gridconnect

DSTni-EX User Guide

Section Four

Part Number 900-335
Revision A 3/04

Copyright & Trademark

Lantronix and the Lantronix logo, and combinations thereof are registered trademarks of Lantronix, Inc. DSTni is a registered trademark of Lantronix, Inc. Ethernet is a registered trademark of Xerox Corporation. All other product names, company names, logos or other designations mentioned herein are trademarks of their respective owners.

- Am186 is a trademark of Advanced Micro Devices, Inc.
- Ethernet is a registered trademark of Xerox Corporation.
- SPI is a trademark of Motorola, Inc.

Lantronix
15353 Barranca Parkway
Irvine, CA 92618, USA
Phone: 949-453-3990
Fax: 949-453-3995

Technical Support
Phone: 630-245-1445
Fax: 630-245-1717

Master Distributor
Grid Connect
1841 Centre Point Circle, Suite 143
Naperville, IL 60563
Phone: 630-245-1445
www.gridconnect.com

Am186 is a trademark of Advanced Micro Devices, Inc.
Ethernet is a registered trademark of Xerox Corporation.
SPI is a trademark of Motorola, Inc.

REV	Changes	Released Date
A	Reformat. Add changes from Design Spec. 1.1	3-24-04

Warranty

Lantronix warrants each Lantronix product to be free from defects in material and workmanship for a period specified on the product warranty registration card after the date of shipment. During this period, if a customer is unable to resolve a product problem with Lantronix Technical Support, a Return Material Authorization (RMA) will be issued. Following receipt of an RMA number, the customer shall return the product to Lantronix, freight prepaid. Upon verification of warranty, Lantronix will -- at its option -- repair or replace the product and return it to the customer freight prepaid. If the product is not under warranty, the customer may have Lantronix repair the unit on a fee basis or return it. No services are handled at the customer's site under this warranty. This warranty is voided if the customer uses the product in an unauthorized or improper way, or in an environment for which it was not designed.

Lantronix warrants the media containing its software product to be free from defects and warrants that the software will operate substantially according to Lantronix specifications for a period of **60 DAYS** after the date of shipment. The customer will ship defective media to Lantronix. Lantronix will ship the replacement media to the customer.

* * * *

In no event will Lantronix be responsible to the user in contract, in tort (including negligence), strict liability or otherwise for any special, indirect, incidental or consequential damage or loss of equipment, plant or power system, cost of capital, loss of profits or revenues, cost of replacement power, additional expenses in the use of existing software, hardware, equipment or facilities, or claims against the user by its employees or customers resulting from the use of the information, recommendations, descriptions and safety notations supplied by Lantronix. Lantronix liability is limited (at its election) to:

refund of buyer's purchase price for such affected products (without interest)

repair or replacement of such products, provided that the buyer follows the above procedures.

There are no understandings, agreements, representations or warranties, express or implied, including warranties of merchantability or fitness for a particular purpose, other than those specifically set out above or by any existing contract between the parties. Any such contract states the entire obligation of Lantronix. The contents of this document shall not become part of or modify any prior or existing agreement, commitment or relationship.

For details on the Lantronix warranty replacement policy, go to our web site at http://www.lantronix.com/support/warranty/index.html

Contents

List of Figures

List of Tables

1: About This User Guide

This User Guide describes the technical features and programming interfaces of the Lantronix DSTni-EX chip (hereafter referred to as “DSTni”).

DSTni is an Application Specific Integrated Circuit (ASIC)-based single-chip solution (SCS) that integrates the leading-edge functionalities needed to develop low-cost, high-performance device server products. On a single chip, the DSTni integrates an x186 microprocessor, 16K-byte ROM, 256K-byte SRAM, programmable input/output (I/O), and serial, Ethernet, and Universal Serial Bus (USB) connectivity — key ingredients for device- server solutions. Although DSTni embeds multiple functions onto a single chip, it can be easily customized, based on the comprehensive feature set designed into the chip.

Providing a complete device server solution on a single chip enables system designers to build affordable, full-function solutions that provide the highest level of performance in both processing power and peripheral systems, while reducing the number of total system components. The advantages gained from this synergy include:

- Simplifying system design and increased reliability.
- Minimizing marketing and administration costs by eliminating the need to source products from multiple vendors.
- Eliminating the compatibility and reliability problems that occur when combining separate subsystems.
- Dramatically reducing implementation costs.
- Increasing performance and functionality, while maintaining quality and cost effectiveness.
- Streamlining development by reducing programming effort and debugging time.
- Enabling solution providers to bring their products to market faster.

These advantages make DSTni the ideal solution for designs requiring x86 compatibility; increased performance; serial, programmable I/O, Ethernet, and USB communications; and a glueless bus interface.

Intended Audience

This User Guide is intended for use by hardware and software engineers, programmers, and designers who understand the basic operating principles of microprocessors and their systems and are considering designing systems that utilize DSTni.

Conventions

This User Guide uses the following conventions to alert you to information of special interest.

The symbols # and n are used throughout this Guide to denote active LOW signals.

Notes: *Notes are information requiring attention.*

Navigating Online

The electronic Portable Document Format (PDF) version of this User Guide contains *hyperlinks*. Clicking one of these hyper links moves you to that location in this User Guide. The PDF file was created with Bookmarks and active links for the Table of Contents, Tables, Figures and cross-references.

Organization

This User Guide contains information essential for system architects and design engineers. The information in this User Guide is organized into the following chapters and appendixes.

- *Section 1: Introduction*
  Describes the DSTni architecture, design benefits, theory of operations, ball assignments, packaging, and electrical specifications. This chapter includes a DSTni block diagram.
- *Section 2: Microprocessor*
  Describes the DSTni microprocessor and its control registers.
- *Section 2: SDRAM*
  Describes the DSTni SDRAM and the registers associated with it.
- *Section 3: Serial Ports*
  Describes the DSTni serial ports and the registers associated with them.
- *Section 3: Programmable Input/Output*
  Describes DSTni's Programmable Input/ Output (PIO) functions and the registers associated with them.
- *Section 3: Timers*
  Describes the DSTni timers.
- *Section 4: Ethernet Controllers*
  Describes the DSTni Ethernet controllers.
- *Section 4: Ethernet PHY*
  Describes the DSTni Ethernet physical layer core.
- *Section 5: SPI Controller*
  Describes the DSTni Serial Peripheral Interface (SPI) controller.
- *Section 5: I2C Controller*
  Describes the DSTni I^2C controller.
- *Section 5: USB Controller*
  Describes the DSTni USB controller.
- *Section 5: CAN Controllers*
  Describes the DSTni Controller Area Network (CAN) bus controllers.
- *Section 6: Interrupt Controller*
  Describes the DSTni interrupt controller.
- *Section 6: Miscellaneous Registers*
  Describes DSTni registers not covered in other chapters of this Guide.
- *Section 6: Debugging In-circuit Emulator (Delce)*
- *Section 6: Packaging and Electrical*
  Describes DSTni's packaging and electrical characteristics.
- *Section 6: Applications*
  Describes DSTni's packaging and electrical characteristics.
- *Section 6: Instruction Clocks*
  Describes the DSTni instruction clocks.
- *Section 6: DSTni Sample Code*
- *Section 6: Baud Rate Calculations*
  Provides baud rate calculation tables.

2: Ethernet Controllers

This chapter describes the DSTni Ethernet controllers. Topics include:

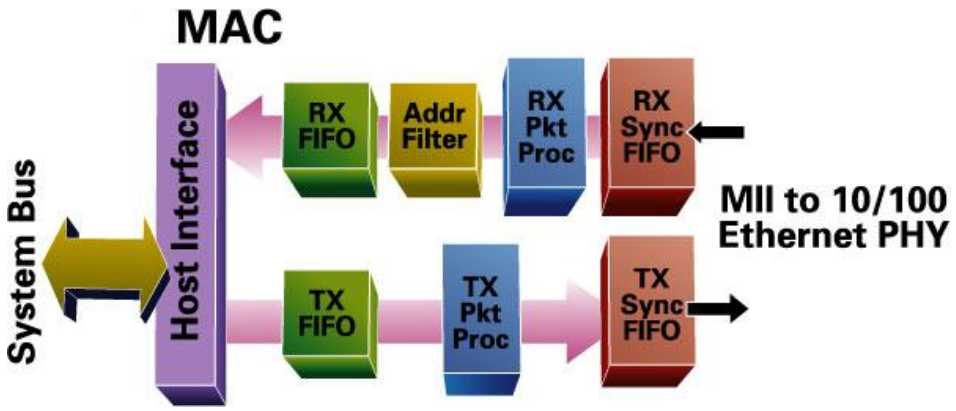
Features

- Connects directly to an external PHY through a MII
- Supports IEEE 802.3, 802.3u, and 802.3af Ethernet standards
- Supports Bus Master mode
- Software compatible with AMD's Am79C960 Ethernet controller

Block Diagram

The Ethernet controller consists of six blocks. Figure 2-1 shows a block diagram of the Ethernet controller.

Figure 2-1. Ethernet Controller Block Diagram



Functional Description

Overview

DSTni provides two Ethernet controllers.

- The first Ethernet controller (MAC0) is located at base I/O address 9000h. It is connected to interrupt 0 (INT0) and has the internal PHY connected to it
- The second Ethernet controller (MAC1) is located at base I/O address 9100h. It is connected to interrupt 1 and is shared with external interrupt 1. This Ethernet controller connects to an external MII port.

Both Ethernet controllers support Bus Master mode. In Bus Master mode, all transfers are performed using integrated DMA controllers. This configuration enhances system performance by allowing the Ethernet controllers to bypass the platform DMA controllers and address the full 24-bit memory space directly. Implementing Bus Master mode also allows minimum parts count for the majority of applications.

The Ethernet controller is organized into seven interconnecting blocks Figure 5-1 shows a block diagram of the Ethernet controller, with the flow of data and control signals between these blocks.

There are three separate clock domains within the Ethernet controller. All signals that cross clock domains, from CLK_RX to the microprocessor clock signal CLK, are synchronized in the RX_RATE block.

The TX_RATE block synchronizes signals from CLK domain to MII CLK_TX domain. In the same block, signals such as CRS and COL get synchronized to CLK and/or CLK_TX. Carrier sense and collision are considered asynchronous to any clock.

All other logic operates on the main clock signal CLK.

Bus Interface

The Ethernet controllers provide the logic for interfacing to the microprocessor bus. The bus is optimized for synchronous, single-cycle synchronous RAMs and allows multiple masters.

The Ethernet controller core provides separate data paths for slave and master functions. As a slave, the Ethernet controller responds only to read/write cycles in I/O space and decodes addresses inside a 256-byte block. Only the Ethernet controller Control registers are accessed through the slave interface; as a result, the “slave portion” of the bus interface is integrated in register block, REGS.

The Ethernet controller slave never inserts wait states. Therefore a write cycle always takes 1 clock and “read data” is valid on the next clock after the Ethernet controller receives an IO read command. The Ethernet controller core initiates all transmit/receive data transfers to and from the core.

As a master, the Ethernet controllers generate read/write cycles in memory space only

The core has a 16-bit data interface to the bus. The Ethernet controller master accesses the bus in bursts from 1 to 7 words. A typical memory write cycle is 8-to-12 bytes (4-to-6 words) long. A typical memory read cycle is 8 bytes (4 words) long. All bursts are scheduled automatically, obviating the need for programmable registers that affect bus cycles. DMA access is not necessarily sequential (for example, when updating descriptors).

Bus Master Mode

There are three types of DMA cycles when the Ethernet controller is in Bus Master mode:

- **Initialization DMA Transfers:** After the INIT bit is set to 1, the Ethernet controllers perform two DMA bursts of four words each — both of which are sequential read cycles. This operation fetches the 8 words of the initialization block.
- **Descriptor DMA** Transfers: When the Ethernet controllers need to access a descriptor, such as when a poll or writing the status bits, they perform a number of read or write cycles within one burst. Any single burst consists of only read or write cycles. The bus is released between each burst. These accesses typically are not sequential.
- **Data DMA Transfers:** The Ethernet controllers perform bursts of sequential DMA reads or writes as necessary to fill or empty the FIFO data.

Transmit/Receive FIFOs

The Ethernet controller core has relatively small transmit/receive FIFOs. This approach minimizes the size of the core in “system-on-chip” applications.

When chained buffers are used in half-duplex mode, the size of the first transmit buffer in a chain should be at least 80 bytes and the size of the first receive buffer should be at least 64 bytes. Otherwise, the buffers have no size restrictions during full-duplex mode or during times when chained buffers are not used. However, in full-duplex mode, very small chained buffers can decrease system performance. Therefore, we recommend that you do not use chained buffers when running 100 Mbs mode in full- and half-duplex.

Buffer Management

The BMUSM state machine performs all buffer management functions. This block works with the REGS and DSC_CTRS blocks, which holds the CSR and DMA address registers. The same state machine schedules the initialization sequence. During this sequence, the Ethernet controller DMA cycles update the selected registers.

Bus FIFOs

In addition to the transmit/receive FIFOs, the Ethernet controller has two separate “bus FIFOs” in the RX_BUSFF (receive) and TX_BUSFF (transmit) blocks.

Each bus FIFO is 4 bits long x 16 bits wide. These FIFOs facilitate 16-bit-wide burst-type access to the bus. They align data on uneven byte boundaries and decouple 16-bit data from the 8-bit-wide data path in the remaining blocks of the Ethernet controller.

Register Block

The register block holds all control CSR registers. All control registers are accessible in IO address space. Selected registers can be updated by DMA transfers during initialization sequence. For information about the registers in the register block, see

Ethernet Controller Register Definitions on page 33.

This block also contains a simple interface to the MII management signals:

- Management data
- Input/output
- Management clock

MII management signals are fully under software control.

To poll or write registers of an external PHY, toggle the bits in the MII control register. The MI_MDC signal is the MII management clock. It is an aperiodic signal that has no maximum HIGH or LOW times. Given its behavior, it can easily be emulated in software.

Descriptor Block

The DSC_CTRS block stores local copies of the current field for each descriptor's ring. One set of descriptors for RX ring and one set for TX ring can be stored at the same time. This block also contains the DMA address counters.

Transmit FIFO

The transmit FIFO and its controller are integrated in the TX_FIFO block. The FIFO is built of flip-flops, and is 32 bytes deep.

The FIFO is synchronous and operates in the CLK clock domain only. The main signals to the system control block are:

- An 8-bit signal for data/status
- A 1-bit tag signal
- A 1-bit signal indicating that the FIFO is not full and can accept data

The data bus carries either transmitted data or status signals, depending on the state of the tag signal. A similar interface is provided to the TX_MAC block (8 bits of data/status, 1 tag, and data request).

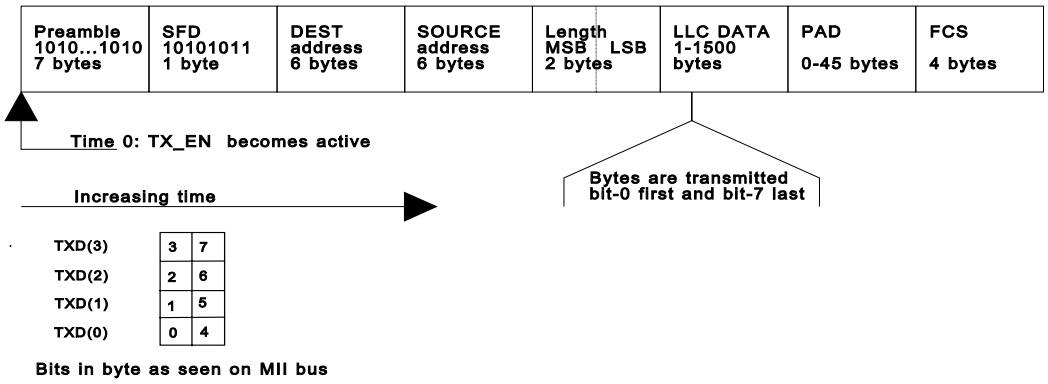
The transmit FIFO does not pass a “start-of-frame status” to the TX_MAC until the FIFO is full. From this point on, the TX_MAC block requests data as needed; the TX_FIFO, in return, requests consecutive transfers from the system control until the end of frame is indicated on the data/status bus. In case of a collision, the TX_FIFO tries to recover its data. If this is not possible, it signals the system controller that the currently transmitted frame must restart.

Transmit Media Access Controller

The Transmit Media Access Controller provides the logic for meeting the IEEE 802.3 Ethernet LAN standard for frame transmission. When a start of frame is detected on data/status bus, the TX_MAC block transmits preamble data. TX_MAC delays the transmission if the receive path is active and ensures that the interframe period is met. In full-duplex mode, only inter frame period is observed, without a deferring process. The host builds the DEST, SOURCE, LENGTH, and LLC data fields in the buffer memory. Figure 5-3 and Figure 5-4 show the fields in the IEEE 802.3 frames.

The Ethernet controller can optionally append the required PAD bytes (zeros) if the LLC data is less than 46 bytes (see CSR4: Features Control Register on page 37). The Ethernet controller always computes the Frame Check Sequence (FCS or CRC) field, and can optionally append it automatically on a frame-by-frame basis (see TMD1: Status on page 20 and CSR15: Mode Register on page 43).

Figure 2-2. IEEE 802.3 Transmit Frame Format



Instead of using the value in the length field, the TX_MAC detects the “end-of-frame status” passed by TX_FIFO. This status is computed based on the current value of the TMD2 descriptor and the ENP bit in the TMD1 descriptor. The TX_MAC formats the transmit data into nibbles and generates the transmit-enable signal.

TX_MAC runs exclusively in the microprocessor clock domain, CLK. Data and transmit-enable signals are sent to the TX_RATE block to be converted to the CLK_TX domain. Due to 802.3 timing requirements, the TX_MAC block does not generate the jam sequence.

The jam sequence gets enabled by the TX_RATE block in CLK_TX domain, to meet fast response to MII's COL signal. If TX_RATE detects a collision then it signals this event to TX_MAC, and after the jam is completed, the TX_MAC block enables its back-off counters.

The backoff counter is loaded with a random value. The retry counter, which is actually a shift register, masks selected bits of this random number and increases its range with each retry. This results in the proper IEEE 802.3 backoff interval equation of:

$0 \leq r \leq 2^{**}k$

where:

- k = min(n,10)
- n = the number of retries
- r = the random number loaded into the backoff counter

The backoff counter then counts the appropriate number of slot times (512 bit times) before allowing the TX_MAC to retry the transmission.

If a collision occurs after 512 bit times, the TX_MAC ends the frame. The frame is not retried, but will have the LCOL bit set in the descriptor. The upper-level software is responsible for retrying the frame.

Receive Media Access Controller

The Receive Media Access Controller provides the logic for meeting the IEEE 802.3 Ethernet LAN standard for frame reception. The Receive MAC is divided into two functional blocks, RX_MAC and RX_ADDRFLT, which operate exclusively on the CLK clock signal.

RX_MAC Block

RX_MAC:

- Detects the SFD pattern.
- Verifies the FCS field.
- Senses framing errors (odd number of nibbles).
- Monitors the RX_ER signal, which indicates errors received from an external PHY.
- Passes received data to the address filter block, RX_ADDRFLT, through a 1-byte wide bus. This bus carries either the data or status information about the received frame.

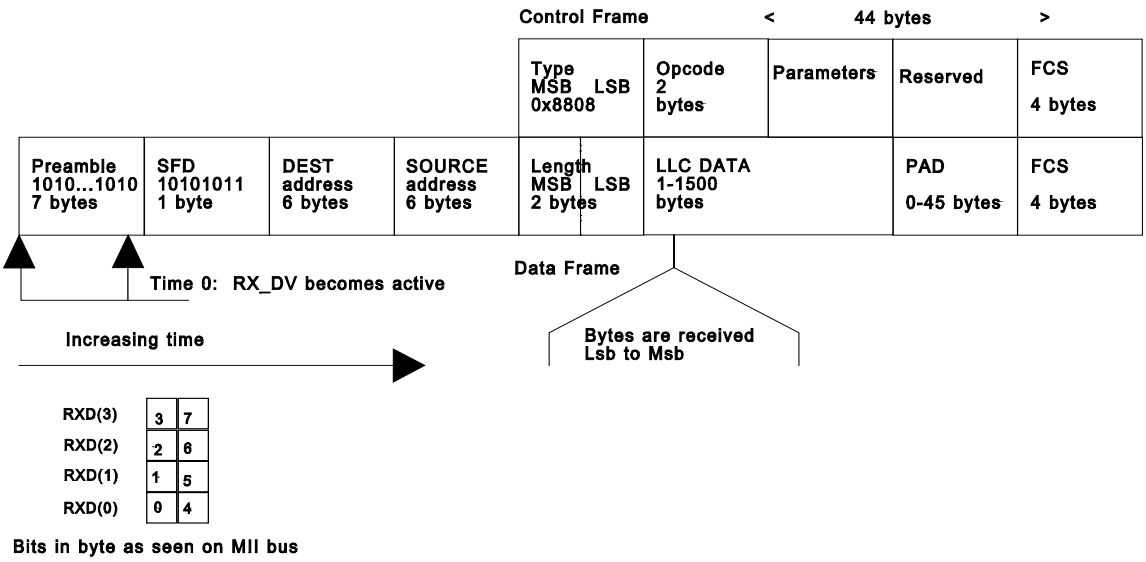
RX_ADDRFLT Block

RX_ADDRFLT determines whether there is a match for the destination address under the currently active addressing mode. While the Destination Address field is being verified, the incoming bytes are stored in a 10-byte FIFO.

