x0	R-0x1	R-0x2

LEGEND: R = Read only; -*n* = value after reset

Table 96. ID Version Register (IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	CPGMACSS_S_IDENT	0-FFFFh	3GSS identification value.
15-11	RTL_VER	0-1Fh	3GSS RTL_version register.
10-8	MAJOR_VER	0-7h	3GSS major version register.
7-0	MINOR_VER	0-FFh	3GSS minor version register.

6.2 Soft Reset Register (SOFT_RESET)

The soft reset register (SOFT_RESET) is shown in Figure 105 and described in Table 97.

Figure 105. Soft Reset Register (SOFT_RESET)

31 16
Reserved
R-0x0

15 1	0
Reserved	SOFT_RESET
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 97. Soft Reset Register (SOFT_RESET) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved.
0	SOFT_RESET	0-1	Software reset. Writing a one to this bit causes the CPSW_3GSS_S logic to be reset (INT, REGS, MDIO). Software reset occurs on the clock following the register bit write.

6.3 Emulation Control Register (EM_CONTROL)

The emulation control register (EM_CONTROL) is shown in Figure 106 and described in Table 98.

Figure 106. Emulation Control Register (EM_CONTROL)

31–16
Reserved
R-0x0

15–2	1	0
Reserved	SOFT	FREE
R-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 98. Emulation Control Register (EM_CONTROL) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1	SOFT	0-1	Emulation soft bit.
0	FREE	0-1	Emulation free bit.

6.4 Interrupt Control Register (INT_CONTROL)

The interrupt control register (INT_CONTROL) is shown in Figure 107 and described in Table 99.

Figure 107. Interrupt Control Register (INT_CONTROL)

31	30–22	21	20	19–16
INT_TEST	Reserved	TX_PULSE_INT_PACE_EN	RX_PULSE_INT_PACE_EN	Reserved
R/W-0x0	R-0x0	R/W-0x0	R/W-0x0	R-0x0

15–12	11–0
Reserved	INT_PRESCALE
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 99. Interrupt Control Register (INT_CONTROL) Field Descriptions

Bit	Field	Value	Description
31	INT_TEST	0-1	Interrupt test. Test bit to the interrupt pacing blocks.
30-22	Reserved	0	Reserved
21	TX_PULSE_INT_PACE_EN		TX _PULSE interrupt pacing enable
		0	Disables TX _PULSE interrupt pacing
		1	Enables TX _PULSE interrupt pacing
20	RX_PULSE_INT_PACE_EN		RX _PULSE interrupt pacing enable
		0	Disables RX _PULSE interrupt pacing
		1	Enables RX _PULSE interrupt pacing
19-12	Reserved	0	Reserved
11-0	INT_PRESCALE	0-FFFh	Interrupt counter prescaler. The number of subsystem clock periods in 4µs.

6.5 Receive Threshold Enable Register (RX_THRESH_EN)

The receive threshold enable register (RX_THRESH_EN) is shown in Figure 108 and described in Table 100.

Figure 108. Receive Threshold Enable Register (RX_THRESH_EN)

31 16
Reserved
R-0x0

15 8	7 0
Reserved	RX_THRESH_EN
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 100. Receive Threshold Enable Register (RX_THRESH_EN) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	RX_THRESH_EN	0-FFh	Receive threshold enable. Each bit in this register corresponds to the bit in the receive threshold interrupt that is enabled to generate an interrupt on RX_THRESH_PULSE.

6.6 Receive Interrupt Enable Register (RX_EN)

The receive interrupt enable register (RX_EN) is shown in Figure 109 and described in Table 101.

Figure 109. Receive Interrupt Enable Register (RX_EN)

31 16
Reserved
R-0x0

15 8	7 0
Reserved	RX_EN
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 101. Receive Interrupt Enable Register (RX_EN) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	RX_EN	0-FFh	Receive enable. Each bit in this register corresponds to the bit in the receive interrupt that is enabled to generate an interrupt on RX_PULSE.

6.7 Transmit Interrupt Enable Register (TX_EN)

The transmit interrupt enable register (TX_EN) is shown in Figure 110 and described in Table 102.

Figure 110. Transmit Interrupt Enable Register (TX_EN)

31	16
Reserved	
R-0x0	

15	8	7	0
Reserved		TX_EN	
R-0x0		R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 102. Transmit Interrupt Enable Register (TX_EN) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	TX_EN	0-FFh	Transmit enable. Each bit in this register corresponds to the bit in the transmit interrupt that is enabled to generate an interrupt on TX_PULSE.

6.8 Miscellaneous Interrupt Enable Register (MISC_EN)

The miscellaneous interrupt enable register (MISC_EN) is shown in Figure 111 and described in Table 103.

Figure 111. Miscellaneous Interrupt Enable Register (MISC_EN)

31 16
Reserved
R-0x0

15 6	5	4	3	2	1	0
Reserved	STAT_PEND_EN	HOST_PEND_EN	Reserved	MDIO_LINKINT_EN	Reserved	MDIO_USERINT_EN
R-0x0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 103. Miscellaneous Interrupt Enable Register (MISC_EN) Field Descriptions

Bit	Field	Value	Description
31-6	Reserved	0	Reserved
5	STAT_PEND_EN		Statistics interrupt enable
		0	Disables the interrupt
		1	Enables the interrupt
4	HOST_PEND_EN		Host error interrupt enable
		0	Disables the interrupt
		1	Enables the interrupt
3	Reserved	0	Reserved (Writes should have a zero)
2	MDIO_LINKINT_EN		MDIO link interrupt enable
		0	Disables the interrupt
		1	Enables the interrupt
1	Reserved	0	Reserved (Writes should have a zero)
0	MDIO_USERINT_EN		MDIO user interrupt enable
		0	Disables the interrupt
		1	Enables the interrupt

6.9 Receive Threshold Status Register (RX_THRESH_STAT)

The receive threshold status register (RX_THRESH_STAT) is shown in Figure 112 and described in Table 104.

Figure 112. Receive Threshold Status Register (RX_THRESH_STAT)

31	16
Reserved	
R-0x0	

15	8	7	0
Reserved		RX_THRESH_STAT	
R-0x0		R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 104. Receive Threshold Status Register (RX_THRESH_STAT) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	RX_THRESH_STAT	0-FFh	Receive threshold masked interrupt status. Each bit in this read only register corresponds to the bit in the receive threshold interrupt that is enabled and generating an interrupt on RX_THRESH_PULSE.

6.10 Receive Status Register (RX_STAT)

The receive status register (RX_STAT) is shown in Figure 113 and described in Table 105.

Figure 113. Receive Status Register (RX_STAT)

31	16
Reserved	
R-0x0	

15	8	7	0
Reserved		RX_STAT	
R-0x0		R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 105. Receive Status Register (RX_STAT) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	RX_STAT	0-FFh	Receive masked interrupt status. Each bit in this read only register corresponds to the bit in the receive interrupt that is enabled and generating an interrupt on RX_PULSE.

6.11 Transmit Status Register (TX_STAT)

The transmit status register (TX_STAT) is shown in Figure 114 and described in Table 106.

Figure 114. Transmit Status Register (TX_STAT)

31	16
Reserved	
R-0x0	

15	8	7	0
Reserved		TX_STAT	
R-0x0		R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 106. Transmit Status Register (TX_STAT) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7-0	TX_STAT	0-FFh	Transmit masked interrupt status. Each bit in this read only register corresponds to the bit in the transmit interrupt that is enabled and generating an interrupt on TX_PULSE .

6.12 *Miscellaneous Interrupt Status Register (MISC_STAT)*

The miscellaneous interrupt status register (MISC_STAT) is shown in Figure 115 and described in Table 107.

Figure 115. Miscellaneous Interrupt Status Register (MISC_STAT)

31	16
Reserved	
R-0x0	

15	6	5	4	3	2	1	0
Reserved		STAT_PEND	HOST_PEND	Reserved	MDIO_LINKINT	Reserved	MDIO_USERINT
R-0x0		R/W-0x0	R/W-0x0	R-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 107. Miscellaneous Interrupt Status Register (MISC_STAT) Field Descriptions

Bit	Field	Value	Description
31-6	Reserved	0	Reserved
5	STAT_PEND		Statistics interrupt status
		0	Interrupt has occurred
		1	Interrupt has not occurred
4	HOST_PEND		Host error interrupt status
		0	Interrupt has occurred
		1	Interrupt has not occurred
3	Reserved	0	Reserved (Writes should have a zero)
2	MDIO_LINKINT		MDIO link interrupt status
		0	Interrupt has occurred
		1	Interrupt has not occurred
1	Reserved	0	Reserved (Writes should have a zero)
0	MDIO_USERINT		MDIO user interrupt status
		0	Interrupt has occurred
		1	Interrupt has not occurred

6.13 Receive Interrupts per Millisecond Register (RX_IMAX)

The receive Interrupts per millisecond register (RX_IMAX) is shown in Figure 116 and described in Table 108.

Figure 116. Receive Interrupts per Millisecond Register (RX_IMAX)

31	16
Reserved	
R-0x0	

15	6	5	0
Reserved		RX_IMAX	
R-0x0		R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 108. Receive Interrupts per Millisecond Register (RX_IMAX) Field Descriptions

Bit	Field	Value	Description
31-6	Reserved	0	Reserved.
5-0	RX_IMAX	0-3Fh	Receive interrupts per millisecond. The maximum number of interrupts per millisecond generated on RX_PULSE if pacing is enabled for this interrupt.

6.14 Transmit Interrupts per Millisecond Register (TX_IMAX)

The transmit interrupts per millisecond register (TX_IMAX) is shown in Figure 117 and described in Table 109.

Figure 117. Transmit Interrupts per Millisecond Register (TX_IMAX)

31	16
Reserved	
R-0x0	

15	6	5	0
Reserved		TX_IMAX	
R-0x0		R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 109. Transmit Interrupts per Millisecond Register (TX_IMAX) Field Descriptions

Bit	Field	Value	Description
31-6	Reserved	0	Reserved.
5-0	TX_IMAX	0-3Fh	Transmit interrupts per millisecond. The maximum number of transmit interrupts per millisecond generated on TX_PULSE if pacing is enabled for this interrupt.

7 Serial Gigabit Media Independent Interface (SGMII) Registers

This section describes the memory-mapped registers for the Serial Gigabit Media Independent Interface (SGMII).

Table 110 lists the memory-mapped registers for the SGMII. See the device-specific data manual for the memory address of these registers.