- If the address does not match, RX_ADDRFLT resets its FIFO and disregards incoming data until the start of the next packet, as indicated by RX_MAC on the data/status bus.
- Once the address matches, the RX_ADDRFLT continues to send data to the SYSCNTRL block until all bytes are received.

All packets regardless of length (longer then 1518 bytes or shorter then 64 bytes) are sent to SYSCNTRL. Packets less than 64 bytes are considered to be fragments resulting from a collision and are automatically discarded from memory by the Buffer Management Unit. Short packets can be received if the Runt Packet Accept bit (RPA of CSR4) is set (see CSR4: Features Control Register on page 37).

Figure 2-3. IEEE802.3 Receive Frame Format



RX_ADDRFLT also checks whether an incoming frame matches a MAC Control frame for Pause Operation. A frame is accepted as a Control frame if no errors are detected during reception and:

- The destination address matches a multicasts address 0x01_80_C2_00_00_01 or a physical address set in the PADDR register,
- Length/Type field contains value 0x88_08,
- Opcode field contains value 0x00_01 (Pause opcode).

When a Pause Frame is received, RX_ADDRFLT notifies the BMUSM block and passes the pause value from the first two bytes of the MAC Control Parameters field. When a currently transmitted frame completes, any following transfers wait until the pause timer expires. The Control Parameter Field value is a 16-bit unsigned integer that indicates the length of a requested pause, in multiples of 512-bit times. Like other data frames, control frames are stored in RX memory buffers and discarded by software. A Pause Frame generates an interrupt if mask pause interrupt is not set.

Physical Layer Device

The Ethernet controller can connect directly to a Physical Layer Device (PHY) equipped with a MII. The Ethernet controller does not provide any functionality below the Reconciliation Sublayer, as described in the IEEE 802.3-1998 specification. The Ethernet controller works in half- or full-duplex mode. The transfer rate is completely transparent to the Ethernet controller, as long as selected CLK clock frequency and bus activities do not limit core throughput. Full-duplex mode (bit FDEN in MIIP register) should be enabled after successful negotiation of modes with PHY.

Theory of Operation

Software Interface

The Ethernet controller relies on descriptor rings built in memory, and modified by both the host software and the Ethernet controller. The Ethernet controller uses DMA transfers to access memory while minimizing impact on overall system throughput.

Initialization

At power-up or following an Ethernet controller reset, the Ethernet controllers must be initialized. Upon initialization, some Ethernet controller registers get set to default values, while others clear and must be initialized by software. Table 2-1 shows the initialization block memory configuration.

Note: *The initialization block must start on an 8-byte boundary.*

Table 2-1. Initialization Block Format

Address	Bits [15:12]	Bits [11:8]	Bits [7:4[	Bits [3:0]
IADR+22	TLEN (CSR78)	0000	TDRA[23:16] (CSR31)	
IADR+20	TDRA[15:0] (CSR30)			
IADR+18	RLEN (CSR76)	0000	RDRA[23:16] (CSR25)	
IADR+16	RDRA[15:0] (CSR24)			
IADR+14	Reserved			
IADR+12	Reserved			
IADR+10	Reserved			
IADR+8	Reserved			
IADR+6	PADR[47:32] (CSR14)			
IADR+4	PADR[31:16] (CSR13)			
IADR+2	PADR[15:0] (CSR12)			
IADR+0	MODE (CSR15)			

RLEN and TLEN

The RLEN and TLEN fields shown in the initialization block specify the number of buffer descriptors in each ring buffer. Table 5-2 shows the corresponding number of buffers for the RLEN/TLEN values. The RLEN/TLEN values are programmed into CSR76 and CSR78 (see CSR76: Receive Ring Length Register on page 47 and CSR78: Transmit Ring Length Register on page 48).

Values other than those listed in Table 2-2 can be programmed by writing to the CSR register bits [76 and CSR78 directly. The RLEN and TLEN fields must be programmed in bits [15:13]. Bit [12] should always be zero.

Table 2-2. RLEN and TLEN Settings

RLEN/TLEN	Buffers
000	1
001	2
010	4
011	8
100	16
101	32
110	64
111	128

RDRA and TDRA

The RDRA and TDRA fields are the base address of the RX and TX buffer rings. For more information, see:

- Register CSR24 (page 45)
- Register CSR25 (page 45)
- Register CSR30 (page 46)
- Register CSR31 (page 46)

Descriptor Rings

Data is sent and received via buffers that are built into memory. These buffers are pointed to by buffer descriptor rings and registers within the Ethernet controller. There are two descriptor rings: a TX ring and an RX ring. Each descriptor entry is 4 words (8 bytes) and must align on 8-byte boundaries.

Each descriptor ring must occupy a contiguous area of memory. Each descriptor comprises 4 words, with various fields in each word. The descriptor contains:

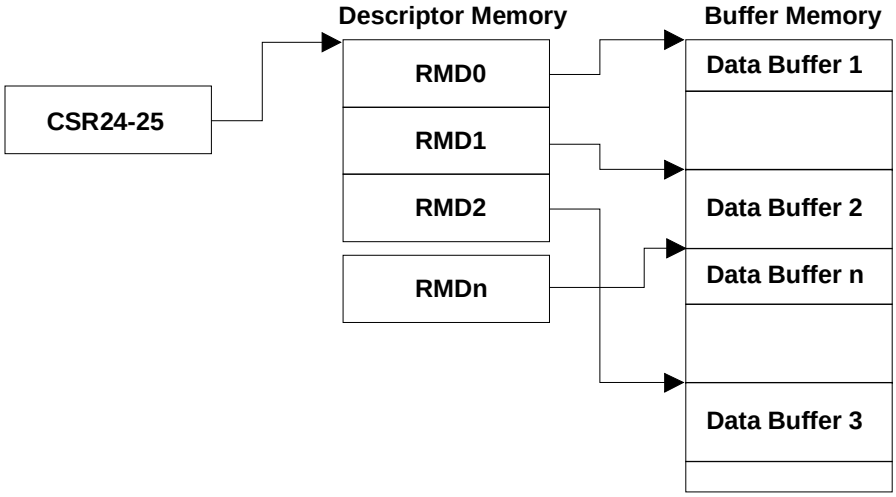
- The address of the start of the buffer
- The number of bytes in the buffer
- Buffer status information, such as whether the buffer is full or empty

Figure 2-4 shows how the RX descriptor rings are built into memory and how they point to data buffers in another part of memory. There are two of descriptor rings, one for TX and one for RX. Figure 2-4 only shows the registers used for the RX descriptor ring. The TX ring is the same except, but uses a different set of registers. CSR24 and CSR25 form a 24-bit address that points to the start of the RX descriptor ring. An internal register in the Ethernet controller points to the current buffer in use. CSR76 contains the number of buffers in the RX descriptor ring (in this example only four are shown). The descriptor entries, in turn, contain a 24-bit address that points to the start of the buffer where data is to be placed.

Note: *Buffers can be placed anywhere in memory and in any order.*

After initialization, the Ethernet controller loads its internal current descriptor register with the descriptor base register. It then DMAs the descriptor to determine the state of the buffer. If the buffer is “owned” by the Ethernet controller, the buffer is used when needed.

Figure 2-4. Descriptor Rings



Transmit Descriptors

The Transmit Descriptor entries comprise 4 words (8 bytes). Each descriptor must be on an 8-byte boundary.

TMD0: ADR[15:0]

Table 2-3. TMD0: ADR[15:0]

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	ADR[15:0]															

Table 2-4. TMD0: ADR[15:0] Definitions

Bits	Field Name	Description
15:0	ADR[15:0]	**TMD0** This field makes up the low 16 bits of the starting address of the transmit data buffer. The buffer can be placed on any byte boundary.

TMD1: Status

Table 2-5. TMD1: Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	OWN	ERR	ADD_FCS	MORE	ONE	DEF	STP	ENP	ADS [23:16]							

Table 2-6. TMD1: Status Definitions

Bits	Field Name	Description
15	OWN	**Ethernet Controller Ownership** 1 = buffer is "owned" or controlled by the Ethernet controller. 0 = buffer is owned by the host. The Ethernet Controller clears this bit after transmitting all data from the buffer.
14	ERR	**Logical OR of UFLO, LCOL, LCAR, and RTRY** 1 = error. 0 = no error. This bit is written by the Ethernet controller.
13	ADD_FCS	**FCS Generation** Dynamically controls the generation of FCS on a frame-by-frame basis. 1 = causes the Ethernet controller to append the FCS sequence to the end of a frame, regardless of the state of the DTXFCS bit. 0 = does not append the FCS sequence to frame. ADD_FCS is valid only when the STP bit is set. It is unchanged by the Ethernet controller.
12	MORE	**More Than One Retry** 1 = more than one retry was needed to transmit this frame. 0 = no retries were needed to transmit a frame. This bit is valid only when the ENP bit is set. It is written by the Ethernet controller.
11	ONE	**One Retry** 1 = exactly one retry was needed to transmit this frame. 0 = was not one retry. ONE is valid only when the ENP bit is set. It is written by the Ethernet controller.
10	DEF	**Defer Transmission** 1 = Ethernet controller deferred transmitting this frame because the media was busy. 0 = Ethernet controller did not defer transmitting this frame because the media was busy. Valid only when the ENP or ERR bit is set. It is written by the Ethernet controller.
9	STP	**First Buffer Used** 1 = the first buffer used for this frame. 0 = the first buffer was not used for this frame. It is used for chaining buffers. The STP bit must be set for the first buffer in a chain or the buffer will be skipped. This bit is unchanged by the Ethernet controller.
8	ENP	**Last Buffer Used** 1 = this is the last buffer used for this frame. 0 = this is not the last buffer used for this frame. If both the STP and ENP bits are set in the same descriptor, the frame fits in a single buffer, and no data chaining is to be used. This bit is unchanged by the Ethernet controller.
7:0	ADS [23:16]	**High-order 8 Bits of the Buffer Starting Address** These bits are unchanged by the Ethernet controller.

TMD2: Buffer Byte Count

Table 2-7. TMD2: Buffer Byte Count

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RES				BCNT											

Table 2-8. TMD2: Buffer Byte Count

Bits	Field Name	Description
15:12	RES	**Must be all ones**
11:0	BCNT	**Buffer Byte Length** This field is the length of this buffer, in bytes. It is expressed as the two's complement of the length. There is no minimum buffer size restriction. Zero length buffers are allowed. These bits are unchanged by the Ethernet controller.

TMD3: Error Status

Table 2-9. TMD3: Error Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	BUFF	UFLO	///	LCOL	LCAR	RTRY	///									

Table 2-10. TMD3: Error Status Definitions

Bits	Field Name	Description
15	BUFF	**No Current Chained Frame Transmitting** 1 = Ethernet controller finds OWN=1 and STP=0, and is not currently transmitting a chained frame. 0 = Ethernet controller is currently transmitting a chained frame. This bit is written by the Ethernet controller.
14	UFLO	**FIFO Underflow** 1 = a FIFO underflow has occurred. The frame will have a bad FCS appended to it and will be dropped. To enforce visibility of such error conditions on MII bus, the Ethernet controller generates TX_ER before deasserting TX_EN. This bit is written by the Ethernet controller. 0 = a FIFO underflow has not occurred.
13	///	**Reserved** This bit is always 0.
12	LCOL	**Late Collision** 1 = a late collision occurred. The frame is dropped. 0 = a late collision has not occurred. This bit is written by the Ethernet controller.
11	LCAR	**Carrier Sense Signal CRS)** 1 = CRS signal went inactive during transmission. 0 = CRS signal did not go inactive during transmission. This bit is written by the Ethernet controller.

Bits	Field Name	Description
10	RTRY	**Retry** 1 = the transmitter tried to transmit the frame up to 16 times, but failed. This indicates a heavily loaded network with excessive traffic and collisions. If DRTY=1, RTRY is set if the first transmission attempt fails. This bit is written by the Ethernet controller. 0 = the transmitter did not fail to transmit a frame.
9:0	///	**Reserved**

Receive Descriptors

The Receive Descriptor entries comprise 4 words (8 bytes). Each descriptor must be on an 8-byte boundary.

RMD0: ADR[15:0]

Table 2-11. RMD0: ADR[15:0]

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	ADR[15:0]															

Table 2-12. RMD0: ADR[15:0] Definitions

Bits	Field Name	Description
15:0	ADR[15:0]	**RMD0** This field makes up the low 16 bits of the starting address of the receive data buffer. The buffer can be placed on any byte boundary.

RMD1: Status

Table 2-13. RMD1: Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	OWN	ERR	FRAM	OFLO	CRC	///	STP	ENP	ADS [23:16]							

Table 2-14. RMD1: Status Definitions

Bits	Field Name	Description
15	OWN	**Ethernet Controller Ownership** 1 = buffer is “owned” or controlled by the Ethernet controller. 0 = buffer is owned by the host. The Ethernet Controller clears this bit after transmitting all data from the buffer.
14	ERR	**Logical OR of FRAM, OFLO, and CRS** 1 = error. 0 = no error. This bit is written by the Ethernet controller.

Bits	Field Name	Description
13	FRAM	**Framing Error** 1 = a framing error has occurred. A framing error occurs when there has been a non-integer multiple of bytes in frame. If the CRC and OFLO bits do not indicate other errors, it is a driver (software) choice to invalidate such a frame (strict check), or to accept the packet (relaxed check). In the latter case, the last byte in the corresponding receive buffer must be discarded. FRAM is valid only when ENP is set and OFLO is not. This bit is written by the Ethernet controller. 0 = a framing error has not occurred.
12	OFLO	**Overflow** 1 = the RX FIFO has overflowed and some data has been lost. OFLO is valid only when the ENP bit is set. This bit is written by the Ethernet controller. 0 = the RX FIFO has not been overflown.
11	CRC	**Cyclic Redundancy Check** 1 = the FCS of the incoming frame did not match the computed FCS (CRC), or an external PHY generated a “receive error “ when sending data to Ethernet controller (RX_ER signal on MII bus). Data is therefore invalid. CRC is valid only when the ENP bit is set and the OFLO bit is not. This bit is written by the Ethernet controller. 0 = the FCS of the incoming frame matched the computed FCS (CRC), or an external PHY did not generate a “receive error “ when sending data to Ethernet controller.
10	///	**Reserved**
9	STP	**First Buffer Used** 1 = the first buffer used to store this frame. 0 = not the first buffer used to store this frame. It is used for chaining buffers. This bit is written by the Ethernet controller.
8	ENP	**Last Buffer Used** 1 = the last buffer used for this frame. 0 = not the last buffer used for this frame. If both the STP and ENP bits are set in the same descriptor, the frame fits in a single buffer, and no data chaining is to be used. This bit is written by the Ethernet controller.
7:0	ADS [23:16]	**High-order 8 Bits of the Buffer Starting Address** These bits are unchanged by the Ethernet controller.

RMD2: Buffer Byte Count

Table 2-15. RMD2: Buffer Byte Count

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	RES				BCNT											

Table 2-16. RMD2: Buffer Byte Count

Bits	Field Name	Description
15:12	RES	**Must be all ones**
11:0	BCNT	**Buffer Byte Length** This field is the length of this buffer, in bytes. It is expressed as the two's complement of the length. These bits are unchanged by the Ethernet controller.

RMD3: Byte Count

Table 2-17. RMD3: Byte Count

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	///				MCNT											

Table 2-18. RMD3: Byte Count Definitions

Bits	Field Name	Description
15:12	///	**Reserved**
11:0	MCNT	**FIFO Underflow** The number of bytes of the received frame. It is expressed as an unsigned binary integer. MCNT is valid only when the ERR bit is clear and the ENP bit is set. This bit is written by the Ethernet controller.

Descriptor Ring Polling

Once the host initializes the descriptor rings and the START bit in CSR0 is set (see CSR0: Status Register on page 33), the Ethernet controller polls the descriptor rings.

- First, the Ethernet controller polls the current RX and TX descriptor ring entries and evaluates their OWN bits. Three words are read from each descriptor: xMD0, xMD1, and xMD2. This results in two 3-word DMA transfers, after which the bus is released.
- If there is no activity on the network, the Ethernet controller continues to poll the descriptor rings periodically, based on the value in the POLLINT register, until the OWN bit in the currently polled ring is found to be 1.
- When the OWN bit is found to be 1, the Ethernet controller stops polling that ring until the operation for that buffer completes. As a result, once the RX OWN bit is polled and found to be 1, the RX descriptor ring is no longer polled until a receive frame uses the buffer.

The Poll Time Counter is a free-running counter, with the maximum poll time of 1.3 ms (clk=50 MHz).

Polling of the descriptor rings can be forced by setting the TDMD bit in CSR0 (see CSR0: Status Register on page 33). Setting TDMD after placing a frame in the TX ring reduces transmit latency, since the Ethernet controller immediately polls and begins operating on the buffer.

Each time the Ethernet controller clears a descriptor's OWN bit, the Ethernet controller polls the descriptor rings to determine the state of the OWN bit of the next descriptor. Only the required rings are polled. Therefore, if the controller just released an RX buffer, but has a TX frame waiting in the TX FIFO, only the RX ring is polled. The Ethernet controller polls one descriptor at a time; it does not perform look-ahead operations.

Transmit Descriptor Processing

If the Ethernet Controller finds the OWN bit of a TX descriptor to be zero after performing a poll operation, it waits until the next poll operation to re-evaluate the OWN bit. At any time (except transmission of chained frames), if the Ethernet controller finds the OWN bit =1 and STP=0, it:

- Requests the bus.
- Clears the OWN bit.
- Sets BUFF bit in TMD3.

The Ethernet controller always writes TMD1 and TMD3 when updating a TX descriptor ring entry.

If OWN=1 but the buffer length is zero, the Ethernet controller immediately requests the bus to clear the OWN bit. After releasing the bus, another poll is requested.

If OWN=1 and STP=1, the Ethernet controller requests the bus and begins DMA operations to move data to the TX FIFO, which is 32 bytes deep. To minimize retansmisssion retries, the TXMAC does not transmit data until the FIFO is full. Consequently, the minimum frame size that can be transmitted is 32 bytes. DMA continues until the Ethernet controller fetches all data in this buffer.

If ENP=0:

- The descriptor is released (OWN cleared to zero).
- A poll operation is performed.
- The data in the next buffer is DMAed.

After the Ethernet controller finds a descriptor with ENP=1:

- The data is DMAed into the FIFO.
- The Ethernet controller waits until the entire frame has been sent.
- After the entire frame is sent, the Ethernet controller updates TMD1 and TMD3 with the appropriate status bits.
- A poll operation is performed and processing continues.

Note: *The current frame must complete transmission, the descriptor must be released, a poll operation must be performed, and the next TX frame must load into the FIFO, when there are back to back frames.*

When the last transmit descriptor of a frame is released (OWN cleared to 0 and ENP or ERR set to 1), the TINT bit of CSR0 is set to indicate to the host that the frame has been released.

Receive Descriptor Processing

If the Ethernet controller finds the OWN bit of an RX descriptor to be zero after a poll operation, it waits until the next poll operation to re-evaluate the OWN bit. If the OWN bit is found to be a one, no more polls of the RX descriptor ring occur until the current buffer fills with an incoming frame.

When a frame is received, the Ethernet controller waits until destination address field of the frame is placed in the FIFO. This provides time for the Ethernet controller to verify the destination address with the currently active addressing schemes. If the address does not match, the FIFO write pointer resets and no data is sent from the address filter block. Any received frame shorter than 64 bytes is considered to be a fragment resulting from collision. The next frame that meets the minimum size uses same set of descriptors, since short frames do not increment the descriptor pointer.

Note: *For testing purposes, the RPA bit in CSR4(7) can be set to 1 to have the Ethernet controller accept frames shorter then 64 bytes.*

After the Ethernet controller verifies the destination address, the BMU checks the status of the current receive buffer, which is stored in the Ethernet controller.

- If the OWN bit = 1, DMA begins immediately to store the data into the host RAM.
- If the OWN bit = 0 and the FIFO does overflow, the data remains in the FIFO for eventual DMA. The OFLO bit is set in the descriptor to indicate that the data is bad. If the OWN bit remains zero and consecutive frames cannot be received, a “missed frames counter” in CSR112 increments.

If the frame length exceeds the length of the current receive buffer, the Ethernet controller updates the RMD1 and RMD3 fields of the current descriptor, and polls the next descriptor immediately. If the OWN bit of the next descriptor is a one, DMA begins again until all data for the frame is stored into the host's memory; the frame is spread across multiple buffers via data chaining.