Table 110. Serial Gigabit Media Independent Interface (SGMII) Registers

Offset	Acronym	Register Description	Section
0	IDVER	ID Version Register	Section 7.1
4h	SOFT_RESET	Soft Reset Register	Section 7.2
10h	CONTROL	Control Register	Section 7.3
14h	STATUS	Status Register	Section 7.4
18h	MR_ADV_ABILITY	Advertised Ability Register	Section 7.5
1Ch	MR_NP_TX	Transmit Next Page Register	Section 7.6
20h	MR_LP_ADV_ABILITY	Link Partner Advertised Ability	Section 7.7
24h	MR_NP_RX	Link Partner Next Page Received Register	Section 7.8
40h	DIAG_CLEAR	Diagnostics Clear Register	Section 7.9
44h	DIAG_CONTROL	Diagnostics Control Register	Section 7.10
48h	DIAG_STATUS	Diagnostics Status	Section 7.11

7.1 *Identification and Version Register (IDVER)*

The identification and version register (IDVER) is shown in Figure 118 and described in Table 111.

Figure 118. Identification and Version Register (IDVER)

31 16
TX_IDENT
R-0x002C

15 11	10 8	7 0
RTL_VER	MAJOR_VER	MINOR_VER
R-0x0	R-0x1	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 111. Identification and Version Register (IDVER) Field Descriptions

Bit	Field	Value	Description
31-16	TX_IDENT	0-FFFFh	Transmit identification value.
15-11	RTL_VER	0-1Fh	RTL_version register.
10-8	MAJOR_VER	0-7h	Major version register.
7-0	MINOR_VER	0-FFh	Minor version register.

7.2 Soft Reset Register (SOFT_RESET)

The soft reset register (SOFT_RESET) is shown in Figure 119 and described in Table 112.

Figure 119. Soft Reset Register (SOFT_RESET)

31–16
Reserved
R-0x0

15–2	1	0
Reserved	RT_SOFT_RESET	SOFT_RESET
R-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 112. Soft Reset Register (SOFT_RESET) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved. Read as zero.
1	RT_SOFT_RESET	0-1	Transmit and receive software reset. Writing a one to this bit causes the SGMII transmit and receive logic to be in the reset condition. The reset condition is removed when a zero is written to this bit. This bit is intended to be used when changing between loopback mode and normal mode of operation.
0	SOFT_RESET	0-1	Software reset. Writing a one to this bit causes the SGMII logic to be reset. Software reset occurs immediately. This bit reads as a zero.

7.3 Control Register (CONTROL)

The control register (CONTROL) is shown in Figure 120 and described in Table 113.

Figure 120. Control Register (CONTROL)

31 16
Reserved
R-0x0

15 7	6	5	4	3	2	1	0
Reserved	TEST_ PATTERN_EN	MASTER	LOOPBACK	MR_NP_ LOADED	FAST_LINK_ TIMER	MR_AN_ RESTART	MR_AN_ ENABLE
R-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 113. Control Register (CONTROL) Field Descriptions

Bit	Field	Value	Description
31-7	Reserved	0	Reserved. Read as zero.
6	TEST_PATTERN_EN		Test pattern enable. Force the output of K28.5 for test purposes.
		0	Normal operation.
		1	Forced K28.5 on transmit output.
5	MASTER		Master mode. Set to one for one side of a direct connection. When this bit is set, the control logic uses the MR_ADV_ABILITY register to determine speed and duplexity instead of the MR_LP_ADV_ABILITY register. Master mode allows a SGMII direct connection with auto-negotiation or with a forced link.
		0	Slave mode.
		1	Master mode.
4	LOOPBACK		Loopback mode.
		0	Not in internal loopback mode.
		1	Internal loopback mode. The transmit clock (TX_CLK) is used for transmit and receive.
3	MR_NP_LOADED	0-1	Next page loaded. Writing a one to this bit informs the auto-negotiation process that the next page register has been loaded. This bit is cleared by the auto-negotiation state machine before the MR_PAGE_RX bit in the STATUS register is set. This bit is not used when the SGMII_MODE input is asserted.
2	FAST_LINK_TIMER		Fast link timer.
		0	The link timer value is 10ms in FIBER mode and 1.6ms in SGMII mode.
		1	The link timer value is 2μs in FIBER and SGMII mode. This is included for test purposes.
1	MR_AN_RESTART	0-1	Auto-negotiation restart. Writing a one and then a zero to this bit causes the auto-negotiation process to be restarted.
0	MR_AN_ENABLE	0-1	Auto-negotiation enable. Writing a one to this bit enables the auto-negotiation process.

7.4 Status Register (STATUS)

The status register (STATUS) is shown in Figure 121 and described in Table 114.

Figure 121. Status Register (STATUS)

31-16
Reserved
R-0x0

15-6	5	4	3	2	1	0
Reserved	FIB_SIG_DETECT	LOCK	MR_PAGE_RX	MR_AN_COMPLETE	AN_ERROR	LINK
R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0	R-0x0

LEGEND: R = Read only; -*n* = value after reset

Table 114. Status Register (STATUS) Field Descriptions

Bit	Field	Value	Description
31-6	Reserved	0	Reserved. Read as zero.
5	FIB_SIG_DETECT	0-1	Fiber signal detect. This is the FIB_SIG_DETECT input pin.
4	LOCK	0-1	This is the LOCK input pin.
3	MR_PAGE_RX	0-1	Next page received. This bit is set to one by the auto-negotiation state machine when the next page has been received. This bit is cleared to zero by a host write of a one to the MR_NP_LOADED bit in the CONTROL register. This value is not valid until the lock status bit is asserted.
2	MR_AN_COMPLETE		Auto-negotiation complete. This value is not valid until the lock status bit is asserted.
		0	Auto-negotiation is not complete.
		1	Auto-negotiation is completed.
1	AN_ERROR		Auto-negotiation error. For SGMII mode, an auto-negotiation error occurs when halfduplex gigabit is commanded. For FIBER mode, an auto-negotiation error occurs if both sides cannot be fullduplex. This value is not valid until the LOCK bit in the STATUS register is asserted.
		0	No auto-negotiation error.
		1	Auto-negotiation error.
0	LINK		Link indicator. This value is not valid until the LOCK bit in the STATUS register is asserted.
		0	Link is not up.
		1	Link is up.