Note: *RMD3 contains the total number of bytes of the frame when ENP is set.*

Management Functions

The MII management interface provides a simple, 2-wire, serial interface for connecting the Ethernet controller to a managed PHY. This interface allows the Ethernet controller to control and collect status information from the PHY.

The management interface consists of a pair of signals that physically transport the management information across the MII, a frame format and a protocol specification for exchanging management frames, and a register set that can be read and written using these frames. The register definition specifies a basic register set with an extension mechanism.

The MII basic register set consists of two registers:

- Management Register 0 — see Table 2-20 on page 27.
- Management Register 1 — see Table 2-21 on page 27.

All PHYs that provide an MII shall incorporate the basic register set. Registers 2 through 10 are part of the extended register set.

Table 2-19 lists the full set of 802.3 management registers. This section describes only Management Registers 0 and 1, which must be implemented in an MI-compatible PHY. For all other PHY registers, refer to the documentation provided by the PHY manufacturer.

Table 2-19. Ethernet Management Register Summary

Register Address	Register Name	Basic/Extended	See Page
0	Management Register 0	Basic	27
1	Management Register 1	Basic	30
2:3	PHY Identifier	Extended	72 and 73
4	Auto-negotiation Advertisement	Extended	74
5	Auto-negotiation Link Partner Ability	Extended	75
6	Auto-negotiation Expansion	Extended	76
16:31	Vendor Specific	Extended	77 through 87

Reset

Resetting a PHY sets the management registers to default states. In this way, resetting the PHY can change the PHY’s internal state and the physical link associated with the PHY.

The RESET bit [15] in Management Register 0 is self-clearing; a PHY returns a value of one in bit [15] until the reset process completes. A PHY is not required to accept a write transaction to Management Register 0 until the reset process completes. In addition, writes to bits of Management Register 0 other than bit [15] have no effect until the reset process completes. The reset process must complete within 0.5 seconds after setting bit [15] of Management Register 0.

Table 2-20. Management Register 0

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	Reset	Loopback	Speed Selection (LSB)	Auto-Negotiation Enable	Power Down	Isolate	Restart Auto-Negotiation	Duplex Mode	Collision Test	Speed Selection (MSB)	///					
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-21. Management Register 0 Definitions

Bits	Field Name	Description
15	Reset	**Reset** 1 = PHY reset. 0 = normal operation. This bit is self-clearing.
14	Loopback	**Loopback Mode** 1 = enable loopback mode. 0 = disable loopback mode.
13	Speed Selection (LSB)	**Speed Selection (LSB)** This bit is used with bit [6] to select the speed. See Table 2-22.
12	Auto-Negotiation Enable	**Auto-Negotiation Enable** 1 = enable auto-negotiation process. 0 = Disable auto-negotiation process.
11	Power Down	**Power Down** 1 = power down. 0 = normal operation.
10	Isolate	**Isolate PHY from MII** 1 = electrically Isolate PHY from MII. 0 = normal operation.
9	Restart Auto-Negotiation	**Restart Auto-Negotiation** 1 = restart auto-negotiation process. 0 = normal operation. This bit is self-clearing.
8	Duplex Mode	**Half-/Full-Duplex Mode** 1 = full-duplex. 0 = half-duplex.
7	Collision Test	**Collision Test** 1 = enable COL signal test. 0 = disable COL signal test.
6	Speed Selection (MSB)	**Speed Selection (MSB)** This bit is used with bit [13] to select the speed. See Table 2-22.
5:0	///	**Reserved** Write as 0, ignore on read.

Table 2-22. Selecting Speed with Bits [6] and [13]

Bit [6]	Bit [13]	Speed
1	1	Reserved
1	0	1000 Mb/s
0	1	100 Mb/s
0	0	10 Mb/s

Speed Selection

Link speed can be selected via the auto-negotiation process. If auto-negotiation is disabled (bit [12] of Management Register 0 cleared to zero), link speed can be selected using the speed-selection bits [6] and [13] of Management Register 0, as shown in Table 2-22.

When auto-negotiation is enabled, bits [6] and [13] of Management Register 0 can be read or written, but their state has no effect on the link configuration. Moreover, it is not necessary for bits [6] and [13] to reflect the operating speed of the link when read.

If a PHY reports via bits [15:9] of Management Register 0 and bits [15:13] of Management Register 5 that it cannot operate at all speeds, the value of bits [6] and [13] of Management Register 0 correspond to a speed at which the PHY can operate and any attempt to change the bits to an invalid setting is ignored. The default value of bits [6] and [13] of Management Register 0 are the encoding of the highest data rate at which the PHY can operate, as indicated by bits [15:9] of Management Register 1.

Duplex Mode

Like speed selection, duplex mode can be selected either via the auto-negotiation process or through duplex selection (bit [8] of the Management Register 0) when auto-negotiation is disabled (bit [12] of the Management Register 0 cleared).

When auto-negotiation is enabled, bit [8] of the Management Register 0 can be read or written; however, the state of bit [8] has no effect on the link configuration. If a PHY reports via bits [15:9] of Management Register 0 and bits [15:12] of Management Register 5 that it can operate in only one duplex mode, the bit [8] value of Management Register 0 corresponds to the mode in which the PHY can operate; any attempt to change the bit [8] setting of Management Register 0 is ignored.

Auto-Negotiation

To enable the auto-negotiation process, set bit [12] of Management Register 0 to one. Enabling or disabling auto-negotiation also determines whether bits [13], [8], and [6] of Management Register 0 affect the link configuration and station operation.

- If a PHY reports via bit [3] of Management Register 1 that it lacks the ability to perform auto-negotiation, the PHY returns a value of zero in bit [12] of Management Register 0.
- If a PHY reports via bit [3] of Management Register 1 that it lacks the ability to perform auto-negotiation, bit [12] of Management Register 0 should always write as zero. Any attempt to write a one to bit [12] should be ignored.

The default value of bit [12] of Management Register 0 is one; however, if the PHY reports via bit [3] of Management Register 1 that it lacks the ability to perform auto-negotiation, the default value of bit [12] is zero.

Restarting Auto-Negotiation

If a PHY reports via bit [3] of Management Register 1 that it lacks the ability to perform auto-negotiation, or if auto-negotiation is disabled, the PHY returns a value of zero in bit [9] of Management Register 0. In this case, bit [9] always write as zero, and any attempt to write a one to bit [9] should be ignored. Otherwise, the auto-negotiation process restarts by setting bit [9] to a logical one.

The restart auto-negotiation bit (bit [9] of Management Register 0) is self-clearing; a PHY returns a value of one in bit [9[until the auto-negotiation process starts. The auto-negotiation process is not affected by writing a zero to bit [9]. The default value of bit [9] is zero.

All bits in Management Register 1 are Read Only; a write to this register has no effect.

Neither Management Register 0 or Management Register 1 provides status about whether the current operating mode is full-duplex or 100 Mb/s. These registers show the capabilities of the

PHY or link, and do not necessarily match parameters of actual connections. To take advantage of full-duplex mode, bit FDEN of the MIIP: MII Management Register (see Table 2-77 on page 50) must be set appropriately.

If auto-negotiation is disabled, the PHY and Ethernet controller can be set to the same mode (full- or half-duplex). If auto-negotiation is enabled and performed, the current mode can be read from one of the vendor-specific MII registers (Management registers 16 through 31).

Table 2-23. Management Register 1

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	100Base-T4	100Base-TX FULL-DUPLEX	100Base-TX HALF-DUPLEX	10MB/S FULL-DUPLEX	10MB/S HALF-DUPLEX	100Base-T2 FULL-DUPLEX	100Base-T2 HALF-DUPLEX	EXTENDED STATUS	///	MF PREAMBLE SUPPRESSION	AUTO-NEGOTIATION COMPLETE	REMOVE FAULT	AUTO-NEGOTIATION ABILITY	LINK STATUS	JABBER DETECT	EXTENDED CAPABILITY
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2-24. Management Register 1 Definitions

Bits	Field Name	Description
15	100Base-T4	**100Base-T4** 1 = PHY can perform 100Base-T4. 0 = PHY cannot perform 100Base-T4.
14	100Base-TX FULL-DUPLEX	**100Base-X Full-Duplex** 1 = PHY can perform full duplex 100Base-X. 0 = PHY cannot perform full duplex 100Base-X.
13	100Base-TX HALF-DUPLEX	**100Base-X Half-Duplex** 1 = PHY can perform half duplex 100Base-X. 0 = PHY cannot perform half duplex 100Base-X.
12	10 MB/S FULL-DUPLEX	**10 Mb/s Full-Duplex** 1 = PHY can operate at 10 Mb/s in full duplex mode. 0 = PHY cannot operate at 10 Mb/s in full duplex mode.
11	10 MB/S HALF-DUPLEX	**10 Mb/s Half-Duplex** 1 = PHY can operate at 10 Mb/s in half-duplex mode. 0 = PHY cannot operate at 10 Mb/s in half-duplex mode.
10	100Base-T2 FULL-DUPLEX	**100Base-T2 Full-Duplex** 1 = PHY can perform full-duplex 100Base-T2. 0 = PHY cannot perform full-duplex 100Base-T2.
9	100Base-T2 HALF-DUPLEX	**100Base-T2 Half-Duplex** 1 = PHY can perform half duplex 100Base-T2. 0 = PHY cannot perform half duplex 100Base-T2.
8	EXTENDED STATUS	**Extended Status** 1 = Extended status information in Register 15. 0 = No extended status information in Register 15.
7	///	**Reserved** This bit is ignored if read.
6	MF PREAMBLE SUPPRESSION	**Suppression** 1 = PHY accepts management frames with preamble suppressed. 0 = PHY will not accept management frames with preamble suppressed.
5	AUTO-NEGOTIATION COMPLETE	**Auto-Negotiation Complete** 1 = auto-negotiation process completed. 0 = auto-negotiation process not completed.
4	REMOVE FAULT	**Remove Fault** 1 = remote fault condition detected. 0 = no remote fault condition detected.

Bits	Field Name	Description
3	AUTO-NEGOTIATION ABILITY	**Auto-Negotiation Ability** 1 = PHY can perform auto-negotiation. 0 = PHY is cannot perform auto-negotiation.
2	LINK STATUS	**Link Status** 1 = link is up. 0 = link is down.
1	JABBER DETECT	**Jabber Condition Detected** 1 = jabber condition detected. 0 = no jabber condition detected.
0	EXTENDED CAPABILITY	**Extended Register Capabilities** 1 = extended register capabilities. 0 = basic register set capabilities only.

Management Frame

CSR46: Poll Time Counter (described in Table 2-64 on page 46) describes the format of Management frames transmitted on the MII Management Interface. The order of bit transmission is from left to right. The first transmitted bits of Device Address, Register Address, and DATA fields are the most-significant bit of the PHY address, register address, and data of the register being accessed.

The IDLE condition on MDIO is a high-impedance state. All three state drivers are disabled and the PHY's pull-up resistor pulls the MDIO line to a logical one.

At the beginning of each transaction, the Ethernet controller sends a preamble sequence (PRE) of 32 contiguous logical one bits on MDIO, with 32 corresponding cycles on MDC, to provide the PHY with a pattern for establishing synchronization. The PHY observes the sequence of 32 contiguous one bits on MDIO, with 32 corresponding cycles on MDC, before responding to any transaction.

If the Ethernet controller determines that the PHY connected to the MDIO signal can accept management frames that are not preceded by the preamble pattern, the controller can suppress the pre-amble pattern and initiate management frames with the Start-of-Frame pattern.

Table 2-25. Management Frame Format

	Fields							
	PRE	Start	Opcode	Device Address	Register Address	TA	Data	Idle
Read	1…1	01	10	AAAAA	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z
Write	1…1	01	01	AAAAA	RRRRR		DDDDDDDDDDDDDDDD	Z

**

Ethernet Controller Registers

The Ethernet controller is software compatible with AMD's Am79C960 Ethernet controller. All Ethernet controller registers are addressed in microprocessor I/O space. Table 2-26 shows the I/O address map for the Ethernet controller registers.

Table 2-26. Ethernet I/O Address Map

MAC 0 Hex Address	MAC 1 Hex Address	Number of Bytes	Register
9000	9100	16	Reserved
9010	9110	2	RDP
9012	9112	2	RAP
9014	9114	2	RESET
9018	9118	2	MIIP

Register Access Port

All CSR registers are accessed using a 2-step process.

- First, the address of the register to be accessed is written into the Register Address Port (RAP) register.
- Then the contents of the register pointed to by the RAP is accessed via the Register Data Port (RDP) register.

RAP is used as an index register or pointer to the register you want to access. All CSR registers clear on reset, unless shown differently. Reading the Reset register causes the corresponding MAC to be reset the same as a hardware reset.

Ethernet Controller Register Summary

Table 2-27. Ethernet Controller Register Summary

Hex Offset	Mnemonic	Register Description	Page
0	CSR0	Status register	33
1	CSR1	IADR register	35
2	CSR2	IADR register	35
3	CSR3	Interrupt Mask register	36
4	CSR4	Features Control register	37
8	CSR8	LADF register [15:0]	40
9	CSR9	LADF register [31:16]	40
A	CSR10	LADF register [47:32]	41
B	CSR11	LADF register [63:48]	41
C	CSR12	PADR register [15:0]	42
D	CSR13	PADR register [31:16]	42
E	CSR14	PADR register [47:32]	43
F	CSR15	Mode register	43
10	CSR16	IADR register	44
11	CSR17	IADR register	45
18 19	CSR24 CSR25	Base Address of Rx Ring register	45
1E 1F	CSR30 CSR31	Base Address of Tx Ring register	46
2E	CSR46	Poll Time Counter register	46
2F	CSR47	Polling Interval register	47
4C	CSR76	Receive Ring Length register	47

Hex Offset	Mnemonic	Register Description	Page
4E	CSR78	Transmit Ring Length register	48
58 59	CSR88 CSR89	Chip ID register	48
70	CSR112	Missed Frame Count register	49
72	CSR114	Receive Collision Count register	49

Ethernet Controller Register Definitions

CSR0: Status Register

Table 2-28. CSR0: Status Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	0h															
FIELD	ERR	///	CERR	MISS	///	RINT	TINT	IDON	INSTR	IENA	RXON	TXON	TDMD	STOP	STRT	INIT
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W

Table 2-29. CSR0: Status Register Definitions

Bits	Field Name	Description
15	ERR	**Logical OR of CERR and MISS** This bit is Read Only.
14	///	**Reserved** This Read Only bit always reads back as 0.
13	CERR	**COL Pin** 1 = COL pin did not go HIGH within 20 network bit times after transmission stops. Used during SQE test, and is valid only at 10Mb/s half-duplex mode. CERR does not assert IRQ; however, it does set ERR. CERR is cleared by writing a 1. 0 = no effect. To clear CERR, write a 1 or set the STOP bit.
12	MISS	**Incoming Frame Lost** 1 = an incoming frame is lost because the RX_ADDRFLT FIFO has no room for the incoming packet. This is the only indication that data has been lost, since the packet was not placed in the FIFO and/or no descriptors are available to record the information. MISS asserts IRQ if the mask bits are 0. 0 = no effect. To clear MISS, write a 1 or set the STOP bit.
11	///	**Reserved** Always read and write as 0.
10	RINT	**RINT** 1 = OWN bit RMD1 clears and ENP=1 because a complete frame was placed in the RX descriptor ring. RINT asserts IRQ if the mask bits are 0. 0 = no effect. To clear RINT, write a 1 or set the STOP bit.

Bits	Field Name	Description
9	TINT	**TINT** 1 = the OWN bit TMD1 clears, ENP=1, and TX_EN goes LOW. This occurs when a complete frame is transmitted. TINT asserts IRQ if the mask bits are 0. 0 = no effect. To clear TNT, write a 1 or set the STOP bit.
8	IDON	**Read Initialization Block** 1 = Ethernet controller reads the entire initialization block from memory. IDON asserts IRQ if mask bits are 0. 0 = no effect. To clear IDON, write a 1 or set the STOP bit.
7	INSTR	**Logical OR of MISS, MFCO, RCVCCO, RINT, TINT, IDON, TXSTRT, and PAUSE** The IRQ pin asserts if IENA=1. This bit is Read Only.
6	IENA	**Enable INSTR Bit** 1 = enable the INSTR bit to assert the IRQ pin. 0 = disable IRQ. To set IENA, write a 1. To clear it, write a 0 or set the STOP bit.
5	RXON	**Receive Function** 1 = enable receive function. RXON is set when DRX (CSR15.0)=0 and STRT=1. 0 = disable receive function. This bit is Read Only.
4	TXON	**Transmit Function** 1 = enable transmit function. TXON is set when DTX (CSR15.1)=0 and STRT=1. 0 = disable transmit function. This bit is Read Only.
3	TDMD	**Poll RX and TX Descriptors** 1 = Buffer Management State Machine polls the RX and TX descriptor rings. Typically, this bit hastens frame transmission. If this bit is not set HIGH after a TX frame is added to the ring, the Ethernet controller does not DMA until the poll-time counter expires. To set TDMD, write a 1. 0 = no effect. The Buffer Management State Machine clears TDMD after the Descriptor Ring Poll completes. To clear TDMD, set the STOP bit.
2	STOP	**Disable from External Activity** 1 or RESET = disable DSTni from all external activity and force numerous bits into their reset state. This setting overrides STRT or INIT when the bits are set at the same time. 0 = no effect. To clear STOP, set the STRT or INIT bit.
1	STRT	**Enable Ethernet Controller** 1 = enable Ethernet controller. Buffer Management starts and frames can be transmitted or received. Setting STRT clears the STOP bit. If STRT and INIT are set at the same time, the INIT block is read, then the Ethernet controller starts operating. 0 = no effect. To clear STRT, set the STOP bit.
0	INIT	**DMA Block from Memory** 1 = Ethernet controller to DMA the INIT block from memory. Setting INIT clears the STOP bit. If STRT and INIT are set at the same time, the INIT block is read first, then the Ethernet controller begins operation. 0 = no effect. To clear INIT, set the STOP bit.

CSR1: IADDR Register

CSR1 is identical to CSR16 (see page 44).

Table 2-30. CSR1: IADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	1h															
FIELD	LADR [15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-31. CSR1: IADDR Register Definitions

Bits	Field Name	Description
15:0	LADR[15:0]	**Lower 16 Bits of the Initialization Address Register** Bits [2:0] must be 0. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR2: IADDR Register

CSR2 is identical to CSR17 (see page 45).

Table 2-32. CSR2: IADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	2h															
FIELD	///								IADR[23:16]							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-33. CSR2: IADDR Register Definitions

Bits	Field Name	Description
15:8	///	**Reserved** Read and write as 0.
7:0	IADR[23:16]	**Upper 8 Bits of the Initialization Address Register** Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR3: Interrupt Mask Register

Table 2-34. CSR3: Interrupt Mask Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	3h															
FIELD	///			MISSM	///	RINTM	TINTM	IDONM	///			DTX2PD	///			
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R

Table 2-35. CSR3: Interrupt Mask Register Definitions

Bits	Field Name	Description
15:13	///	**Reserved** Read and write as 0.
12	MISSM	**Mask MISS Bit** To clear INIT, set the STOP bit (bit [2] of CSR0: Status Register). 1 = mask MISS bit in CSR0 and do not set the INSTR bit. MISSM is not affected by STOP.
11	///	**Reserved** Read and write as 0.
10	RINTM	**Mask RINT Bit** 1 = mask RINT bit [10] in CSR0 and do not set the INSTR bit [7]. RINTM is not affected by STOP.
9	TINTM	**Mask TINT Bit** 1 = mask TINT bit [9] in CSR0 and do not set the INSTR bit [7]. TINTM is not affected by STOP.
8	IDONM	**Mask IDON Bit** 1 = mask IDON bit [8] in CSR0 and do not set the INSTR bit [7]. IDONM is not affected by STOP.
7:5	///	**Reserved** Read and write as 0.
4	DTX2PD	**Disable Transmit 2-part Deferral** 1 = disable transmit 2-part deferral. DTX2PD is not affected by STOP.
3:0	///	**Reserved** Read and write as 0.