7.5 Advertised Ability Register (MR_ADV_ABILITY)

The advertised ability register (MR_ADV_ABILITY) is shown in Figure 122 and described in Table 115.

Figure 122. Advertised Ability Register (MR_ADV_ABILITY)

31	16
Reserved	
R-0x0	

15	0
MR_ADV_ABILITY	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 115. Advertised Ability Register (MR_ADV_ABILITY) Field Descriptions

<table>
<tr><th>Bit</th><th>Field</th><th>Value</th><th colspan="3">Description</th></tr>
<tr><td>31-16</td><td>Reserved</td><td>0</td><td colspan="3">Reserved. Read as zero.</td></tr>
<tr><td rowspan="9">15-0</td><td rowspan="9">MR_ADV_ABILITY</td><td rowspan="9">0-FFFFh</td><td colspan="3">Advertised ability. When in SGMII mode, this corresponds to the value in Serial -GMII specification.</td></tr>
<tr><th>MR_ADV_ABILITY bit</th><th>MAC</th><th>PHY</th></tr>
<tr><td>15</td><td>Link. When set to 1, link is up. When set to 0, link is down.</td><td>0</td></tr>
<tr><td>14</td><td>Auto-negotiation knowledge</td><td>1</td></tr>
<tr><td>13</td><td>0</td><td>0</td></tr>
<tr><td>12</td><td>Duplex mode. When set to 1, full duplex. When set to 0, half duplex.</td><td>0</td></tr>
<tr><td>11-10</td><td>Speed. 10-gig, 01-100 Mbit, 00-10 Mbit</td><td>00</td></tr>
<tr><td>9-1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
</table>

7.6 Transmit Next Page Register (MR_NP_TX)

The transmit next page register (MR_NP_TX) is shown in Figure 123 and described in Table 116.

Figure 123. Transmit Next Page Register (MR_NP_TX)

31	16
Reserved	
R-0x0	

15	0
MR_NP_TX	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 116. Transmit Next Page Register (MR_NP_TX) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-0	MR_NP_TX	0-FFFFh	Next page transmit. This value corresponds to the MR_NP_TX[16:1] value in the IEEE specification. Next page is used only in FIBER mode.

7.7 Link Partner Advertised Ability Register (MR_LP_ADV_ABILITY)

The link partner advertised ability register (MR_LP_ADV_ABILITY) is shown in Figure 124 and described in Table 117.

Figure 124. Link Partner Advertised Ability Register (MR_LP_ADV_ABILITY)

31	16
Reserved	
R-0x0	

15	0
MR_LP_ADV_ABILITY	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 117. Link Partner Advertised Ability Register (MR_LP_ADV_ABILITY) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-0	MR_LP_ADV_ABILITY	0-FFFFh	Link partner advertised ability. Readable when auto-negotiation is complete. When in FIBER mode, this value corresponds to the MR_LP_ADV_ABILITY[16:1] in the IEEE. When in SGMII mode, this value corresponds to the TX_CONFIG_REG[15:0] register value in the Serial-GMII specification.

7.8 Link Partner Next Page Received Register (MR_NP_RX)

The link partner next page received register (MR_NP_RX) is shown in Figure 125 and described in Table 118.

Figure 125. Link Partner Next Page Received Register (MR_NP_RX)

31	16
Reserved	
R-0x0	

15	0
MR_NP_RX	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 118. Link Partner Next Page Received Register (MR_NP_RX) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-0	MR_NP_RX	0-FFFFh	Link partner next page received. Readable when the next page is received. These bits are as defined in the IEEE 802.3 standard. Next page is used only in FIBER mode.

7.9 Diagnostics Clear Register (DIAG_CLEAR)

The diagnostics clear register (DIAG_CLEAR) is shown in Figure 126 and described in Table 119.

Figure 126. Diagnostics Clear Register (DIAG_CLEAR)

31	16
Reserved	
R-0x0	

15 1	0
Reserved	DIAG_CLEAR
R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 119. Diagnostics Clear Register (DIAG_CLEAR) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved. Read as zero.
0	DIAG_CLEAR	0-1	Diagnostics clear. Clears all diagnostic status bits when set to one. Some bits may be set back to one immediately following reset. The reset requires several clocks due to synchronizers.

7.10 Diagnostics Control Register (DIAG_CONTROL)

The diagnostics control register (DIAG_CONTROL) is shown in Figure 127 and described in Table 120.

Figure 127. Diagnostics Control Register (DIAG_CONTROL)

31	16
Reserved	
R-0x0	

15 7	6 4	3 2	1 0
Reserved	DIAG_SM_SEL	Reserved	DIAG_EDGE_SEL
R-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 120. Diagnostics Control Register (DIAG_CONTROL) Field Descriptions

Bit	Field	Value	Description
31-7	Reserved	0	Reserved. Read as zero.
6-4	DIAG_SM_SEL		Diagnostic select. Determines which internal diagnostic bits are muxed onto DIAG_STATUS bit in the diagnostics status register.
		000	Reserved.
		001	Diagnostic hold signals.
		010	Diagnostic sync status (sync_sm state machine).
		011	Diagnostic AN status (amsm state machine).
		100	Diagnostic TXOS status (TXos state machine).
		101	Diagnostic TXCG status (TXcg state machine).
		110	Diagnostic RXSM status (rxsm state machine lower bits).
		111	Diagnostic RXSM status (rxsm state machine upper bits).
3-2	Reserved	0	Reserved. Read as zero.
1-0	DIAG_EDGE_SEL		Diagnostics hold signals edge select.
		00	Diagnostic hold signals level.
		01	Diagnostic hold signals rising edge detected.
		10	Diagnostic hold signals falling edge detected.
		11	Diagnostic hold signals ether (both) edge detected.

7.11 Diagnostics Status Register (DIAG_STATUS)

The diagnostics status register (DIAG_STATUS) is shown in Figure 128 and described in Table 121.

Figure 128. Diagnostics Status Register (DIAG_STATUS)

31	16
Reserved	
R-0x0	

15	0
DIAG_STATUS	
R/W-0x0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 121. Diagnostics Status Register (DIAG_STATUS) Field Descriptions

Bit	Field	Value	Description
31-16	Reserved	0	Reserved. Read as zero.
15-0	DIAG_STATUS	0-FFFFh	Diagnostics status.

8 Management Data Input/Output (MDIO) Registers

This section describes the memory-mapped registers for the Management Data Input/Output (MDIO).

Table 122 lists the memory-mapped registers for the Management Data Input/Output (MDIO). See the device-specific data manual for the memory address of these registers.

Table 122. Management Data Input/Output (MDIO) Registers

Offset	Acronym	Register Description	Section
0	MDIOVER	MDIO Version Register	Section 8.1
4h	MDIOCONTROL	MDIO Control Register	Section 8.2
8h	MDIOALIVE	PHY Alive Status Register	Section 8.3
Ch	MDIOLINK	PHY Link Status Register	Section 8.4
10h	MDIOLINKINTRAW	MDIO Link Status Change Interrupt Register	Section 8.5
14h	MDIOLINKINTMASKED	MDIO Link Status Change Interrupt Register (Masked Value)	Section 8.6
20h	MDIOUSERINTRAW	MDIO User Command Complete Interrupt Register (Raw Value)	Section 8.7
24h	MDIOUSERINTMASKED	MDIO User Command Complete Interrupt Register (Masked Value)	Section 8.8
28h	MDIOUSERINTMASKSET	MDIO User Command Complete Interrupt Mask Set Register	Section 8.9
2Ch	MDIOUSERINTMASKCLR	MDIO User Interrupt Mask Clear Register	Section 8.10
80h	MDIOUSERACCESS0	MDIO User Access Register 0	Section 8.11
84h	MDIOUSERPHYSEL0	MDIO User PHY Select Register 0	Section 8.12
88h	MDIOUSERACCESS1	MDIO User Access Register 1	Section 8.13
8Ch	MDIOUSERPHYSEL1	MDIO User PHY Select Register 1	Section 8.14

8.1 *MDIO Version Register (MDIOVER)*

The MDIO version register (MDIOVER) is shown in Figure 129 and described in Table 123.

Figure 129. MDIO Version Register (MDIOVER)

31 16
MODID
R-0x7

15 8	7 0
REVMAJ	REVMIN
R-0x1	R-0x4

LEGEND: R = Read only; -*n* = value after reset

Table 123. MDIO Version Register (MDIOVER) Field Descriptions

Bit	Field	Value	Description
31-16	MODID	0-FFFFh	Identifies type of peripheral.
15-8	REVMAJ	0-FFh	Management interface module major revision value.
7-0	REVMIN	0-FFh	Management interface module minor revision value.

8.2 MDIO Control Register (MDIOCONTROL)

The MDIO control register (MDIOCONTROL) is shown in Figure 130 and described in Table 124.

Figure 130. MDIO Control Register (MDIOCONTROL)

31	30	29	28 24	23 21	20	19	18	17	16
IDLE	ENABLE	Reserved	HIGHEST_USER_CHANNEL	Reserved	PREAMBLE	FAULT	FAULTENB	INTTESTENB	Reserved
R-0x1	R/W-0x0	R-0x0	R-0x1	R-0x0	R/W-0x0	RWC-0x0	R/W-0x0	R/W-0x0	R-0x0

15 0
CLKDIV
R/W-0x255

LEGEND: R/W = Read/Write; RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 124. MDIO Control Register (MDIOCONTROL) Field Descriptions