CSR4: Features Control Register

Table 2-36. CSR4: Features Control Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	4h															
FIELD	ENTST	///		DPOLL	APAD_TX	///	MFCO	MFCOM	RPA	BAK FAST	RCVCCO	RCVCCOM	TXSTRT	TXSTRTM	PAUSE	PAUSEM
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2-37. CSR4: Features Control Register Definitions

Bits	Field Name	Description
15	ENTST	**Enable Reading and Writing** 1 = enable reading and writing to various registers, even when the STOP bit (bit [2] of CSR0: Status Register) is 0. Also, enable various test modes to simplify chip testing. Used during chip testing only. 0 = test mode is not enabled.
14:13	///	**Reserved** Read and write as 0.
12	DPOLL	**Disable Transmit Polling** 1 = disable transmit polling. Transmit ring is polled only when receive ring is polled or when the TDMD bit in CSR0 is set. 0 = enable automatic polling of transmit ring buffers when the Poll-Time Counter (CSR46) reaches zero.
11	APAD_TX	**Auto-Pad Frames** 1 = automatically pad frames less than 64 bytes long. Add zeroes to the frame to extend its length to 64 bytes, including the FCS. The FCS includes the pad bytes. APAD_TX overrides the DXMTFCS bit (CSR15.3). 0 = frames are not padded.
10	///	**Reserved** Read and write as 0.
9	MFCO	**Missed Frame Counter Overflow** 1 = Missed Frame Counter in CSR112 overflowed. MFCO asserts INSTR if the mask bits are 0. To clear MFCO, write a 1 or set the STOP bit. 0 = no effect.
8	MFCOM	**Mask MFCO** 1 = mask the MFCO from asserting INSTR. 0 = enable MFCO to assert INSTR.MFCOM is set by RESET.
7	RPA	**Accept Runts** Provided for testing purposes only. 1 = let RX MAC receive runt packets. 0 = do not accept runt packets. TX MAC always accepts runts.
6	BAKFAST	**Backoff Timer** Provided for testing purposes only. 0 = required for normal operation. 1 (and ENTEST =1) = backoff timer in the TX MAC state machine limits the backoff time to 32 slot times, even if number of retransmission attempts is greater than 5.

Bits	Field Name	Description
5	RCVCCO	**Receive Collision Counter** 1 = Receive Collision Counter (CSR114) has overflowed. RCVCCO asserts INSTR if the mask bits are 0. To clear RCVCCO, write a 1 or set the STOP bit. 0 = no effect.
4	RCVCCOM	**Mask RCVCCO** 1 = mask RCVCCO from asserting INSTR. 0 = enable RCVCCO to assert INSTR. RCVCCOM is set by RESET.
3	TXSTRT	**Transmit Frame Started** 1 = transmit frame has begun. 0 = no effect. The TX_MAC state machine sets this bit when TENA is asserted. To clear TXSTRT, write a 1 or set the STOP bit. 0
2	TXSTRTM	**Mask TXSTRT** 1 = mask the TXSTRT from asserting INSTR. 0 = enable TXSTRT to assert INSTR. TXSTRTM is set by RESET.
1	PAUSE	**Pause Control Frame Received** 1 = Ethernet controller received a Pause Control Frame. To clear PAUSE, write a 1 or set the STOP. 0 = no effect. PAUSE assert INSTR if the mask bit is 0.
0	PAUSEM	**Mask PAUSE** 1 = mask PAUSE from asserting INSTR. 0 = enable PAUSE to assert INSTR. PAUSEM is set by RESET.

The logical address filter (LADF) in the MAC lets you perform multicasting. This functionality is implemented through a 64-bit hash table and CRC calculations.

When the Ethernet controller receives an incoming packet's destination field, the following actions occur (note that the Ethernet controller receives data a nibble at a time):

- The Ethernet controller performs a CRC calculation on the destination data.
- A counter counts the number of destination bytes.
- After all six bytes of destination data are received (bcnt=5), the CRC's 6 most-significant bits are 000100 and entry 4 of the hash table is read (LADF[4]).
  - If LADF[4]= 1, the packet is accepted.
  - If LADF[4] = 0, the packet is rejected.

Table 2-38 shows a sample mapping from a given LADF bit to a destination address that is accepted if the corresponding LADF bit is set. Each bit in the LADF selects an entire range of DEST addresses to be accepted.

The 64-bit hash table for the LADF is composed of four 16-bit registers, CSR8-CSR11, as shown in "CSR8: LADF", "CSR9: LADF", "CSR10: LADF", and "CSR11: LADF".

Table 2-38. Sample Mapping, LADF Bit

Byte #	Byte #	LADF Bit	Destination Address	Byte #	Byte #	LADF Bit	Destination Address
0	0	0	85 00 00 00 00 00	4	0	32	21 00 00 00 00 00
0	1	1	A5 00 00 00 00 00	4	1	33	01 00 00 00 00 00
0	2	2	E5 00 00 00 00 00	4	2	34	41 00 00 00 00 00
0	3	3	C5 00 00 00 00 00	4	3	35	71 00 00 00 00 00
0	4	4	45 00 00 00 00 00	4	4	36	E1 00 00 00 00 00
0	5	5	65 00 00 00 00 00	4	5	37	C1 00 00 00 00 00
0	6	6	25 00 00 00 00 00	4	6	38	81 00 00 00 00 00
0	7	7	05 00 00 00 00 00	4	7	39	A1 00 00 00 00 00
1	0	8	2B 00 00 00 00 00	5	0	40	8F 00 00 00 00 00
1	1	9	0B 00 00 00 00 00	5	1	41	BF 00 00 00 00 00
1	2	10	4B 00 00 00 00 00	5	2	42	EF 00 00 00 00 00
1	3	11	6B 00 00 00 00 00	5	3	43	CF 00 00 00 00 00
1	4	12	EB 00 00 00 00 00	5	4	44	4F 00 00 00 00 00
1	5	13	CB 00 00 00 00 00	5	5	45	6F 00 00 00 00 00
1	6	14	8B 00 00 00 00 00	5	6	46	2F 00 00 00 00 00
1	7	15	BB 00 00 00 00 00	5	7	47	0F 00 00 00 00 00
2	0	16	C7 00 00 00 00 00	6	0	48	63 00 00 00 00 00
2	1	17	E7 00 00 00 00 00	6	1	49	43 00 00 00 00 00
2	2	18	A7 00 00 00 00 00	6	2	50	03 00 00 00 00 00
2	3	19	87 00 00 00 00 00	6	3	51	23 00 00 00 00 00
2	4	20	07 00 00 00 00 00	6	4	52	A3 00 00 00 00 00
2	5	21	27 00 00 00 00 00	6	5	53	83 00 00 00 00 00
2	6	22	67 00 00 00 00 00	6	6	54	C3 00 00 00 00 00
2	7	23	47 00 00 00 00 00	6	7	55	E3 00 00 00 00 00
3	0	24	69 00 00 00 00 00	7	0	56	CD 00 00 00 00 00
3	1	25	49 00 00 00 00 00	7	1	57	ED 00 00 00 00 00
3	2	26	09 00 00 00 00 00	7	2	58	AD 00 00 00 00 00
3	3	27	29 00 00 00 00 00	7	3	59	8D 00 00 00 00 00
3	4	28	A9 00 00 00 00 00	7	4	60	0D 00 00 00 00 00
3	5	29	89 00 00 00 00 00	7	5	61	2D 00 00 00 00 00
3	6	30	C9 00 00 00 00 00	7	6	62	6D 00 00 00 00 00
3	7	31	E9 00 00 00 00 00	7	7	63	4D 00 00 00 00 00

CSR8: LADF Register

Table 2-39. CSR8: LADF Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	8h															
FIELD	LADF [15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-40. CSR8: LADF Register Definitions

Bits	Field Name	Description
15:0	LADF[15:0]	**Logical Address Filter Bits [15:0]**

CSR9: LADF Register

Table 2-41. CSR9: LADF Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	9h															
FIELD	LADF [31:16]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-42. CSR9: LADF Register Definitions

Bits	Field Name	Description
15:0	LADF[31:16]	**Logical Address Filter Bits [31:16]**

CSR10: LADF Register

Table 2-43. CSR10: LADF Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Ah															
FIELD	LADF [47:32]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-44. CSR10: LADF Register Definitions

Bits	Field Name	Description
15:0	LADF[47:32]	**Logical Address Filter Bits [47:32]**

CSR11: LADF Register

Table 2-45. CSR11: LADF Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Bh															
FIELD	LADF [63:48]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-46. CSR11: LADF Register Definitions

Bits	Field Name	Description
15:0	LADF[63:48]	**Logical Address Filter Bits [63:48]**

CSR12: PADR Register

Table 2-47. CSR12: PADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Ch															
FIELD	PADR [15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-48. CSR12: PADR Register Definitions

Bits	Field Name	Description
15:0	PADR[15:0]	**Physical Address Filter Bits [15:0]** The PADR bits are received PADR[0] first and PADR[47] last. Read/write only when STOP bit (bit [2] of CSR0: Status Register) =1.

CSR13: PADR Register

Table 2-49. CSR13: PADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Dh															
FIELD	PADR [31:16]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-50. CSR13: PADR Register Definitions

Bits	Field Name	Description
15:0	PADR[31:16]	**Physical Address Filter Bits [31:16]** Read/write only when STOP bit (bit [2] of CSR0: Status Register) =1.

CSR14: PADR Register

Table 2-51. CSR14: PADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Eh															
FIELD	PADR [47:32]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-52. CSR13: PADR Register Definitions

Bits	Field Name	Description
15:0	PADR[47:32]	**Physical Address Filter Bits [47:32]** Read/write only when STOP bit (bit [2] of CSR0: Status Register) =1.

CSR15: Mode Register

Table 2-53. CSR15: Mode Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	Fh															
FIELD	PROM	DRXBC	DRXPA	///	DPAUSE	///					DRTY	///	DTXFCS	///	RTX	DRX
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R/W	R/W	R/W	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W

Table 2-54. CSR15: Mode Register Definitions

Bits	Field Name	Description
15	PROM	**Promiscuous Mode** 1 = enable promiscuous mode. All receive frames are accepted, regardless of the destination address. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
14	DRXBC	**Disable Receive Multicast Messages** 1 = disable receive Multicast messages. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
13	DRXPA	**Disable Receive Physical Address Messages** 1 = disable receive Physical Address messages. Ignore frames that match the physical address. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
12	///	**Reserved** Read and write as zero.
11	DPAUSE	**Disable Auto-Pause Operation** 1 = disable automatic pause operation. 0 = enable pause operation. When the Ethernet controller receives a Control Frame with a Pause opcode, the next frame to be transmitted waits until the requested pause time expires.
10:6	///	**Reserved** Read and write as zero.

Bits	Field Name	Description
5	DRTY	**Disable Transmit Retries** 1 = disable transmit retries. Only one attempt is made to transmit a TX frame. 0 = enable up to 16 TX retries. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
4	///	**Reserved** Read and write as zero.
3	DTXFCS	**Transmit CRC Generation** 1 = disable transmit CRC generation. 0 = enable transmit CRC generation; FCS appends to the frame. See the ADD_FCS bit in TMD1 (see page 20). Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
2	///	**Reserved** Read and write as zero.
1	RTX	**Disable Transmit Operation** 1 = disable transmit operation. Accesses to the transmit ring buffers stops. 0 = let Buffer Management Unit access the TX ring buffers. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.
0	DRX	**Disable Receive Operation** 1 = disable receive operation. Accesses to the receive ring buffers stops. 0 = enable Buffer Management Unit to access the RX ring buffers. Read/write only when STOP bit (bit [2] of CSR0: Status Register) is set.

CSR16: IADDR Register

CSR16 is identical to CSR1 (see page 35).

Table 2-55. CSR16: IADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	10h															
FIELD	LADR [15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-56. CSR16: IADDR Register Definitions

Bits	Field Name	Description
15:0	LADR[15:0]	**Lower 16 Bits of the Initialization Address Register** Bits [2:0] must be 0. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR17: IADDR Register

CSR17 is identical to CSR2 (see page 35).

Table 2-57. CSR17: IADR Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	11h															
FIELD	///								IADR[23:16]							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-58. CSR17: IADDR Register Definitions

Bits	Field Name	Description
15:8	///	**Reserved** Read and write as 0.
7:0	IADR[23:16]	**Upper 8 Bits of the Initialization Address Register** Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR24-25: Base Address of RX Ring Register

Table 2-59. CSR24-25: Base Address of RX Ring Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSET	CSR24: 18h CSR25: 19h															
FIELD	///								BADR							
RESET	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	BADR															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-60. CSR24-25: Base Address of RX Ring Register Definitions

Bits	Field Name	Description
31:24	///	**Reserved** Read and write as 0.
23:0	BADR	**RX Ring Base Address** CSR24[0] is the least-significant bit and CSR25[7] is the most-significant bit. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR30-31: Base Address of TX Ring Register

Table 2-61. CSR30-31: Base Address of TX Ring Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSET	CSR30: 1Eh CSR31: 1Fh															
FIELD	///								BADR							
RESET	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	BADR															
RESET	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-62. CSR30-31: Base Address of TX Ring Register Definitions

Bits	Field Name	Description
31:24	///	**Reserved** Read and write as 0.
23:0	BADR	**TX Ring Base Address** CSR30[0] is the least-significant bit and CSR31[7] is the most-significant bit. Bits[2:0] are assumed to be 000. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR46: Poll Time Counter Register

Table 2-63. CSR46: Poll Time Counter Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	2Eh															
FIELD	POLL [15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-64. CSR46: Poll Time Counter Register Definitions

Bits	Field Name	Description
15:0	POLL [15:0]	**Poll Time Counter Value** When this counter overflows, a poll of the TX and RX descriptor rings is performed and the counter reloads with CSR47. The Poll Time Counter increments on every CLK when CSR0 STRT=1. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR47: Polling Interval Register

Table 2-65. CSR47: Polling Interval Register

BIT	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	2Fh																
FIELD	POLLINT [15:0]																
RESET	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-66. CSR47: Polling Interval Register Definitions

Bits	Field Name	Description
15:0	POLLINT [15:0]	**Polling Interval** Contains the value that loads into the Poll Time Counter when the counter overflows. The value is the two's complement of the desired interval. Each count is one CLK period. The least-significant 4 bits, [3:0], are not used and assumed to be zero. The register is 17 bits wide, with bit [16] assumed to be a one. The following value loads into CSR46: {1, POLLINT[15:4],000} Consequently, the default value of 0000 corresponds to 32K clocks = 1.3 ms @CLK=50 MHz. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR76: Receive Ring Length Register

Table 2-67. CSR76: Receive Ring Length Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	4Ch															
FIELD	RXLEN															
RESET	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-68. CSR76: Receive Ring Length Register Definitions

Bits	Field Name	Description
15:0	RXLEN	**Length of Rx Ring** Contains the length of the RX ring in two's-complement form. It is assigned the value from the RLEN field during initialization. This register can also be programmed manually to accommodate ring length of any size. Only the low 12 bits are used, bits [15:12] are assumed to be ones. Consequently, the maximum ring length is 4K buffers. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR78: Transmit Ring Length Register

Table 2-69. CSR78: Transmit Ring Length Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	4Eh															
FIELD	TXLEN															
RESET	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 2-70. CSR78: Transmit Ring Length Register Definitions

Bits	Field Name	Description
15:0	TXLEN	**Length of Tx Ring** Contains the length of the TX ring in two's-complement form. It is assigned the value from the TLEN field during initialization. This register can also be programmed manually to allow ring length of any size. Only the low 12 bits are used, bits [15:12] are assumed to be ones. Consequently, the maximum ring length is 4K buffers. Read/write only when the STOP bit (bit [2] of CSR0: Status Register) =1.

CSR88-89: Chip ID Register

Table 2-71. CSR88-89: Chip ID Register

BIT	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSET	CSR88: 58h CSR89: 59h															
FIELD	REV				PART_NUM											
RESET	–	–	–	–	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FIELD	PART_NUM				MFG_ID											///
RESET	0	0	0	0	0	1	0	0	1	1	1	1	1	1	1	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2-72. CSR88-89: Chip ID Register Definitions

Bits	Field Name	Description
31:28	REV	**Chip Revision** These Read Only bits show the chip revision level.
27:12	PART_NUM	**Part Number** These Read Only bits are the 16-bit code for the part number.
11:1	MFG_ID	**Manufacturer ID**
0	///	**Reserved** This Read Only bit is always 1.

CSR112: Missed Frame Count Register

Table 2-73. CSR112: Missed Frame Count Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	70h															
FIELD	///								MFC							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2-74. CSR112: Missed Frame Count Register Definitions

Bits	Field Name	Description
15:8	///	**Reserved** Read as 0.
7:0	MFC	**Number of Missed Frames** Counts the number of missed frames. When this counter overflows, it sets the MFCO bit (CSR4: Features Control Register, bit [9]). Cleared by STOP bit (bit [2] of CSR0: Status Register) =1.

CSR114: Receive Collision Count Register

Table 2-75. CSR114: Receive Collision Count Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	72h															
FIELD	///								RXCC							
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2-76. CSR114: Receive Collision Count Register Definitions

Bits	Field Name	Description
15:8	///	**Reserved** Read as 0.
7:0	RXCC	**Number of Regular and Late Receive Frames** Counts the number of regular and late receive collisions. If this counter overflows, it sets the RCVCCO bit (CSR4: Features Control Register, bit [5]). Cleared by STOP bit (bit [2] of CSR0: Status Register) =1.

MII Port Register

The MIIP register is accessed directly at offset 0x18. Bits MDC and MDO of this register drive the MDC and MDIO signals of the MII bus. A sequence of writes to bits MDC, MDI, and MDOE generates the MII management frame.

Table 2-77. MIIP: MII Management Register

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET	0x18															
FIELD	FDEN	///						MDI	MDOE	///					MDC	MD0
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	R	R	R	R	R	R	R	RW	R	R	R	R	R	RW	RW

Table 2-78. MIIP: MII Management Register Definitions

Bits	Field Name	Description
15	FDEN	**Full-/Half-Duplex Mode** 1 = set Ethernet controller to full-duplex mode. 0 = set Ethernet controller to half-duplex mode. This bit is used internally by the Ethernet controller; it is not part of MII management interface. The value written to FDEN must match the mode of the external PHY. FDEN is cleared by RESET.
14:9	///	**Reserved** Read as 0.
8	MDI	**Logical Value on MII MDIO Signal** This Read Only bit indicates the actual logical value on MII MDIO bidirectional signal. During a Management Read Frame, this bit reads serial data sent from the PHY to the Ethernet controller.
7	MDOE	**Output-Enable Pin Driving MDIO** Controls the Output-Enable Pin of I/O buffer driving MDIO signal. 1 = Ethernet controller drives MDIO signal. 0 = Ethernet controller does not drive MDIO (MDIO is Z or driven by PHY). If MDOE=1 and MDI reads different value than MDO, there is a contention on MDIO signal (programming sequence error). MDOE is cleared by RESET.
6:2	///	**Reserved** Read as 0.
1	MDC	**Clock Signal on MII Management Interface** Drives the clock signal (MDC) of the MII management interface. Every transition of this bit causes a transition of the MDC signal. The MII MDC clock is not required to be periodic. A value written to the MDC bit should remain unchanged for at least 160 ns (8 CPU clocks @ CLK=50 MHz) and a minimum period of 400 ns must be ensured (20 CPU clocks @ CLK=50 MHz). MDC can be 0 or 1 for an unlimited time. MDC is cleared by RESET.
0	MDO	**MDIO Signal on MII Interfaces** If MDOE=1, MDO drives the data signal (MDIO) of the MII interfaces. MDO must not be changed when a 1 is written to the MDC bit. Cleared by RESET.

Timing

Internally, the Ethernet controller boasts a fully synchronous design, with virtually all timing parameters relative to the rising edge of the respective clock signal. Many signals are stable one or more clock cycles before they are sampled by the next stage of logic. This is especially true of signals that traverse clock domains.

TX Timing

Figure 2-5 shows the Ethernet transmit signal- timing relationship at the MII I/Os. The signals MI_COL_A and MI_CRS_A are asynchronous to CLK_TX and to any other clock. (Figure 2-5 does not show inputs to the Ethernet controller.)

Figure 2-5. MII Transmit Timing

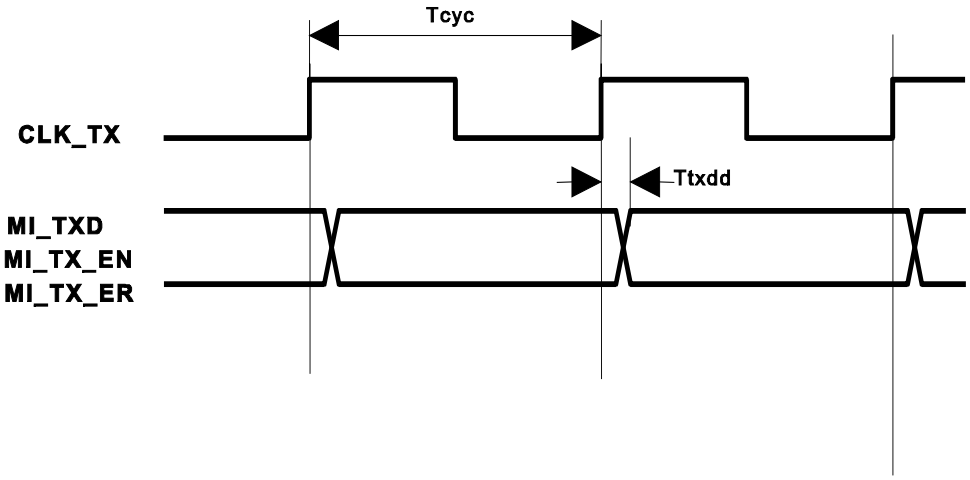


Table 2-79. TX Timing

Parameter	Description	Max (ns)	Typ (ns)	Min (ns)
Tcyc	CLK_TX period (100/10 Mb/s)		40/400	
Ttxdd	CLK_TX to TXD, TX_EN, TX_ER delay	25	10	0

RX Timing

Figure 2-6 shows the Ethernet receive signal-timing relationship at the MII I/Os.

Figure 2-6. MII Receive Timing

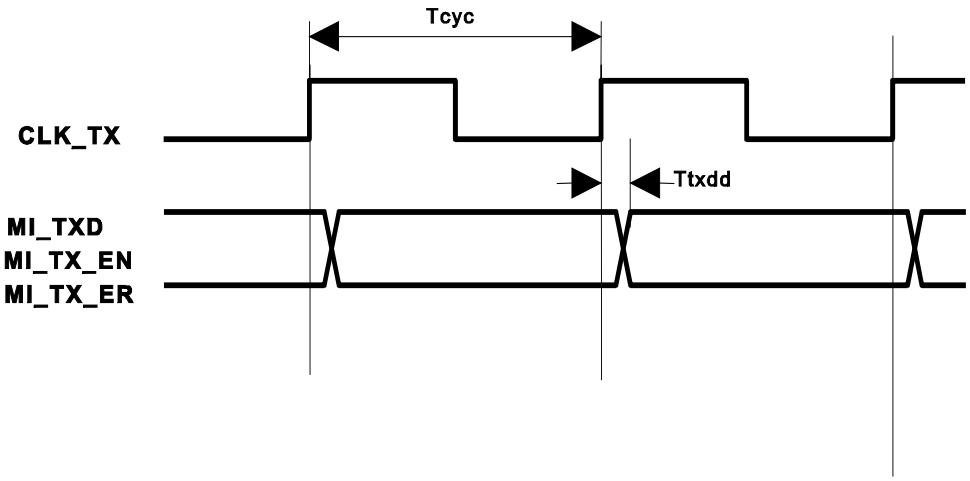


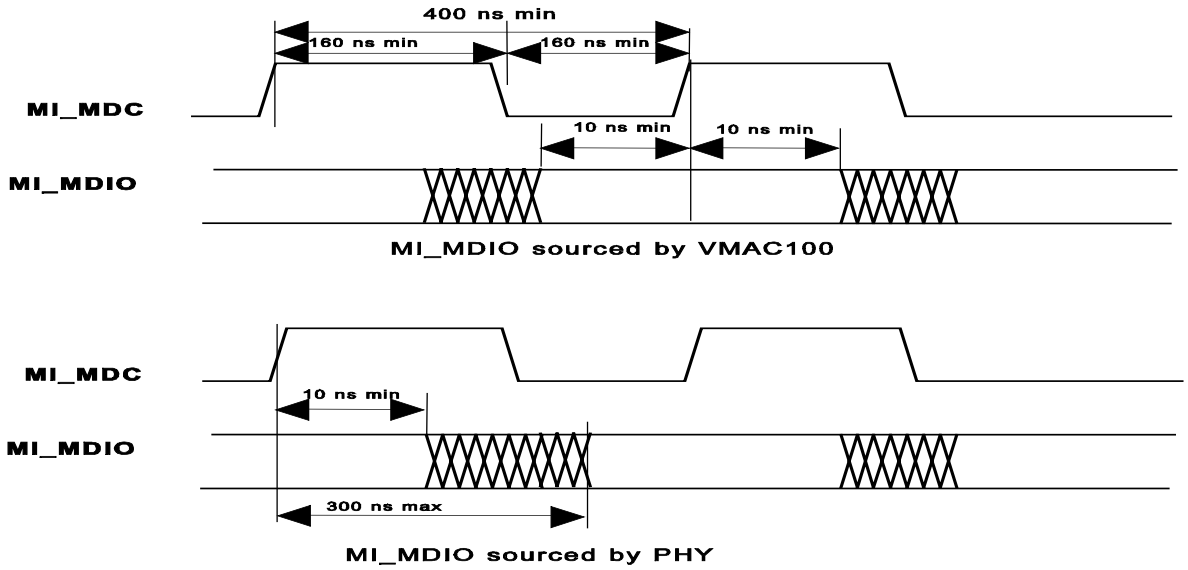
Table 2-80. RX Timing

Parameter	Description	Max (ns)	Typ (ns)	Min (ns)
Tcyc	CLK_RX period		40/400	
Trxdsu	MI_RXD, MI_RX_DV, MI_RX_ER setup time	10		
Trxdh	MI_RXD, MI_RX_DV, MI_RX_ER hold time	10		

MII Management Interface Timing

Figure 2-7 shows the MII management signal- timing relationship at MII I/Os.

Figure 2-7. MII Management Timing

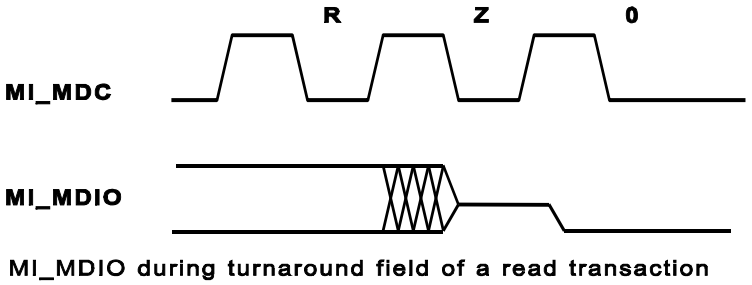


To avoid contention during a read transaction, the turnaround time is a 2-bit time spacing between the Register Address field and the Data field of a management frame. Both the Ethernet controller (STA) and the PHY remain in a high-impedance state for the first bit time of the turnaround. The PHY drives a zero bit during the second bit time of the turnaround of a read transaction.

During a write transaction, the Ethernet controller drives a one bit for the first bit time of the turnaround and a zero bit for the second bit time of the turnaround.

Figure 5-8 shows the behavior of the MI_MDIO signal during the turnaround field of a read transaction.

Figure 2-8. MII MDIO Behavior



3: Ethernet PHY

This chapter describes the DSTni Ethernet Physical Layer Device (PHY). Topics include:

For information about the DSTni Ethernet controller, see Ethernet Controllers on page 10.

Overview

The DSTni Ethernet PHY provides a complete Ethernet physical-layer transceiver core to interface analog Ethernet-compatible signals to an Ethernet MAC layer or switch controller. The PHY communicates with the MAC Layer or controller via the standard IEEE 802.3u Media Independent Interface (MII).

The PHY can work in the following environments:

- 10Base-T — the IEEE 802.3 physical layer specification for 10 Mbps Ethernet over two pairs of Category 3, 4, or 5 UTP wire.
- 100Base-TX — the IEEE physical layer specification for 100 Mbps CSMA/CD over two pairs of Category 5 UTP or STP wire.
- 100Base-FX — the IEEE 802.3 physical layer specification for 100 Mbps Ethernet (Fast Ethernet) over two strands of fiber.

The Ethernet PHY fully complies with IEEE 802.3/802.3u standards. It can connect to twisted-pair media via standard magnetics or to fiber via industry-standard fiber PMD modules.

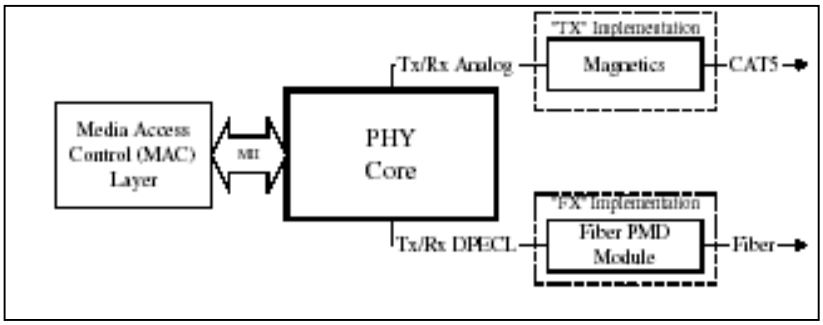
Features

- Fully integrated IEEE 802.3/802.3u compatible10/100Mbps Ethernet PHY
- Integrated DSP adaptive equalizer and Baseline Wander (BLW) correction proven for up to 150m cable with killer packet and added crosstalk
- IEEE 802.3u auto-negotiation and parallel detect capability for automatic speed and duplex selection
- Special power-down modes and management features available
- Half- and full-duplex capability in both 10 and 100Mbps operation
- Automatic polarity correction for 10Base-T
- Extended cable length option for10Base-T
- 100Base-FX support
- Link, Activity, Duplex, and Speed LED outputs
- Serial Management Interface (SMI)
- 25 MHz clock input
- Extensive added features for testability, configuration, and control of options.

Block Diagram

Figure 3-1 shows a block diagram of the DSTni Ethernet PHY.

Figure 3-1. Ethernet PHY Block Diagram



Theory of Operation

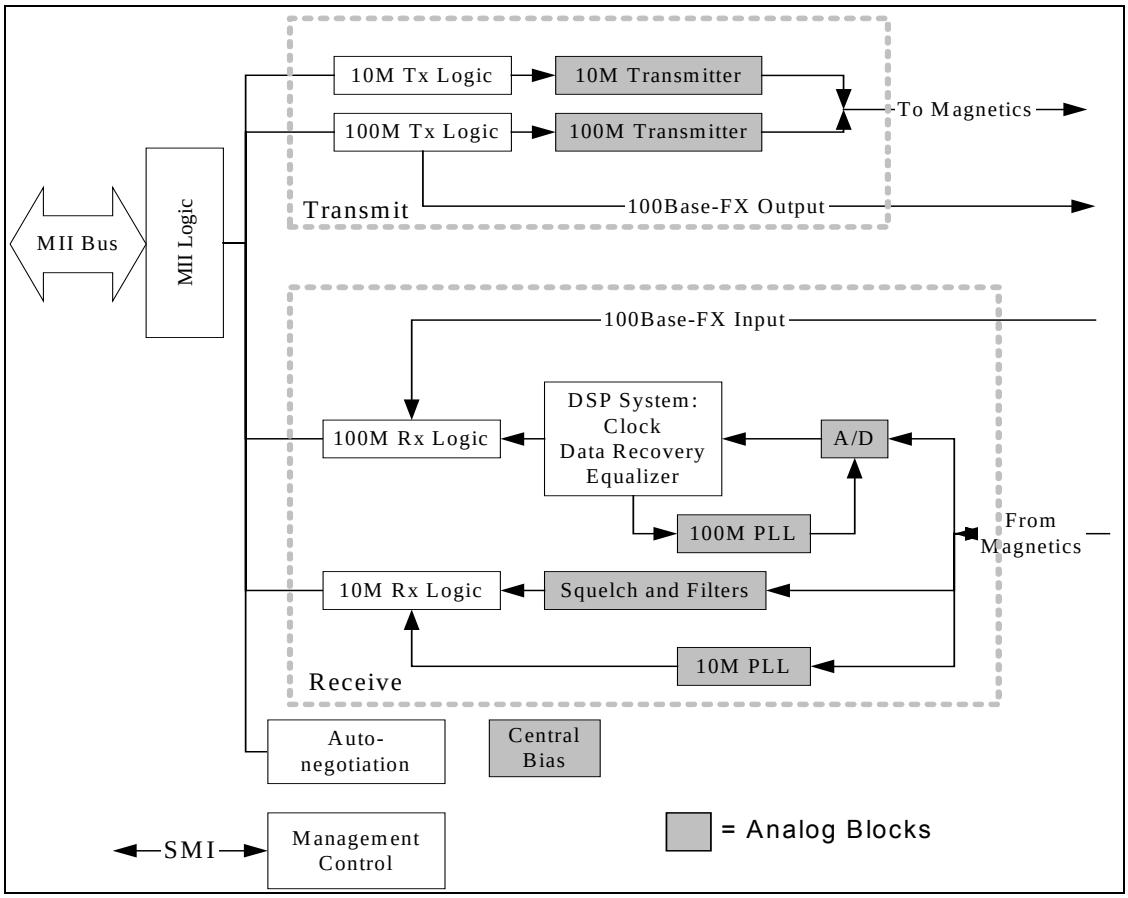
Top-Level Functional Architecture

The core PHY can be organized into the following functional categories:

- 100Base-TX and 100Base-FX transmit and receive
- 10Base-T transmit and receive
- MII interface to the controller
- Auto-negotiation to automatically determine the best speed and duplex possible
- Management Control to read status registers and write control registers

Figure 3-2 shows an overview of the DSTni Ethernet architecture.

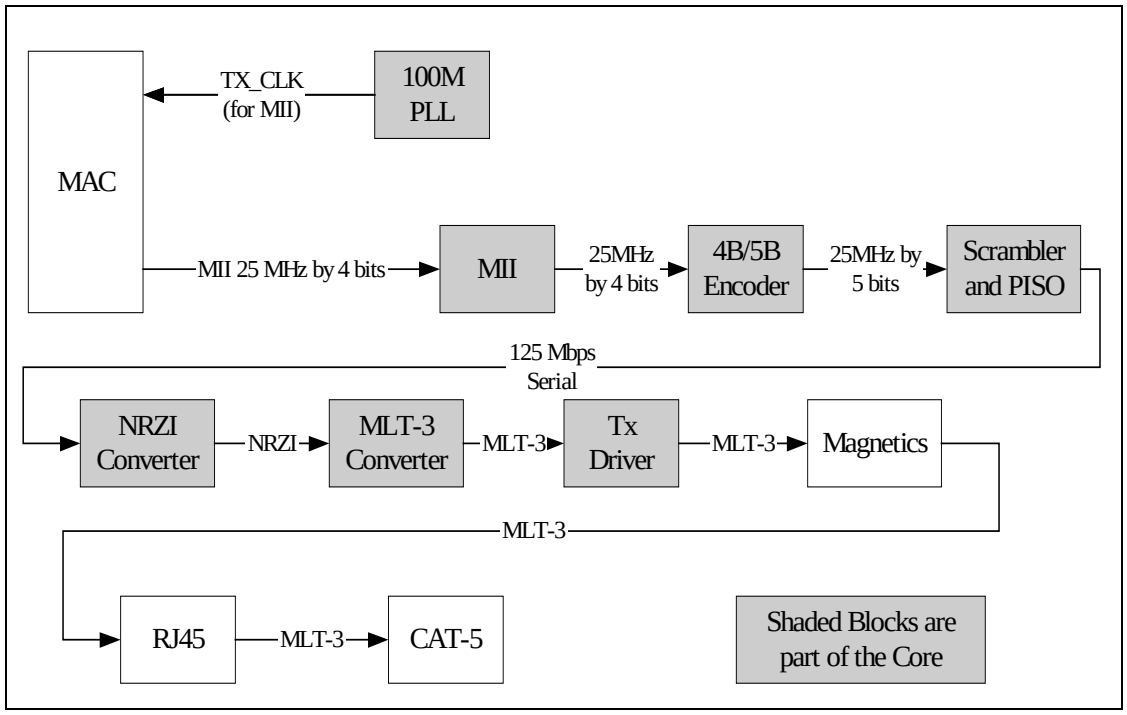
Figure 3-2. Ethernet Architecture Overview



100Base-TX Transmit Path

Figure 3-3 shows the data path of the 100Base-TX. Following the figure is an explanation of each major block shown in the figure.

Figure 3-3. 100Base-TX Data Path



100M Transmit Data Across the MII/RMII

The MAC controller drives the transmit data onto the TXD bus and asserts TX_EN to indicate valid data. The data is latched by the PHY's MII block on the rising edge of TX_CLK. The data is in the form of 4-bit-wide 25 MHz data when the device is in MII mode.

4B/5B Encoding

The transmit data passes from the MII block to the 4B/5B encoder. This block encodes the data from 4-bit nibbles to 5-bit symbols, called “code-groups” by the IEEE 802.3 committee (see Table 3-1).

The 4-bit data maps onto 16 of the 32 possible code-groups. The remaining 16 code-groups are either used for control information or not valid.

- The first 16 code-groups are referred to as hexadecimal numbers 0 through F.
- The remaining code-groups are given other alphabetical designators and shown in documents with slashes on either side. For example, an IDLE code-group is /I/, while a transmit error code-group is /H/.

To bypass the encoding process, clear bit [6] of PHY Register 31 (Table 3-39. PHY Register 31 — PHY Special Control/Status on page 87). Bypassing encoding sets the 5th transmit data bit equivalent to TX_ER.

Table 3-1. 4B/5B Code Table

Code Group	Symbol	Receiver Interpretation	Transmitter Interpretation
11110	0	0 0000 DATA	0 0000 DATA
01001	1	1 0001	1 0001
10100	2	2 0010	2 0010
10101	3	3 0011	3 0011
01010	4	4 0100	4 0100
01011	5	5 0101	5 0101
01110	6	6 0110	6 0110
01111	7	7 0111	7 0111
10010	8	8 1000	8 1000
10011	9	9 1001	9 1001
10110	A	A 1010	A 1010
10111	B	B 1011	B 1011
11010	C	C 1100	C 1100
11011	D	D 1101	D 1101
11100	E	E 1110	E 1110
11101	F	F 1111	F 1111
11111	I	IDLE	Sent after /T/R until TX_EN
11000	J	First nibble of SOP, 0101 if following IDLE, else RX_ER	Sent for rising TX_EN
10001	K	Second nibble of SOP, 0101 if following J, else RX_ER	Sent for rising TX_EN
01101	T	First nibble of EOP, CRS if followed by R, else RX_ER	Sent for falling TX_EN
00111	R	Second nibble of EOP, CRS if after T, else RX_ER	Sent for falling TX_EN
00100	H	Transmit Error Symbol	Sent for rising TX_ER
00110	V	INVALID, RX_ER if during RX_DV	INVALID
11001	V	INVALID, RX_ER if during RX_DV	INVALID
00000	V	INVALID, RX_ER if during RX_DV	INVALID
00001	V	INVALID, RX_ER if during RX_DV	INVALID
00010	V	INVALID, RX_ER if during RX_DV	INVALID
00011	V	INVALID, RX_ER if during RX_DV	INVALID
00101	V	INVALID, RX_ER if during RX_DV	INVALID
01000	V	INVALID, RX_ER if during RX_DV	INVALID
01100	V	INVALID, RX_ER if during RX_DV	INVALID
10000	V	INVALID, RX_ER if during RX_DV	INVALID

Scrambling

Repeated data patterns, especially the IDLE code-group, can have power spectral densities with large narrow-band peaks. Scrambling the data helps eliminate these peaks and distribute the signal power more uniformly over the entire channel bandwidth. FCC regulations require this uniform spectral density to prevent the physical wiring from radiating excessive EMI.

The scrambler seed is generated from the PHY address, PHYAD[4:0]. This ensures that multiple PHYs in an application (for example, a repeater) have different scrambler sequences.

The scrambler also performs the Parallel In Serial Out (PISO) data conversion.

NRZI and MLT3 Encoding

The scrambler block passes the 5-bit-wide parallel data to the NRZI converter where it becomes a serial 125MHz NRZI data stream. The NRZI is encoded to MLT-3. MLT3 is a tri-level code:

- A change in the logic level represents a data bit 1.
- The logical output remaining at the same level represents a data bit 0.

100M Transmit Driver

The MLT3 data passes to the analog transmitter, which launches the differential MLT-3 signal (on outputs TXP and TXN) to the twisted-pair media via a 1:1 ratio isolation transformer. The 10Base-T and 100Base-TX signals pass through the same transformer, so that common "magnetics" can be used for both. The transmitter drives into the 100W impedance of the CAT-5 cable. Cable termination and impedance matching require external components.

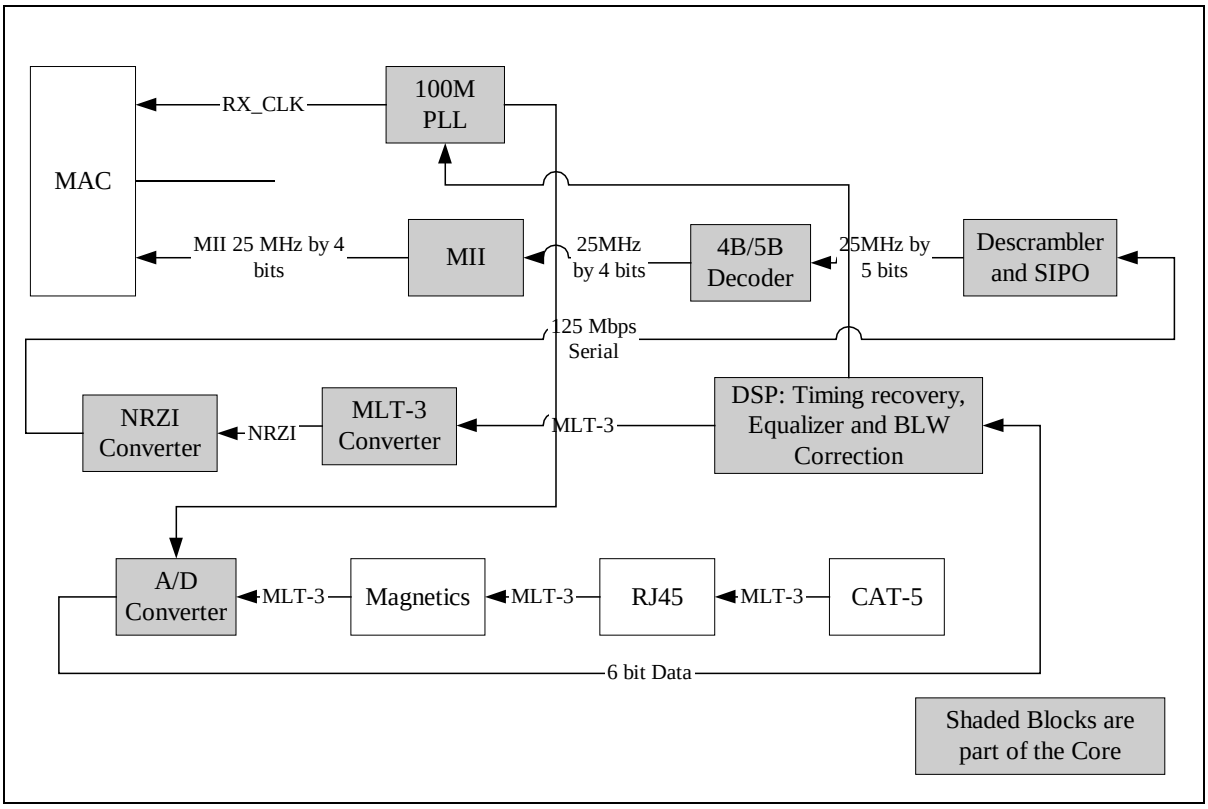
100M Phase Lock Loop

The 100M Phase Lock Loop (PLL) locks onto 50 MHz clock and generates the 125 MHz clock used to drive the 125 MHz logic and the 100Base-Tx transmitter.