Bit	Field	Value	Description
31	IDLE		MDIO state machine IDLE. Set to 1 when the state machine is in the idle state.
		0	State machine is not in idle state.
		1	State machine is in idle state.
30	ENABLE		Enable control. If the MDIO state machine is active at the time it is disabled, it will complete the current operation before halting and setting the idle bit. If using byte access, the enable bit has to be the last bit written in this register.
		0	Disables the MDIO state machine.
		1	Enable the MDIO state machine.
29	Reserved	0	Reserved.
28-24	HIGHEST_USER_CHANNEL	0-1Fh	Highest user channel. This field specifies the highest user access channel that is available in the module and is currently set to 1. This implies that the MDIOUSERACCESS1 register is the highest available user access channel.
23-21	Reserved	0	Reserved.
20	PREAMBLE		Preamble disable.
		0	Standard MDIO preamble is used.
		1	Disables this device from sending MDIO frame preambles.
19	FAULT		Fault indicator. This bit is set to 1 if the MDIO pins fail to read back what the device is driving onto them. This indicates a physical layer fault and the module state machine is reset. Writing a 1 to it clears this bit.
		0	No failure.
		1	Physical layer fault; the MDIO state machine is reset.
18	FAULTENB		Fault detect enable. This bit has to be set to 1 to enable the physical layer fault detection.
		0	Disables the physical layer fault detection.
		1	Enables the physical layer fault detection.
17	INTTESTENB		Interrupt test enable. This bit can be set to 1 to enable the host to set the USERINT and LINKINT bits for test purposes.
		0	Interrupt bits are not set.
		1	Enables the host to set the USERINT and LINKINT bits for test purposes.
16	Reserved	0	Reserved.
15-0	CLKDIV	0-FFFFh	Clock divider. This field specifies the division ratio between CLK and the frequency of MDCLK. MDCLK is disabled when clkdiv is set to 0. MDCLK frequency = clk frequency/(clkdiv+1).

8.3 PHY Acknowledge Status Register (MDIOALIVE)

The PHY acknowledge status register (MDIOALIVE) is shown in Figure 131 and described in Table 125.

Figure 131. PHY Acknowledge Status Register (MDIOALIVE)

31	0
ALIVE	
RWC-0x0	

LEGEND: RWC = Read/Write/Clear

Table 125. PHY Acknowledge Status Register (MDIOALIVE) Field Descriptions

Bit	Field	Value	Description
31-0	ALIVE	0-FFFF FFFFh	MDIO alive. Each of the 32 bits of this register is set if the most recent access to the PHY with address corresponding to the register bit number was acknowledged by the PHY, the bit is reset if the PHY fails to acknowledge the access. Both the user and polling accesses to a PHY will cause the corresponding alive bit to be updated. The alive bits are only meant to be used to give an indication of the presence or not of a PHY with the corresponding address. Writing a 1 to any bit will clear it, writing a 0 has no effect.

8.4 PHY Link Status Register (MDIOLINK)

The PHY link status register (MDIOLINK) is shown in Figure 132 and described in Table 126.

Figure 132. PHY Link Status Register (MDIOLINK)

31	0
LINK	
R-0x0	

LEGEND: R = Read only; -*n* = value after reset

Table 126. PHY Link Status Register (MDIOLINK) Field Descriptions

Bit	Field	Value	Description
31-0	LINK	0-FFFF FFFFh	MDIO link state. This register is updated after a read of the Generic Status Register of a PHY. The bit is set if the PHY with the corresponding address has link and the PHY acknowledges the read transaction. The bit is reset if the PHY indicates it does not have link or fails to acknowledge the read transaction. Writes to the register have no effect. In addition, the status of the two PHYs specified in the MDIOUSERPHYSEL*n* registers can be determined using the MLINK input pins. This is determined by the LINKSEL bit in the MDIOUSERPHYSEL*n* register.

8.5 MDIO Link Status Change Interrupt Register (MDIOLINKINTRAW)

The MDIO link status change interrupt register (MDIOLINKINTRAW) is shown in Figure 133 and described in Table 127.

Figure 133. MDIO Link Status Change Interrupt Register (MDIOLINKINTRAW)

31 – 16
Reserved
R-0x0

15 – 2	1 – 0
Reserved	LINKINTRAW
R-0x0	RWC-0x0

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 127. MDIO Link Status Change Interrupt Register (MDIOLINKINTRAW) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1-0	LINKINTRAW	0-3h	MDIO link change event, raw value. When asserted 1, a bit indicates that there was an MDIO link change event (i.e. change in the MDIOLINK register) corresponding to the PHY address in the MDIOUSERPHYSEL*n* register. LINKINTRAW[0] and LINKINTRAW[1] correspond to MDIOUSERPHYSEL0 and MDIOUSERPHYSEL1, respectively. Writing a 1 will clear the event and writing 0 has no effect. If the INTTESTENB bit in the MDIOCONTROL register is set, the host may set the LINKINTRAW bits to a 1. This mode may be used for test purposes.

8.6 MDIO Link Status Change Interrupt Register (Masked Value) (MDIOLINKINTMASKED)

The MDIO link status change interrupt register (Masked Value) (MDIOLINKINTMASKED) is shown in Figure 134 and described in Table 128.

Figure 134. MDIO Link Status Change Interrupt Register (Masked Value) (MDIOLINKINTMASKED)

31 – 16
Reserved
R-0x0

15 – 2	1 – 0
Reserved	LINKINTMASKED
R-0x0	RWC-0x0

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 128. MDIO Link Status Change Interrupt Register (Masked Value) (MDIOLINKINTMASKED) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1-0	LINKINTMASKED	0-3h	MDIO link change interrupt, masked value. When asserted 1, a bit indicates that there was an MDIO link change event (i.e. change in the MDIO Link register) corresponding to the PHY address in the MDIOUSERPHYSEL*n* register and the corresponding LINKINTENB bit was set. LINKINTMASKED[0] and LINKINTMASKED[1] correspond to MDIOUSERPHYSEL0 and MDIOUSERPHYSEL1, respectively. Writing a 1 will clear the interrupt and writing 0 has no effect. If the INTTESTENB bit in the MDIOCONTROL register is set, the host may set the LINKINT bits to a 1. This mode may be used for test purposes.