100Base-TX Receive

Figure 3-4 shows the receive data path. Following the figure is an explanation of each major block shown in the figure.

Figure 3-4. PHY Receive Data Path



100M Receive Input

A 1:1 ratio transformer feeds the MLT-3 from the cable into the core PHY on inputs RXP and RXN. The ADC samples the incoming differential signal at a rate of 125M samples per second. Using a 64 level quantizer, the ADC generates 6 digital bits to represent each sample. The DSP adjusts the gain of the ADC according to the observed signal levels, allowing the full dynamic range of the ADC to be used.

Equalizer, Baseline Wander Correction, and Clock and Data Recovery

The 6 bits from the ADC are fed to the DSP block. The equalizer functions in the DSP section compensate for phase and amplitude distortion in the physical channel (magnetics, connectors, and CAT 5 cable). The signal can be restored for any good-quality CAT 5 cable from 1m to 150m in length.

If the DC content of the signal causes the low frequency components to fall below the low frequency pole of the isolation transformer, the transformer's droop characteristics become significant, causing a Baseline Wander (BLW) on the received signal. To prevent the received data from being corrupted, the core PHY corrects for BLW and can receive the ANSI X3.263-1995 FDDI TP-PMD-defined "killer packet" with no bit errors.

The 100M PLL generates multiple phases of the 125 MHz clock. A multiplexer, controlled by the timing unit of the DSP, selects the phase nearest to that of the received signal clock and uses it as the received recovered clock. This clock is used to extract the serial data from the received signal.

NRZI and MLT-3 Decoding

The DSP generates the MLT-3 recovered levels that are fed to the MLT-3 converter. The MLT-3 then converts to an NRZI data stream.

Descrambling

The de-scrambler performs an inverse function to the scrambler in the transmitter. It also performs the Serial In Parallel Out (SIPO) conversion of the data. To bypass the de-scrambler, set bit [0] in Register 31 (see Table 3-40. PHY Register 31 — PHY Special Control/Status Definitions on page 87).

Alignment

The de-scrambled signal then aligns into 5-bit code-groups by recognizing the /J/K/ Start-of-Stream Delimiter pair at the start of a packet. Once the code word alignment is determined, it is stored and utilized until the next start of frame is received.

5B/4B Decoding

The 5-bit code-groups are decoded to 4-bit data by performing the translation according to the 4B/5B table (see Table 3-1 on page 58).

The translated data is presented on the RXD[3:0] signal lines. The SSD, /J/K is translated to "0101 0101" to restore the first 2 nibbles of the MAC preamble. It is also decoded and used to assert the RX_DV signal to tell the controller that valid data is available on the RXD bus. Successive valid code-groups are translated to data nibbles. The End-of-Stream Delimiter (ESD), /T/R, or at least 2 consecutive IDLE code-groups are used to terminate the carrier sense and RX_DV signals at the end of the frame. They do not translate to data.

To bypass the decoding process, clear bit [6] of PHY Register 31 (see Table 3-39. PHY Register 31 — PHY Special Control/Status on page 87). If decoding is bypassed, the 5th receive data bit is driven out on RX_ER.

Receive Data Valid Signal

The Receive Data Valid signal (RX_DV) indicates that recovered and decoded nibbles are being presented on the RXD[3:0] outputs synchronous to RX_CLK. RX_DV becomes active after the /J/K/ delimiter is recognized and RXD is aligned to nibble boundaries. It remains active until the /T/R/ delimiter is recognized, the link test indicates failure, or the SIGDET becomes false.

RX_DV is asserted when the first nibble of translated /J/K/ is ready for transfer over the MII.

Receiver Errors

During a frame, the PHY considers unexpected code-groups as receive errors. Expected code groups are the DATA set (0 through F), and the /T/R/ (ESD) symbol pair. When a receive error occurs, the RX_ER signal is asserted and arbitrary data is driven onto the RXD lines. If an error is detected during the time that the /J/K/ delimiter is being decoded (bad SSD error), RX_ER is asserted true, and the value “1110” is driven onto the RXD lines.

***Note**: The Valid Data signal is not asserted when the bad SSD error occurs.*

100M Receive Data Across the MII

The 4-bit data nibbles are sent to the MII block. In MII mode, these data nibbles are clocked to the controller at a rate of 25 MHz. The controller samples the data on the rising edge of RX_CLK. To ensure that the setup and hold requirements are met, the nibbles are clocked out of the PHY on the falling edge of RX_CLK. RX_CLK is the 25 MHz output clock for the MII bus. It is recovered from the received data to clock the RXD bus. If there is no received signal, it is derived from the system reference clock, SYS_CLK. When tracking the received data, RX_CLK has a maximum jitter of 0.45ns.

100M FX Logic

The following configurations occur in the PHY during 100Base-FX mode:

- The scrambler and de-scrambler are bypassed.
- All analog circuits, except for the 100M PLL, are powered-down.
- Auto-negotiation is disabled.
- The DSP and the 10Base-T logic are powered-down.
- When FSD is deactivated, the PECL IO are powered down..

100M FX Transmitter

The 100Base-FX transmitter shares logic with the 100Base-TX transmitter. Data passes across the MII and is encoded; however, the 5-bit symbols are not scrambled, since fiber media does not suffer from emission anomalies. The 5-bit symbols convert to a serial NRZI data stream and appear at the FTX signal.

100M FX Receiver

The 100M FX receiver shares much of the logic with the 100Base-TX receiver. The FRX input signal is an NRZI data stream. By detecting the edges of the data signal, the receiver selects one of the 100M PLL phases and use it as the sample clock for the data. The sampled data is converted to 25 MHz, 5-bit-wide code-groups. The code-groups are aligned and decoded before being passed across the MII.

100M FX Signal Detect

The 100M FX signal detect (FSD) is an input signal to the core from the PMD FX transceiver. FSD is asserted to indicate a valid analog FX signal on the fiber.

100M FX Far End Fault Indication

To ensure correct operation, disable auto-negotiation during fiber mode. Disabling auto-negotiation also disables the mechanism for 2 link partners to communicate connectivity information. However, you can use Far End Fault Indication to pass this information.

A Far End Fault is a fault that is visible to one link partner. For example, a local node is aware that the receive link is up when the signal detect is active. However, it is unaware of the status of its transmit link. If the transmit link is down, the remote partner sees no signal detect. In this case, the remote partner can modify the IDLE code it transmits into a sequence of 16 IDLE code-groups, followed by a data 0 code-group. This is referred to as the Far End Fault IDLE (FEFI) pattern. If FEFI is detected on the receive stream, the link status goes to the inactive state and the PHY sends IDLE code-groups (not FEFI).For more information about FEFI, refer to Section 24.3.2.1 in the IEEE 802.3u-1995 standard.

To disable the FEFI feature, clear bit 27.5 (see Table 3-34. PHY Register 27 — Special Control/Status Indications Definitions on page 84). By default, this bit is set to 1 when the FXMODE is enabled.

10Base-T Transmit

Data to be transmitted comes from the MAC layer controller. The 10Base-T transmitter gets 4-bit nibbles across the MII at 2.5 MHz and converts them to a 10 Mbps serial data stream. The data stream is Manchester-encoded and sent to the analog transmitter, which drives a signal onto the twisted-pair via the magnetics.

The 10M transmitter uses the following blocks:

- MII (digital)
- TX 10M (digital)
- 10M Transmitter (analog)
- 10M PLL (analog)

10M Transmit Data Across the MII

The MAC controller drives the transmit data onto the TXD bus. When the controller drives TX_EN HIGH to indicate valid data, the MII block latches the data on the rising edge of TX_CLK. The data is 4-bit-wide 2.5 MHz data.

Manchester Encoding

The 4-bit-wide data is sent to the TX10M block. The nibbles convert to a 10 Mbps serial NRZI data stream. The 10M PLL locks onto the external clock or internal oscillator and produces a 20 MHz clock. This clock is used to Manchester-encode the NRZ data stream.

When no data is transmitted (TX_EN is LOW), the TX10M block outputs Normal Link Pulses (NLPs) to maintain communications with the remote link partner.

10M Transmit Drivers

The Manchester-encoded data is sent to the analog transmitter, where it is shaped and filtered. It is then driven out as a differential signal across the TXP and TXN outputs.

10Base-T Receive

The 10Base-T receiver gets the Manchester-encoded analog signal from the cable via the magnetics. It recovers the receive clock from the signal and uses the receive clock to recover the NRZI data stream. This 10M serial data converts to 4-bit data nibbles, which pass to the controller across the MII at a rate of 2.5 MHz.

This 10M receiver uses the following blocks:

- Filter and SQUELCH (analog)
- 10M PLL (analog)
- RX 10M (digital)
- MII (digital)

10M Receive Input and Squelch

The Manchester signal from the cable is:

- Filtered to reduce any out-of-band noise.
- Fed into the core PHY (on inputs RXP and RXN) via 1:1 ratio magnetics.
- Passed through a SQUELCH circuit. The SQUELCH is a set of amplitude and timing comparators that normally reject differential voltage levels below 300 mV, and detect and recognize differential voltages above 585 mV.

Manchester Decoding

The output of the SQUELCH goes to the RX10M block, where it is validated as Manchester-encoded data. The polarity of the signal is also checked. If the polarity is reversed (local RXP is connected to RXN of the remote partner and vice versa), it is identified and corrected. The reversed condition is indicated by the flag “XPOL,” bit 27.4 (see Table 3-33. PHY Register 27 — Special Control/Status Indications on page 84).

The 10M PLL is locked onto the received Manchester signal and from this, generates the received 20 MHz clock. Using this clock, the Manchester-encoded data is extracted and converts to a 10 MHz NRZI data stream. It then converts from serial to 4-bit-wide parallel data.

The RX10M block also detects valid 10Base-T IDLE signals — Normal Link Pulses (NLPs) — to maintain the link.

10M Receive Data across the MII

The 4-bit data nibbles are sent to the MII block. In MII mode, these data nibbles are valid on the rising edge of the 2.5 MHz RX_CLK.

MII Block

The MII block handles communications with the controller. Special sets of handshake signals indicate that valid received/transmitted data is present on the 4-bit receive/transmit bus.

The MII includes the interface signals:

The PHY drives the transmit clock, TX_CLK, to the controller. The controller, in turn:

- Synchronizes the transmit data to the rising edge of TX_CLK.
- Drives TX_EN HIGH to indicate valid transmit data.
- Drives TX_ER HIGH if a transmit error is detected.
-

The PHY drives both the receive data, RXD, and the RX_CLK signal. The controller clocks in the receive data on the rising edge of RX_CLK when the PHY drives RX_DV HIGH. The PHY drives RX_ER HIGH if a receive error is detected.

Auto-negotiation

The auto-negotiation function automatically configures PHYs with multiple speed, duplex, and protocol capabilities based on the capabilities of the link partners. Auto-negotiation achieves this by defining a way to exchange configuration information between two ends of a link segment and automatically select the highest performance mode supported by both sides. Auto-negotiation is fully defined in clause 28 of the IEEE 802.3 specification.

After auto-negotiation completes, information about the resolved link passes to the controller via the Serial Management Interface (SMI). The results of the negotiation process are reflected in the Speed Indication bits in Register 31 (see Table 3-39. PHY Register 31 — PHY Special Control/Status on page 87) and the Register 5, Link Partner Ability Register (see Table 3-14. PHY Register 5 — Auto-negotiation Link Partner Capability on page 75).

The auto-negotiation protocol is a purely physical-layer activity and operates independently of the MAC controller.

The advertised capabilities of the PHY are stored in Register 4 of the SMI registers. The default advertised by the core PHY is determined by user-defined on-chip signal options.

The following blocks are activated during an auto-negotiation session:

- Auto-negotiation (digital)
- 100M ADC (analog)
- 100M PLL (analog)
- 100M equalizer/BLW/clock recovery (DSP)
- 10M SQUELCH (analog)
- 10M PLL (analog)
- 10M Transmitter (analog)

When enabled, auto-negotiation starts when one of the following events occurs:

- Hardware reset
- Software reset
- Power-down reset
- Link status down
- Setting bit [9] of Register 0 HIGH to restart auto-negotiation (see Table 3-4. PHY Register 0 — Basic Control on page 70)

When the PHY detects one of these events, it begins auto-negotiation by transmitting bursts of Fast Link Pulses (FLP). These pulses are bursts of link pulses from the 10M transmitter. They are shaped as Normal Link Pulses and can pass uncorrupted down CAT-3 or CAT-5 cable. A Fast Link Pulse Burst consists of 33 pulse positions. The 17 odd-numbered pulse positions contain a link pulse and represent clock information. A pulse present in one of the 16 even-numbered pulse positions represents a data logical one. The absence of a pulse in the even-numbered pulse position represents data logical zero.

Link Code Words are encoded in the data fields of the FLP Bursts. These are defined fully in IEEE 802.3 clause 28. In summary, the core PHY advertises 802.3 compliance in its selector field (the first 5 bits of the Link Code Word). Its technological ability is advertised according to the bits set in Register 4 of the SMI registers.

There are four technological abilities. In order of priority, they are:

- 100M full-duplex (highest priority)
- 100m half-duplex
- 10m full-duplex
- 10m half-duplex

If the full capabilities of the core PHY are advertised (100M, full-duplex), and if the link partner can operate at 10M and 100M, auto-negotiation selects 100M as the highest performance mode. If the link partner can operate in half- and full-duplex modes, auto-negotiation selects full-duplex as the highest performance operation.

After the core PHY matches capabilities, the link code words repeat with the acknowledge bit set. Any difference in the main content of the link code words restarts the auto-negotiation process. Auto-negotiation also restarts if the full auto-negotiation sequence does not complete.

After reset completes, the logic levels latched on the MODE[2:0] bus determine the capabilities that the core PHY advertises during auto-negotiation. This bus can disable auto-negotiation on power-up.

Writing Register 4 bits [8:5] allows software control of the capabilities advertised by the core PHY (Table 3-12. PHY Register 4 — Auto-negotiation Advertisement on page 74). Writing Register 4 does not automatically restart auto-negotiation. Bit [9] of Register 0 must be set before the new abilities are advertised (see Table 3-4. PHY Register 0 — Basic Control on page 70). You can also disable auto-negotiation via software by clearing bit [12] of Register 0.

The Ethernet PHY does not support the Next Page capability.

Parallel Detection

If the Ethernet PHY connects to a device without the auto-negotiation capability (i.e., no FLPs are detected), it can “Parallel Detect” either the 100M stream or the 10M Normal Link Pulses and form a half-duplex link at the appropriate speed. This feature ensures interoperability with legacy link partners.

If a link is formed via parallel detection, bit [0] of Register 6 clears, indicating that the link partner cannot auto-negotiate (see Table 3-16. PHY Register 6 — Auto-negotiation Expansion on page 76). The controller has access to this information via the management interface. If a fault occurs during parallel detection, bit 6.4 is set.

Register 5 stores the Link Partner Ability information, which is coded in the received FLPs (see Table 3-14. PHY Register 5 — Auto-negotiation Link Partner Capability on page 75). If the Link Partner cannot auto-negotiate, Register 5 is updated after parallel detection completes to reflect the link partner’s speed capability.

Restarting Auto-negotiation

Auto-negotiation restarts at any time by setting bit [9] of Register 0 (see Table 3-4. PHY Register 0 — Basic Control on page 70). It also re-starts if a signal loss causes the link to break at any time. This can occur if a cable breaks or if the link partner’s signal is interrupted. Auto-negotiation resumes in an attempt to determine the new link configuration.

If the management entity re-starts auto-negotiation by writing to bit [9] of Register 0:

- The Ethernet PHY stops transmitting data packets and link pulse IDLE signals.
- The break_link_timer times-out after approximately 1200 ms.
- The loss of link restarts auto-negotiation.

The link partner also drops the link due to lack of a received signal, so it too resumes auto-negotiation.

Disabling Auto-negotiation

To disable auto-negotiation, set bit [12] of Register 0 to zero (see Table 3-4. PHY Register 0 — Basic Control on page 70). Disabling auto-negotiation causes the device to force its speed of operation to reflect the speed information in bit [13] of Register 0 and the duplex information in bit [8] of Register 0. The speed and duplex bits in Register 0 are ignored when auto-negotiation is enabled.

Half-Duplex vs. Full-Duplex

Half-duplex operation uses the Carrier Sense Multiple Access / Collision Detect (CSMA/CD) protocol to handle network traffic and collisions. In this mode, the carrier sense signal, CRS and responds to both transmit and receive activity.

Full-duplex mode supports simultaneous transmit and receive data. CRS is redefined to respond only to receive activity. The CSMA/CD protocol does not apply and collision detection is disabled.

PHY Management Control

The Management Control module includes the following blocks:

- Serial Management Interface (SMI)
- Management Registers Set
- Interrupt

Serial Management Interface

The Serial Management Interface (SMI) controls the Ethernet PHY and obtains its status. The SMI corresponds to the MII spec for 100Base-TX (Clause 22), and supports Registers 0 through 6. Additional “vendor-specific” registers are implemented within the range of 16 to 31, as allowed by the specification.

At the system level, there are two signals: MDIO and MDC.

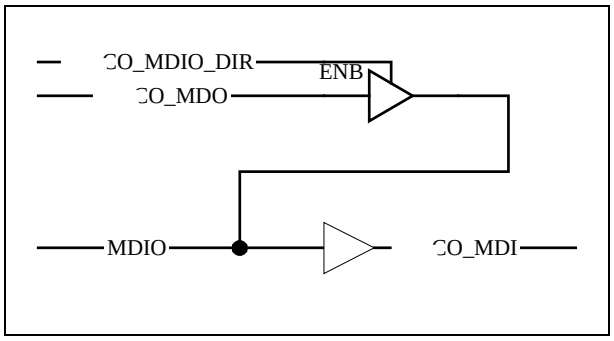
- MDIO is bidirectional open-drain.
- MDC is the clock

A core has no concept of bdirectional signals, so the MDIO signal is implemented as three signals:

- MDIO_DIR
- MDO
- MDI

Figure 3-5 shows the relationship among these signals. In Figure 6-7, the drivers shown should be open-drain.

Figure 3-5. Deriving the MDIO Signal from the Core Signals



The MDC signal is a periodic clock that the station management controller (SMC) provides. The MDI signal receives serial data (commands) from the controller SMC. The MDO sends serial data (status) to the SMC.

The minimum time between edges of the MDC is 160 ns. There is no maximum time between edges. The minimum cycle time (time between two consecutive rising or two consecutive falling edges) is 400 nS. These modest timing requirements allow the input/output (I/O) port of a microcontroller to drive this interface easily.

The data on the MDO and MDI lines is latched on the rising edge of the MDC. Figure 3-5 and Figure 3-6 show the frame structure and timing of the data.

The timing relationships of the MDIO signals are further described in Table 3-2.

Table 3-2. Timing Relationships of the MDIO Signals

Relationship	Minimum	Maximum
Delay from rising edge of MDC to next change in MDIO_DIR	25 ns	125 ns
Delay from rising edge of MDC to valid data on MDO	5 ns	125 ns
Setup time of MDI before rising edge of MDC	10 ns	
Hold time of MDI after rising edge of MDC	10 ns	

Figure 3-6. MDIO Timing and Frame Structure – READ cycle

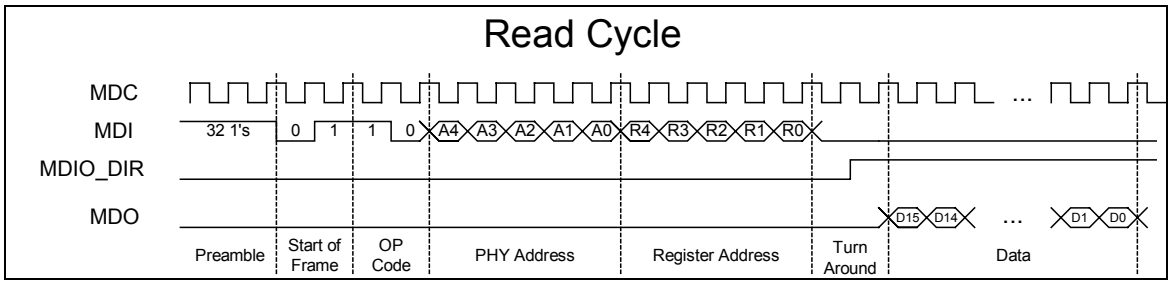
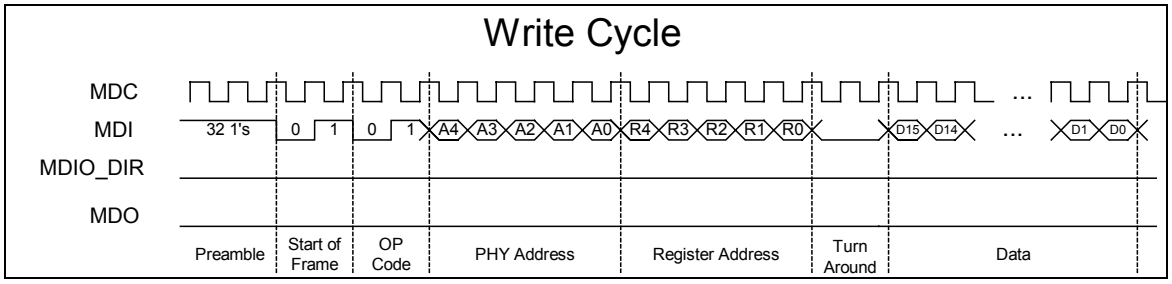


Figure 3-7. MDIO Timing and Frame Structure – WRITE cycle



SMI Register Summary

Table 3-3. SMI Register Summary

Register Description	Group	Page
PHY Register 0 — Basic Control	Basic	70
PHY Register 1 — Basic Status	Basic	72
PHY Register 2 — PHY Identifier 1	Extended	73
PHY Register 3 — PHY Identifier 2	Extended	74
PHY Register 4 — Auto-negotiation Advertisement	Extended	74
PHY Register 5 — Auto-negotiation Link Partner Capability	Extended	75
PHY Register 6 — Auto-negotiation Expansion	Extended	76
PHY Register 16 — Silicon Revision	Vendor Specific	77
PHY Register 17 — Mode/Control Status	Vendor Specific	78
PHY Register 18 — Special Modes	Vendor Specific	79
PHY Register 20 — TSTCNTL	Vendor Specific	82
PHY Register 21 — TSTREAD1	Vendor Specific	82
PHY Register 22 — TSTREAD2	Vendor Specific	83
PHY Register 23 — TSTWRITE	Vendor Specific	83
PHY Register 27 — Special Control/Status Indications	Vendor Specific	84
PHY Register 29 — Interrupt Source Flags	Vendor Specific	85
PHY Register 30 — Interrupt Mask	Vendor Specific	86
PHY Register 31 — PHY Special Control/Status	Vendor Specific	87

Table 3-4. PHY Register 0 — Basic Control

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	RESET	LOOPBACK	SPEED SELECT	AUTO-NEGOTIATION ENABLE	POWER DOWN	ISOLATE	RESTART AUTO-NEGOTIATION	DUPLEX MODE	COLLISION TEST	SPEED SELECT (MSB)	///					
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R

Table 3-5. PHY Register 0 — Basic Control Definitions

Bits	Field Name	Description
15	RESET	**Reset** 1 = software reset. 0 = normal operation (*default*). This bit is self-clearing.
14	LOOPBACK	**Loopback** 1 = loopback mode. 0 = normal operation (*default*).
13	SPEED SELECT	**Speed Select** 1 = 100 Mbps. 0 = 10 Mbps. Ignored if Auto Negotiation is enabled – bit [12] = 1 (*default*).
12	AUTO-NEGOTIATION ENABLE	**Auto-negotiation Enable** 1 = enable auto-negotiate process (overrides bits [13] and [8] of this register). 0 = disable auto-negotiate process (*default*). Set by MODE, bits [2:0].
11	POWER DOWN	**Power Down** 1 = general power-down mode. 0 = normal operation (*default*).
10	ISOLATE	**Electrical Isolation** 1 = PHY is electrically isolated from MII. 0 = normal operation (*default*).
9	RESTART AUTO-NEGOTIATION	**Restart Auto-negotiation** 1 = restart auto-negotiation process. 0 = normal operation (*default*). This bit is self-clearing.
8	DUPLEX MODE	**Duplex Mode** 1 = full-duplex. 0 = half-duplex (*default*). This bit is ignored if auto-negotiation is enabled – bit [12]=1.
7	COLLISION TEST	**Collision Test** 1 = enable collision test. 0 = disable collision test (*default*).
6	SPEED SELECT (MSB)	**Speed Select** The most-significant bit of Speed Select. This bit is always '0' for this PHY.
5:0	///	**Reserved**

Table 3-6. PHY Register 1 — Basic Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	100Base-T4	100Base-TX FULL-DUPLEX	100Base-TX HALF-DUPLEX	10Base-T FULL-DUPLEX	10Base-T HALF-DUPLEX	///					AUTO-NEGOTIATE COMPLETE	REMOTE FAULT	AUTO-NEGOTIATE ABILITY	LINK STATUS	JABBER DETECT	EXTENDED CAPABILITIES
RESET	0	1	1	1	1	0	0	0	0	0	0	0	1	0	0	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-7. PHY Register 1 — Basic Status Definitions

Bits	Field Name	Description
15	100Base-T4	**100Base-T4** 1 = T4 capability. 0 = no T4 capability (*default*).
14	100Base-TX FULL-DUPLEX	**100Base-TX Full-Duplex** 1 = TX with full-duplex (*default*). 0 = no TX full-duplex capability.
13	100Base-TX HALF-DUPLEX	**100Base-TX Half-Duplex** 1 = TX with half-duplex (*default*). 0 = no TX half-duplex capability.
12	10Base-T FULL-DUPLEX	**10Base-T Full-Duplex** 1 = 10 Mbps with full-duplex (*default*). 0 = no 10 Mbps with full-duplex capability.
11	10Base-T HALF-DUPLEX	**10Base-T Half-Duplex** 1 = 10 Mbps with half-duplex (*default*). 0 = no 10 Mbps with half-duplex capability.
10:6	///	**Reserved**
5	AUTO-NEGOTIATE COMPLETE	**Auto-negotiate Complete** 1 = auto-negotiate process completed. 0 = auto-negotiate process not completed (*default*).
4	REMOTE FAULT	**Remote Fault** 1 = remote fault condition detected. 0 = no remote fault (*default*).
3	AUTO-NEGOTIATE ABILITY	**Auto-negotiate Ability** 1 = can perform auto-negotiation (*default*). 0 = unable to perform auto-negotiation.
2	LINK STATUS	**Link Status** 1 = link is up. 0 = link is down (*default*).
1	JABBER DETECT	**Jabber Detect** 1 = jabber condition detected. 0 = no jabber condition detected (*default*)
0	EXTENDED CAPABILITIES	**Extended Capabilities** 1 = support extended capabilities registers (*default*). 0 = do not support extended capabilities registers.

Table 3-8. PHY Register 2 — PHY Identifier 1

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	PHY ID NUMBER															
RESET	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-9. PHY Register 2 — PHY Identifier 1 Definitions

Bits	Field Name	Description
15:0	PHY ID NUMBER	**PHY ID Number** Assigned to bits [18:3] of the Organizationally Unique Identifier (OUI), respectively. Initialization is set by the signal REG2_OUI_IN, bits [15:0].

Table 3-10. PHY Register 3 — PHY Identifier 2

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	PHY ID NUMBER B						MODEL NUMBER						REVISION NUMBER			
RESET	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-11. PHY Register 3 — PHY Identifier 2 Definitions

Bits	Field Name	Description
15:10	PHY ID NUMBER B	**PHY ID Number B** Assigned to bits [24:19] of the (OUI), respectively. Initialization is set by the signal REG3_OUI_IN, bits [15:0].
9:4	MODEL NUMBER	**Model Number** Six-bit manufacturer's model number.
3:0	REVISION NUMBER	**Revision Number** Four-bit manufacturer's revision number.

Table 3-12. PHY Register 4 — Auto-negotiation Advertisement

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	NEXT PAGE	///	REMOTE FAULT	///		FLOW CONTROL	100Base-T4	100Base-TX FULL-DUPLEX	100Base-TX	10Base-T FULL-DUPLEX	10Base-T	SELECTOR FIELD				
RESET	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
RW	R	R	RW	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-13. PHY Register 4 — Auto-negotiation Advertisement Definitions

Bits	Field Name	Description
15	NEXT PAGE	**Next Page** 1 = next page capable. 0 = no next page capability (*default*). DSTni does not support next-page capability.
14	///	**Reserved**
13	REMOTE FAULT	**Remote Fault** 1 = remote fault detected. 0 = remote fault not detected (*default*).
12:11	///	**Reserved**
10	FLOW CONTROL	**Flow Control** 1 = flow control is supported by MAC. 0 = flow control is not supported by MAC (*default*).
9	100Base-T4	**100Base-T4** 1 = T4 capable. 0 = no T4 capability (*default*).
8	100Base-TX FULL-DUPLEX	**100Base-TX Full-Duplex** 1 = TX with full-duplex. 0 = no TX full-duplex capability (*default*).
7	100Base-TX	**100Base-TX** 1 = TX capability. 0 = no TX capability (*default*).
6	10Base-T FULL-DUPLEX	**10Base-T Full-Duplex** 1 = 10 Mbps with full-duplex. 0 = no 10 Mbps with full-duplex capability (*default*).
5	10Base-T	**10Base-T** 1 = 10 Mbps capable (*default*). 0 = no 10 Mbps capability.
4:0	SELECTOR FIELD	**Selector Field** 00001 = IEEE 802.3.

Table 3-14. PHY Register 5 — Auto-negotiation Link Partner Capability

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	NEXT PAGE	ACKNOWLEDGE	REMOTE FAULT	///			100Base-T4	100Base-TX FULL-DUPLEX	100Base-TX	10Base-T FULL-DUPLEX	10Base-T	SELECTOR FIELD				
RESET	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-15. PHY Register 5 — Auto-negotiation Link Partner Capability Definitions

Bits	Field Name	Description
15	NEXT PAGE	**Next Page** 1 = next page capable. 0 = no next page capability (*default*). DSTni does not support next-page capability.
14	ACKNOWLEDGE	**Acknowledge** 1 = link code word received from partner. 0 = link code word not yet received (*default*).
13	REMOTE FAULT	**Remote Fault** 1 = remote fault detected. 0 = remote fault not detected (*default*).
12:10	///	**Reserved**
9	100Base-T4	**100Base-T4** 1 = T4 capable. 0 = no T4 capability (*default*).
8	100Base-TX FULL-DUPLEX	**100Base-TX Half-Duplex** 1 = TX with full-duplex. 0 = no TX full-duplex capability (*default*).
7	100Base-TX	**100Base-TX** 1 = TX with full-duplex (*default*). 0 = no TX full-duplex capability.
6	10Base-T FULL-DUPLEX	**10Base-T Full-Duplex** 1 = 10 Mbps with full-duplex. 0 = no 10 Mbps with full-duplex capability (*default*).
5	10Base-T	**10Base-T** 1 = 10 Mbps capable. 0 = no 10 Mbps capability (*default*).
4:0	SELECTOR FIELD	**Selector Field** [00001] = IEEE 802.3.

Table 3-16. PHY Register 6 — Auto-negotiation Expansion

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///											PARALLEL DETECTION FAULT	LINK PARTNER NEXT PAGE ABLE	NEXT PAGE ABLE	PAGE RECEIVED	LINK PARTNER AUTO-NEGOTIATION ABLE
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-17. PHY Register 6 — Auto-negotiation Expansion Definitions

Bits	Field Name	Description
15:5	///	**Reserved**
4	PARALLEL DETECTION FAULT	**Parallel Detection Fault** 1 = fault detected by parallel detection logic. 0 = no fault detected by parallel detection logic (*default*).
3	LINK PARTNER NEXT PAGE ABLE	**Link Partner Next Page Capability** 1 = link partner has next page ability. 0 = link partner does not have next page ability (*default*).
2	NEXT PAGE ABLE	**Local Device Next Page Capability** 1 = local device has next page ability. 0 = local device does not have next page ability (*default*).
1	PAGE RECEIVED	**New Page Received** 1 = new page received. 0 = new page not yet received (*default*)
0	LINK PARTNER AUTO-NEGOTIATION ABLE	**Link Partner Auto-negotiation Capability** 1 = link partner has auto-negotiation capability. 0 = link partner does not have auto-negotiation capability (*default*)

Table 3-18. PHY Register 16 — Silicon Revision

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///						SILICON REVISION				///					
RESET	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-19. PHY Register 16 — Silicon Revision Definitions

Bits	Field Name	Description
15:10	///	**Reserved**
9:6	SILICON REVISION	**Silicon Revision** 4-bit silicon revision identifier.
5:0	///	**Reserved**

Table 3-20. PHY Register 17 — Mode/Control Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	EDSHORT	FASTRIP	EDPWRDOWN	///	LOWSQEN	MDPREBP	///	FASTEST	///			REFCLKEN	PHYADBP	FORCE GOOD LINK STATUS	ENERGYON	///
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Table 3-21. PHY Register 17 — Mode/Control Status Definitions

Bits	Field Name	Description
15	EDSHORT	**Energy Detect Short Detection Mode** 1 = short detect mode. 0 = normal detect mode (*default*).
14	FASTRIP	**10Base-T Fast Mode** 1= activate PHYT_10 test mode. 0 = normal operation (*default*).
13	EDPWRDOWN	**Enable the Energy Detect Power-Down Mode** 1 = Energy Detect Power-Down is enabled. 0 = Energy Detect Power-Down is disabled (*default*).
12	///	**Reserved** Must be 0.
11	LOWSQEN	**Low Squelch Noise** The Low_Squelch signal is equal to LOWSQEN and EDPWRDOWN. 1 = imply a lower, more-sensitive threshold. 0 = imply a higher, less-sensitive threshold (*default*).
10	MDPREBP	**Management Data Preamble Bypass** 1 = detect SMI packets without preamble 0 = detect SMI packets with preamble (*default*).
9	///	**Reserved** Leave this bit set to 0.
8	FASTEST	**Auto-negotiation Test Mode** 1 = activate test mode. 0 = normal operation (*default*).
7:5	///	**Reserved**
4	REFCLKEN	1 = enable special filtering using a 50 MHz clock in 10Base-T mode. 0 = do not enable special filtering using a 50 MHz clock in 10Base-T mode (*default*).
3	PHYADBP	**PHY Address in SMI Access Write** 1 = PHY disregards PHY address in SMI access write. 0 = PHY does not disregard PHY address in SMI access write (*default*).
2	FORCE GOOD LINK STATUS	**Force Good Link Status** 1 = force link status in 100Base-TX to good state for testing. 0 = normal operation (*default*).
1	ENERGYON	**ENERGYON Signal State** The ENERGYON signal is asserted when there is valid energy from the line: 100Base-TX, 10Base-T, or Auto-negotiation signals. The device is powered-down if the ENERGYON signal is at logic zero. When ENERGYON rises the PHY is powered-up. For more information, see Energy Detect Power-Down on page 89.
0	///	**Reserved**

Table 3-22. PHY Register 18 — Special Modes

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///		///	DSPBP	SQBP	FXMODE	PLLBP	ADCBP	MODE			PHYADD				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-23. PHY Register 18 — Special Mode Definitions

Bits	Field Name	Description
15:14	///	**Reserved** Leave these bits set to 00.
13	///	**Reserved** This bit is always 0.
12	DSPBP	**DSP Bypass Mode** 1 = enabled. 0 = disabled (*default*).
11	SQBP	**Squelch Bypass Mode** 1 = enabled. 0 = disabled (*default*).
10	FXMODE	**100Base-FX Mode** 1 = enabled. 0 = disabled (*default*).
9	PLLBP	**PLL Bypass Mode** 1 = enabled. 0 = disabled (*default*).
8	ADCBP	**ADC Bypass Mode** 1 = enabled. 0 = disabled (*default*).
7:5	MODE	**PHY Operating Mode** Controls the configuration of the 10/100 digital block. See Table 3-24.
4:0	PHYADD	**PHY Address** The PHY address is used for the SMI address and for the initializing the cipher (scrambler) key.

Table 3-24. Mode Bus

MODE Bits [2:0]	Mode Definitions	Default Register Bit Values	
		Register 0 Bits [13], [12], [10], [8]	Register 4 Bits [8], [7], [6], [5]
000 (*default*)	10Base-T Half-Duplex. Auto-negotiation disabled.	0000	///
001	10Base-T Full-Duplex. Auto-negotiation disabled.	0001	///
010	100Base-TX Half-Duplex. Auto-negotiation disabled. CRS is active during transmit and receive.	1000	///
011	100Base-TX Full-Duplex. Auto-negotiation disabled. CRS is active during receive.	1001	///
100	100Base-TX Half-Duplex is advertised. Auto-negotiation is enabled. CRS is active during transmit and receive.	1100	0100
101	Repeater mode. 100Base-TX Half-Duplex is advertised. Auto-negotiation is enabled. CRS is active during receive.	1100	0100
110	Power Down mode. In this mode the PHY wake-up in Power-Down mode.	///	///
111	All capable. Auto-negotiation enabled.	0100	1111

Table 3-25. PHY Register 20 — TSTCNTL

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	READ	WRITE	///			TEST MODE	READ ADDRESS					WRITE ADDRESS				
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-26. PHY Register 20 — TSTCNTL Definitions

Bits	Field Name	Description
15	READ	**Read Address** 1 = content of the register that is selected by the READ ADDRESS will be latched to the TSTREAD1/2 registers. 0 = register contents will not be latched to the TSTREAD1/2 registers (*default*). This bit is self-clearing.
14	WRITE	**Write Address** 1 = register that is selected by the WRITE ADDRESS is going to be written with the data from the TSTWRITE register. 0 = register is not going to be written with the data from the TSTWRITE register (*default*). This bit is self-clearing.
13:11	///	**Reserved**
10	TEST MODE	**Enable the Testability/Configuration Mode** 1 = Testability/Configuration mode enabled. 0 = Testability/Configuration mode disabled (*default*).
9:5	READ ADDRESS	**READ ADDRESS** The address of the Testability/Configuration register that will be latched into the TSTREAD1 and TSTREAD2 registers.
4:0	WRITE ADDRESS	**WRITE ADDRESS** The address of the Testability/Configuration register that will be written.

Table 3-27. PHY Register 21 — TSTREAD1

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	READ_DATA[15:0]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-28. PHY Register 21 — TSTREAD1 Definitions

Bits	Field Name	Description
15:0	READ_DATA[15:0]	**Read_Data** When reading registers with a size of less than 16 bits, this register contain the register data, starting from bit [0]. When reading registers with a size of more then 16 bits, this register contain the less-significant 16 bits of the register data.

Table 3-29. PHY Register 22 — TSTREAD2

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	READ_DATA[31:16]															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-30. PHY Register 22 — TSTREAD2 Definitions

Bits	Field Name	Description
15:0	READ_DATA[31:16]	**Read_Data** When reading registers with a size of less than 16 bits, this register clears to zeros. When reading registers with a size of more than 16 bits, this register contains the most-significant bits of the register data, starting from bit [16].

Table 3-31. PHY Register 23 — TSTWRITE

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	WRT_DATA															
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-32. PHY Register 23 — TSTWRITE Definitions

Bits	Field Name	Description
15:0	WRT_DATA	**Write_Data** This field contains the data that will be written to a specific register on the “programming” transaction.