8.7 MDIO User Command Complete Interrupt Register (Raw Value) (MDIOUSERINTRAW)

The MDIO user command complete interrupt register (Raw Value) (MDIOUSERINTRAW) is shown in Figure 135 and described in Table 129.

Figure 135. MDIO User Command Complete Interrupt Register (Raw Value) (MDIOUSERINTRAW)

31	16
Reserved	
R-0x0	

15	2	1	0
Reserved		USERINTRAW	
R-0x0		RWC-0x0	

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 129. MDIO User Command Complete Interrupt Register (Raw Value) (MDIOUSERINTRAW) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1-0	USERINTRAW	0-3h	Raw value of MDIO user command complete event for the MDIOUSERACCESS1 register through the MDIOUSERACCESS0 register, respectively. When asserted 1, a bit indicates that the previously scheduled PHY read or write command using that particular MDIOUSERACCESS*n* register has completed. Writing a 1 will clear the event and writing 0 has no effect. If the INTTESTENB bit in the MDIOCONTROL register is set, the host may set the USERINTRAW bits to a 1. This mode may be used for test purposes.

8.8 MDIO User Command Complete Interrupt Register (Masked Value) (MDIOUSERINTMASKED)

The MDIO user command complete interrupt register (Masked Value) (MDIOUSERINTMASKED) is shown in Figure 136 and described in Table 130.

Figure 136. MDIO User Command Complete Interrupt Register (Masked Value) (MDIOUSERINTMASKED)

31	16
Reserved	
R-0x0	

15	2	1	0
Reserved		USERINTMASKED	
R-0x0		RWC-0x0	

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 130. MDIO User Command Complete Interrupt Register (Masked Value) (MDIOUSERINTMASKED) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.

Table 130. MDIO User Command Complete Interrupt Register (Masked Value) (MDIOUSERINTMASKED) Field Descriptions (continued)

Bit	Field	Value	Description
1-0	USERINTMASKED	0-3h	Masked value of MDIO user command complete interrupt for the MDIOUSERACCESS1 register through the MDIOUSERACCESS0 register, respectively. When asserted 1, a bit indicates that the previously scheduled PHY read or write command using that particular MDIOUSERACCESS*n* register has completed and the corresponding USERINTMASKSET bit is set to 1. Writing a 1 will clear the interrupt and writing 0 has no effect. If the INTTESTENB bit in the MDIOCONTROL register is set, the host may set the USERINTMASKED bits to a 1. This mode may be used for test purposes.

8.9 MDIO User Command Complete Interrupt Mask Set Register (MDIOUSERINTMASKSET)

The MDIO user command complete interrupt mask set register (MDIOUSERINTMASKSET) is shown in Figure 137 and described in Table 131.

Figure 137. MDIO User Command Complete Interrupt Mask Set Register (MDIOUSERINTMASKSET)

31 16
Reserved
R-0x0

15 2	1 0
Reserved	USERINTMASKSET
R-0x0	RWC-0x0

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 131. MDIO User Command Complete Interrupt Mask Set Register (MDIOUSERINTMASKSET) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1-0	USERINTMASKSET	0-3h	MDIO user interrupt mask set for USERINTMASKED, respectively. Writing a bit to 1 will enable MDIO user command complete interrupts for that particular MDIOUSERACCESS*n* register. MDIO user interrupt for a particular MDIOUSERACCESS*n* register is disabled if the corresponding bit is 0. Writing a 0 to this register has no effect.

8.10 MDIO User Command Complete Interrupt Mask Clear Register (MDIOUSERINTMASKCLR)

The MDIO user interrupt mask clear register (MDIOUSERINTMASKCLR) is shown in Figure 138 and described in Table 132.

Figure 138. MDIO User Command Complete Interrupt Mask Clear Register (MDIOUSERINTMASKCLR)

31 16
Reserved
R-0x0

15 2	1 0
Reserved	USERINTMASKCLEAR
R-0x0	RWC-0x0

LEGEND: RWC = Read/Write/Clear; R = Read only; -*n* = value after reset

Table 132. MDIO User Command Complete Interrupt Mask Clear Register (MDIOUSERINTMASKCLR) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reserved.
1-0	USERINTMASKCLEAR	0-3h	MDIO user command complete interrupt mask clear for USERINTMASKED, respectively. Writing a bit to 1 will disable further user command complete interrupts for that particular MDIOUSERACCESS*n* register. Writing a 0 to this register has no effect.

8.11 MDIO User Access Register 0 (MDIOUSERACCESS0)

The MDIO user access register 0 (MDIOUSERACCESS0) is shown in Figure 139 and described in Table 133.