Table 3-33. PHY Register 27 — Special Control/Status Indications

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///			SWRST_FAST	SQEOFF	VCOOFF_LP	VCOOFF	TENPROD	100BTLB	TENLBT	FEFIEN	XPOL	AUTONEGS			
RESET	0	0	0	0	0	0	0	0	0	0	—	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R

Table 3-34. PHY Register 27 — Special Control/Status Indications Definitions

Bits	Field Name	Description
15:13	///	**Reserved**
12	SWRST_FAST	**Accelerate SW Reset Counter** 1= accelerate SW reset counter from 256 ms to 10 us for production testing. 0 = does not accelerate SW reset counter (*default*).
11	SQEOFF	**SQE Test (Heartbeat)** 1 = SQE test is disabled. 0 = SQE test is enabled (*default*).
10	VCOOFF_LP	**PLL 10M Lock** Forces the Receive PLL 10M to lock on the reference clock and not on the data. 1 = receive PLL 10M is locked on the reference clock (free run). 0 = receive PLL 10M is locked on the received data (*default*).
9	VCOOFF	**Link Pulse Rejection** Disables rejection of Link Pulses (10Base-T) of less than 20 ns. 1 = rejection disabled. 0 = rejection enabled (*default*).
8	TENPROD	**10Base-T Clocking** Forces the source for the Transmit/Receive (20 MHz from the PLLs) clocks of the 10Base-T to be the REF_25 (25 MHz, which is the external REF_CLK divided by 2) clock. 1 = REF_25 becomes the source clock for the 10Base-T. 0 = normal 10Base-T operating mode (*default*).
7	100BTLB	**Internal Loopback** Forces the internal loopback in the 100Base-TX. 1 = forces internal loopback. Loops back the transmitted 5 bit symbols (scrambled) back to the receiver. Forces “Signal-Detect” to the active mode Loops back the transmitted FLPs to the FLP detector in the auto-negotiation module. 0 = normal operation (*default*).
6	TENLBT	**Force 10Base-T CRS to Active State** 1 = 10Base-T’s CRS forced to HIGH. 0 = normal operation (*default*).
5	FEFIEN	**Far End Fault Indication (FEFI) Enable** 1 = FEFI generation and detection are enabled. Default when FXMODE configuration input is HIGH during reset/ 0 = FEFI generation and detection are disabled.
4	XPOL	**10Base-T Polarity State** 1 = reversed polarity. 0 = normal polarity (*default*).
3:0	AUTONEGS	**Auto-negotiation Transmit State-Machine State**

Table 3-35. PHY Register 29 — Interrupt Source Flags

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///								INT7	INT6	INT5	INT4	INT3	INT2	INT1	///
INT2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 3-36. PHY Register 29 — Interrupt Source Flags Definitions

Bits	Field Name	Description
15:8	///	**Reserved**
7	INT7	**ENERGYON** 1 = ENERGYON generated. 0 = not source of interrupt (*default*).
6	INT6	**Auto-negotiation** 1 = auto-negotiation complete. 0 = not source of interrupt (*default*).
5	INT5	**Remote Fault** 1 = remote fault detected. 0 = not source of interrupt (*default*)
4	INT4	**Link Down** 1 = link is down, link status negated. 0 = not source of interrupt (*default*).
3	INT3	**Auto-negotiation LP Acknowledge** 1 = auto-negotiation LP acknowledge. 0 = not source of interrupt (*default*).
2	INT2	**Parallel Detection Fault** 1 = parallel detection fault. 0 = not source of interrupt (*default*).
1	INT1	**Auto-negotiation Page Received** 1 = auto-negotiation page received. 0 = not source of interrupt (*default*).
0	///	**Reserved**

Table 3-37. PHY Register 30 — Interrupt Mask

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	///								INT7 MASK BIT	INT6 MASK BIT	INT5 MASK BIT	INT4 MASK BIT	INT3 MASK BIT	INT2 MASK BIT	INT1 MASK BIT	///
INT2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Table 3-38. PHY Register 30 — Interrupt Mask Definitions

Bits	Field Name	Description
15:8	///	**Reserved**
7	INT7 MASK BIT	**ENERGYON Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
6	INT6 MASK BIT	**Auto-negotiation Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
5	INT5 MASK BIT	**Remote Fault Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
4	INT4 MASK BIT	**Link Down Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
3	INT3 MASK BIT	**Auto-negotiation LP Acknowledge Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
2	INT2 MASK BIT	**Parallel Detection Fault Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
1	INT1 MASK BIT	**Auto-negotiation Page Received Interrupt Source** 1 = interrupt source is enabled. 0 = interrupt source is masked.
0	///	**Reserved**

Table 3-39. PHY Register 31 — PHY Special Control/Status

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OFFSET																
FIELD	BPRMG	///	///	AUTODONE	///					ENABLE4B5B	LINK CUTOFF	SPEED INDICATION			///	SCRAMBLE DISABLE
RESET	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW

Table 3-40. PHY Register 31 — PHY Special Control/Status Definitions

Bits	Field Name	Description
15	BPRMG	**Reserved**
14	///	**Reserved**
13	///	**Reserved**
12	AUTODONE	**Auto-negotiation Done Indication** 1 = auto-negotiation is done. 0 = auto-negotiation is not done, is disabled, or is not active (*default*).
11:7	///	**Reserved**
6	ENABLE4B5B	**4b5b Encoding/Decoding** 1 = enable 4B5B encoding/decoding (*default*). 0 = bypass encoder/decoder.
5	LINK CUTOFF	**Force Transmit State-Machine of Auto-negotiation** Force the Transmit State-machine of auto-negotiation to the “Auto-negotiation Enable” state, continuously: 1 = disable Auto-negotiation by forcing the Transmit State-machine to the “Auto-Negotiation Enable” state. 0 = normal auto-negotiation operation (*default*).
4:2	SPEED INDICATION	**HCDSPEED Value** 000 = no HCDSPEED value (*default*). 001 = 10Mbps half-duplex. 101 =10Mbps full-duplex. 010 =100Base-TX half-duplex. 110 =100Base-TX full-duplex.
1	///	**Reserved**
0	SCRAMBLE DISABLE	**Data Scrambling** 1 = disable data scrambling. 0 = enable data scrambling (*default*).

Miscellaneous Functions

Carrier Sense

The carrier sense is output on CRS. CRS is a signal defined by the MII specification in the IEEE 802.3u standard. CRS becomes active based on receive only, when the chip is in repeater or full-duplex mode. Otherwise, it becomes active based on either transmit or receive activities.

The carrier sense logic uses the encoded, unscrambled data to determine carrier activity status. It activates carrier sense with the detection of 2 noncontiguous zeros within any 10-bit span. Carrier sense ends if a span of 10 consecutive 1s is seen before a /J/K/ Start-of Stream Delimiter pair. If an SSD pair is seen, carrier sense is maintained until either /T/R/ End-of-Stream Delimiter pair or a pair of IDLE symbols is seen. Carrier is negated after the /T/ symbol or the first IDLE. If /T/ is not followed by /R/, then carrier is maintained. Carrier is treated similarly for IDLE followed by some non-IDLE symbol.

Collision Detect

A collision is the occurrence of simultaneous transmit and receive operations. If a collision occurs, the COL output is asserted to indicate that a collision has been detected. COL remains active for the duration of the collision. COL changes asynchronously to both RX_CLK and TX_CLK. The COL output becomes inactive during full-duplex mode.

COL can be tested by setting bit [7] of Register 0 HIGH (see Table 3-4. PHY Register 0 — Basic Control on page 70). This enables the collision test. COL is asserted within 512-bit times of TX_EN rising and de-asserted within 4-bit times of TX_EN falling.

In 10M mode, COL pulses for approximately 10-bit times, 1us after each transmitted packet. This is the Signal Quality Error (SQE) signal and indicates that the transmission was successful.

Link Integrity Test

The Ethernet PHY performs the link-integrity test as outlined in the IEEE 802.3u (Clause 24-15) Link Monitor state diagram. The link status multiplexes with the 10 Mbps link status to form the reportable link status bit in Serial Management Register 1 and is driven to LINK LED.

The DATA_VALID signal indicates a valid MLT-3 waveform on the RXP and RXN signals, as defined by the ANSI X3.263 TP-PMD standard. When DATA_VALID is asserted, the control logic moves into a Link-Ready state and waits for an enable from the Auto Negotiation block. When received, the Link-Up state is entered and the Transmit and Receive logic blocks become active. If auto-negotiation is disabled, the link integrity logic moves immediately to the Link-Up state.

To allow the line to stabilize, the link integrity logic waits at least 330 msec from the time DATA_VALID is asserted until the Link-Ready state is entered. If the DATA_VALID input is negated at any time, this logic immediately negates the LINK signal and enters the Link-Down state.

When the 10/100 digital block is in 10Base-T mode, the link status is from the 10Base-T receiver logic.

Power-Down Modes

There are two power-down modes for the core:

- General power-down
- Energy power-down

General Power-Down

Bit [11] of Register 0 controls this power-down mode (see Table 3-4. PHY Register 0 — Basic Control on page 70). In this mode, the entire PHY, except the management interface, powers-down and stays in that condition as long as bit [11] of Register 0 is HIGH. When bit [11] clears, the PHY powers-up and automatically resets.

Energy Detect Power-Down

This power-down mode is activated by setting bit [13] of Register 17 to a logical 1 (see Table 3-20. PHY Register 17 — Mode/Control Status on page 78). In this mode, the PHY— excluding the management interface, SQUELCH circuit, and ENERGYON logic — powers-down.

The ENERGYON signal is asserted when there is valid energy from the line: 100Base-TX, 10Base-T, or Auto-negotiation signals.

In this mode, the device powers-down if the ENERGYON signal is at logical zero. When the ENERGYON signal rises:

- The PHY powers-up.
- The device automatically resets into the state it was in before it powered-down, and stays in active mode as long as the ENERGYON signal is active.
-

When ENERGYON is asserted, the PHY sets the interrupt output, INT_B, to logical zero (if the ENERGYON interrupt is enabled). Clearing bit [13] of Register 17 disables this mode.

Reset

The core PHY has three reset sources:

- Hardware reset
- Software reset
- Power-down reset

These three reset sources are combined together in the digital block to create the internal “general reset,” SYSRST, which is an asynchronous reset and is active HIGH. SYSRST directly drives the digital, DSP, and RMII blocks. It is also input to the Central Bias block to generate a short reset for the PLLs.

The SMI mechanism resets only by hardware and software resets. During power-down, the SMI registers, do not clear.

Hardware Reset (HWRST)

HWRST is a logical OR of the RSTIN and the PHY Power-On Reset circuit.

Software Reset

Writing bit [15] of Register 0 HIGH activates a software reset (see Table 3-4. PHY Register 0 — Basic Control on page 70). This signal is self-clearing. The internal software reset extends by 256µs after the register write to let the PLLs stabilize before the logic is released from reset. The IEEE 802.3u standard, clause 22 (22.2.4.1.1), states that the reset process should complete within 0.5s from the setting of this bit.

Power-Down Reset

A power-down reset automatically activates when the PHY exits power-down mode. The internal power-down reset extends by 256µs after exiting the power-down mode to let the PLLs stabilize before the logic is released from reset.

LED Description

There are four LED signals available for use.

- Activity
- Link
- Speed 100
- Full-duplex

These LEDs offer an easy way to determine the core's operating mode. All LED signals are active LOW. Table 3-41 describes the LED signals.

Table 3-41. LED Signals

Signal	Description
Activity	The Activity signal is driven low when CRS is active (HIGH). This output stays LOW while CRS is active, but remains ON for at least 128 ms.
Link	Link signal is driven low by the link test returning a true result to indicate a valid link. The use of the 10 Mbps or 100 Mbps link test status is determined by the condition of the internally determined speed selection.
Speed 100	The Speed 100 signal is driven LOW when the operating speed is 100 Mbit/s or during autonegotiation operation. This LED is inactive when the operating speed is 10 Mbit/s or during line isolation (bit [5] of Register 31).
Full-Duplex	The Full-Duplex signal is driven LOW when the link is operating in Full-Duplex mode.

Loopback Operation

The 10/100 digital has an internal loopback mode.

Internal Loopback

The internal loopback mode is enabled by setting bit [14] of Register 0 to a logical 1 (see Table 3-4. PHY Register 0 — Basic Control on page 70). In this mode, the scrambled transmit data (output of the scrambler) loops into the receive logic (input of the descrambler). The COL signal is inactive in this mode, unless collision test (bit [7] of Register 0) is active.

Architecture Details

The following sections provide detailed descriptions of the following main architectural blocks. The term “block” defines a stand-alone entity on the floor plan of the chip.

Mode Bus – MODE

The MODE[2:0] bus controls the configuration of the 10/100 digital block (as shown in Table 3-42).

Table 3-42. 10/100 Digital Block Configuration

MODE[2:0]	Mode Definitions	Default Register Bit Values	
		Register 0	Register 4
		Bits [13,12,10,8]	Bits [8,7,6,5]
000 (*default*)	10Base-T half-duplex. Auto-negotiation is disabled.	0000	///
001	10Base-T full-duplex. Auto-negotiation is disabled.	0001	///
010	100Base-TX half-duplex. Auto-negotiation is disabled. CRS is active during transmit and receive.	1000	///
011	100Base-TX full-duplex. Auto-negotiation is disabled. CRS is active during receive.	1001	///
100	100Base-TX half-duplex is advertised. Auto-negotiation is enabled. CRS is active during transmit and receive.	1100	0100
101	Repeater mode. Auto-negotiation is enabled. 100Base-TX half-duplex is advertised. CRS is active during receive.	1100	0100
110	Power Down mode. In this mode the PHY wakes up in Power-Down mode.	///	///
111	All capable. Auto-negotiation is enabled.	0100	1111

Analog Blocks

The following sections describe the core PHY's analog blocks.

ADC

The ADC is a 6-bit, 125 MHz sample rate Analog to Digital Converter designed to be the analog front end of a digital 100Base-Tx receiver.

Functional Description

The ADC has a full-flash architecture for maximum speed and minimum latency. An internally generated 125 MHz clock is used to time the sampling and processing.

The ADC has a variable gain, which is controlled by the DSP block. This allows accurate A/D conversion over the entire range of input signal amplitudes, which is particularly important for lower amplitude signals (longer cables).

Input Common Mode

The differential input is applied to the RXP/N signals. For proper operation of the ADC the input common mode should match the internal differential reference common mode. To achieve this, the ADC generates the appropriate voltage and drives it via the CM signal.

General Characteristics

Table 6-12 lists the core PHY's general characteristics.

core PHY General Characteristics

100M PLL

The 100M PLL provides three main functions:

- A clock multiplier that generates a 125 MHz clock
- A phase interpolator that synchronizes the receive clock to the receive data
- A transmit wave-shaping delay reference

The clock multiplier generates a multiple-phase 125 MHz from a 25 MHz reference frequency.

The phase interpolator uses a multiplexer to select the phase used as the receive clock, RX_CLK. The multiplexer is controlled by signals generated in the DSP Timing unit. The Timing unit estimates the frequency drift of the received data clock; by incrementing, decrementing, or maintaining the selected phase, it generates a clock that is synchronized to the received data stream.

The 100M PLL also generates a fixed-phase 125 MHz clock, slaved to the VCO. The digital filter uses this clock for accurate wave-shaping of the transmit output. This clock is also used as the transmitter clock of the PHY, TX_CLK. (This clock must be jitter-free and, therefore, cannot be the receive clock).

XMT_100

This block generates the differential outputs driven onto TXP/N in 100Base-TX mode.

This block is a wave-shaped 100Base-TX transmitter, with high-impedance current outputs. The three-level differential output (MLT-3) is shaped by differential current switches whose outputs are connected. The low-pass filtering (wave-shaping) of the current output is done by progressive switching of small current sources. The timing reference for the wave-shaping is the 125 MHz fixed clock from the 100M PLL. The transmitter is designed to operate with a 1:1 transformer.

10M Squelch

The squelch circuit consists of squelch comparators and data comparators which operate according to Section 14.3.1.3.2 of the IEEE 802.3 standard.

10Base-T Low-Pass Filter

The 10Base-T low-pass filter is the front end of 10Base-T signal path. It rejects the high-frequency noise from entering the squelch and data-recovery blocks.

10M PLL – Data Recovery Clock

The data recovery PLL is used for data recovery for the 10Base-T mode of operation. The data recovery PLL synchronizes the phase of the 10Base-T data and the 20 MHz VCO.

The Data recovery PLL has two operating modes:

- Frequency Mode — where the VCO locks to the external reference clock.
- Data Mode — where the VCO locks to the incoming data. When the PLL switches to Data mode, the VCO is held. It is released on an incoming data edge. This provides minimal phase error when the PLL switches from Frequency Mode to Data Mode.

10M PLL – Transmit Clock

The transmit PLL generates a precise delay for the 10Base-T transmitter. It also provides a 20 MHz clock for the transmit digital block.

This PLL provides a Transmit clock to the digital and create a delay for the 10Base-T transmitter.

The Transmit PLL operates continuously in a frequency mode of operation where it is locked to the 25 MHz reference clock.

XMT_10

This block generates the differential outputs driven onto TXP/N in 10Base-T mode. This block is a wave-shaped 10Base-T transmitter, with high impedance current outputs. The low-pass filtering (wave-shaping) of the current output is achieved by progressive switching of small current sources. The timing reference for the wave-shaping is the 10Base-T transmit PLL. The transmitter operates with a 1:1 turn-ratio transformer.

Central Bias

The Central Bias block generates a power-up reset signal, a PLL reset signal, and the bias currents/voltages that other on-chip blocks need. This block has three main functions:

- Reference bias current and voltage generator
- Power-up reset
- PLL reset

The bias generator generates accurate currents and voltages using an on-chip bandgap circuit and an external 12.4K 1% resistor.

The power-up reset circuit generates a signal that remains HIGH for 10 ms. This duration is controlled through counters and a 25 MHz internal clock. An analog power-up circuit sets the initial conditions and ensures proper startup of the circuit.

The PLL reset signal generates after an active RESET_B occurs. The internal reset signal is asserted for the duration of four 25 MHz clocks (160 ns), and is then released. Releasing the PLL reset early ensures that the PLL locks to the reference clock before the system reset (RESET) is released.

DSP Block

The DSP block includes the following modules:

- DSP Core (Equalizer, Timing and BLW correction)
- Testability / Configuration module (Testability / Configuration control)
- Testability / Configuration Registers (excluding any SMI registers)
- Multiplexers (for the testability / configuration signals)

The details of the DSP core are described Architecture Details on oage 91. The Testability / Configuration features give access to the status and control of most of the internal registers in the DSP. The status and control mechanisms are also described under Architecture Details on oage 91.

ADC Gray Code Converting

The PHY ADC generates a 6-bit “modified” Gray code. Normal Gray code outputs number in the range of 0 to 2n – 1. The 6-bit code generates numbers from 0 to 63 (decimal).

The MLT3 analog input has a voltage range of –1V to +1V. You must translate this to –32 to +31 on the ADC output. As a result, the Gray Code is modified by offsetting it by –32. This is translated to 2’s complement before being presented to the DSP.

Operating Modes

The core supports the following modes of operation:

- Normal mode
- Testability / configuration mode
- ADC bypass mode
- Squelch bypass mode

The modes are entered via configuration inputs that are sampled during reset, and/or can be set with Register 18 (see Table 3-22. PHY Register 18 — Special Modes on page 79). Each mode assigns different functions to some of the signals.

Normal Mode – MII

MII must be LOW on the rising edge of RESET to enter this mode. In this mode, the PHY interfaces to the controller via the MII.

Using its management register set (SMI), you can activate the following testing options in the 10/100 logic:

- Different looping options
- Fast Mode options to accelerate timers and long counters
- Bypass option for cipher/decipher
- Bypass option for encoder/decoder
- Forcing/monitoring internal controls

Testability / Configuration Mode

This mode is used to debug the DSP and the analog circuits. This mode lets you:

- Change parameters in the DSP
- Monitor (via external signals or the management registers (SMI)) registers in the DSP
- Control the testing/programming features of the analog circuits
- A special HW mechanism is used to prevent users from entering this mode erroneously

In this mode, you can control/activate the DSP block and analog block features.

Using the DSP block, you can:

- Program (write) most of the registers and default parameters.
- Read most of the registers.
- Monitor the main buses/control signals through external signals.
- Externally trigger the input that is used to latch the contents of a specific register to a global register. The global register can be read later.
- Externally trigger the output that generates the trigger to indicate one of several internal events.

Using the analog blocks, you can perform block isolation, where each block can be enabled/powered-down individually.

PLL Bypass Mode

Setting the PLLBP configuration input HIGH, during reset activates this option. In this mode, the 100M PLL powers-down and two 125 MHz external clocks are injected.

- One clock has the controlled-phase and is used as the receive clock.
- The other clock has the fixed-phase and is used as the transmit clock.
- These clocks bypass the PLL and are used as the system clocks.

This mode is used for testing.Since there is no PLL in this mode, the transmit clock is not locked on the REF_CLK. Consequently, internal logic divides the 125 MHz transmit clock (driven from the external source) to get the 50 MHz clock.

ADC Bypass Mode

Setting the ADCBP configuration input HIGH, during reset activates this option. In this mode, the internal ADC powers-down and the 6-bit-wide data representing the analog input is injected straight to the DSP via external pins. The gain of the internal ADC is controlled by 3 bits from the DSP, which are output on external signals to set the external ADC gain. To generate synchronous vectors from the external source, the chip should be entered to the “Clock Out Mode” and configured to output the CLK125 clock.

Squelch Bypass Mode

This mode is used when the analog SQUELCH circuit needs to be bypassed. In this mode, the 4 input signals are injected to the 10M PCS, via external pins. This mode can be set only with the bits from Register 18. To generate synchronous vectors from the external source, the chip should be entered to the “Clock Out Mode” and configured to output the RX_20 clock.