Figure 139. MDIO User Access Register 0 (MDIOUSERACCESS0)

31	30	29	28 26	25 21	20 16
GO	WRITE	ACK	Reserved	REGADR	PHYADR
R/W/S-0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0	R/W-0x0

15 0
DATA
R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; S = Status; -*n* = value after reset

Table 133. MDIO User Access Register 0 (MDIOUSERACCESS0) Field Descriptions

Bit	Field	Value	Description
31	GO	0-1	Go. Writing a 1 to this bit causes the MDIO state machine to perform an MDIO access when it is convenient for it to do so, this is not an instantaneous process. Writing a 0 to this bit has no effect. This bit is write able only if the MDIO state machine is enabled. This bit will self clear when the requested access has been completed. Any writes to the MDIOUSERACCESS0 register are blocked when the GO bit is 1. If byte access is being used, the GO bit should be written last.
30	WRITE	0-1	Write enable. Setting this bit to a 1 causes the MDIO transaction to be a register write, otherwise it is a register read.
29	ACK	0-1	Acknowledge. This bit is set if the PHY acknowledged the read transaction.
28-26	Reserved	0	Reserved.
25-21	REGADR	0-1Fh	Register address. Specifies the PHY register to be accessed for this transaction.
20-16	PHYADR	0-1Fh	PHY address. Specifies the PHY to be accesses for this transaction.
15-0	DATA	0-FFFFh	User data. The data value read from or to be written to the specified PHY register.

8.12 MDIO User PHY Select Register 0 (MDIOUSERPHYSEL0)

The MDIO user PHY select register 0 (MDIOUSERPHYSEL0) is shown in Figure 140 and described in Table 134.

Figure 140. MDIO User PHY Select Register 0 (MDIOUSERPHYSEL0)

31 … 16
Reserved
R-0x0

15 … 8	7	6	5	4 … 0
Reserved	LINKSEL	LINKINTENB	Reserved	PHYADDRMON
R-0x0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 134. MDIO User PHY Select Register 0 (MDIOUSERPHYSEL0) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7	LINKSEL	0-1	Link status determination select. Set to 1 to determine link status using the MLINK pin. Default value is 0 which implies that the link status is determined by the MDIO state machine.
6	LINKINTENB		Link change interrupt enable. Set to 1 to enable link change status interrupts for PHY address specified in PHYADDRMON. Link change interrupts are disabled if this bit is set to 0.
		0	Link change interrupts are disabled.
		1	Link change status interrupts for PHY address specified in PHYADDRMON bits are enabled.
5	Reserved	0	Reserved.
4-0	PHYADDRMON	0-1Fh	PHY address whose link status is to be monitored.

8.13 MDIO User Access Register 1 (MDIOUSERACCESS1)

The MDIO user access register 1 (MDIOUSERACCESS1) is shown in Figure 141 and described in Table 135.

Figure 141. MDIO User Access Register 1 (MDIOUSERACCESS1)

31	30	29	28 26	25 21	20 16
GO	WRITE	ACK	Reserved	REGADR	PHYADR
R/W/S-0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0	R/W-0x0

15 0
DATA
R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; S = Status; -*n* = value after reset

Table 135. MDIO User Access Register 1 (MDIOUSERACCESS1) Field Descriptions

Bit	Field	Value	Description
31	GO	0-1	Writing a 1 to this bit causes the MDIO state machine to perform an MDIO access when it is convenient for it to do so, this is not an instantaneous process. Writing a 0 to this bit has no effect. This bit is write able only if the MDIO state machine is enabled. This bit will self clear when the requested access has been completed. Any writes to the MDIOUSERACCESS0 register are blocked when the GO bit is 1. If byte access is being used, the GO bit should be written last.
30	WRITE	0-1	Write enable. Setting this bit to a 1 causes the MDIO transaction to be a register write, otherwise it is a register read.
29	ACK	0-1	Acknowledge. This bit is set if the PHY acknowledged the read transaction.
28-26	Reserved	0	Reserved.
25-21	REGADR	0-1Fh	Register address; specifies the PHY register to be accessed for this transaction.
20-16	PHYADR	0-1Fh	PHY address; specifies the PHY to be accesses for this transaction.
15-0	DATA	0-FFFFh	User data. The data value read from or to be written to the specified PHY register.

8.14 MDIO User PHY Select Register 1 (MDIOUSERPHYSEL1)

The MDIO user PHY select register 1 (MDIOUSERPHYSEL1) is shown in Figure 142 and described in Table 136.

Figure 142. MDIO User PHY Select Register 1 (MDIOUSERPHYSEL1)

31	16
Reserved	
R-0x0	

15 8	7	6	5	4 0
Reserved	LINKSEL	LINKINTENB	Reserved	PHYADDRMON
R-0x0	R/W-0x0	R/W-0x0	R-0x0	R/W-0x0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 136. MDIO User PHY Select Register 1 (MDIOUSERPHYSEL1) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved.
7	LINKSEL	0-1	Link status determination select. Set to 1 to determine link status using the MLINK pin. Default value is 0 which implies that the link status is determined by the MDIO state machine.
6	LINKINTENB		Link change interrupt enable. Set to 1 to enable link change status interrupts for PHY address specified in PHYADDRMON. Link change interrupts are disabled if this bit is cleared to 0.
		0	Link change interrupts are disabled
		1	Link change status interrupts for PHY address specified in PHYADDRMON bits are enabled.
5	Reserved	0	Reserved.
4-0	PHYADDRMON	0-1Fh	PHY address whose link status is to be monitored.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Broadband	www.ti.com/broadband
DSP	dsp.ti.com	Digital Control	www.ti.com/digitalcontrol
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Military	www.ti.com/military
Logic	logic.ti.com	Optical Networking	www.ti.com/opticalnetwork
Power Mgmt	power.ti.com	Security	www.ti.com/security
Microcontrollers	microcontroller.ti.com	Telephony	www.ti.com/telephony
RFID	www.ti-rfid.com	Video & Imaging	www.ti.com/video
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265