LSI LOGIC

Full Duplex

HURRICANE™

80C300

Full Duplex CMOS Ethernet
10/100 Mega Bit/Sec Data Link Controller

98012

Features

- ***Low Power CMOS Technology***
- ***10/100 MBit Ethernet Controller Optimized for Switching Hub, Multiport Bridge/Router, & Server Applications***
- ***Meets ANSI/IEEE 802.3 and ISO 8802-3 Standards for Ethernet (10Base-5) Thin Net (10Base-2) (10Base-T) and the Proposed 100Base-T4, 100Base-TX Standards***
- ***10 MHz Serial/Parallel Conversion in 10 MBit/sec Serial Mode.***
- ***Standard 10MBit/sec Serial Mode or Programmable MII Ethernet Interface for 10/100 MBit/sec Applications***
- ***Programmability of Double Word Threshold Count for Space Available/Data Available Ready Condition for Transmit/Receive FIFOs***
- ***Auto Retransmit Upon Collision Sense***
- ***Preamble Generation and Removal***
- ***Automatic 32-Bit FCS (CRC) Generation and Checking***
- ***Collision Handling, Transmission Deferral and Retransmission with Automatic Jam and Backoff Functions***
- ***Error Interrupt and Status Generation***
- ***Selectable Little Endian/Big Endian Transmit Byte Ordering for FIFO Interface for Intel/Motorola Compatibility***
- ***Single 5 V± 5% Power Supply***
- ***Standard CPU and Peripheral Interface Control Signals***
- ***128/128 Byte Independent Transmit/Receive FIFOs with 32 Bit Data Path Interface***
  - ***1 G Bits/sec (133 M Bytes/sec) Peak Data Rate in 32 Bit Mode.***
- ***Loopback Capability for Diagnostics***
- ***32 Bit FIFO Data Path***

Note: Check for latest Data Sheet revision before starting any designs.

SEEQ Data Sheets are now on the Web, at www.lsilogic.com.

This document is an LSI Logic document. Any reference to SEEQ Technology should be considered LSI Logic.

- ***Inputs and Outputs TTL Compatible***
- ***The Following Additional Features can be Programmed for the 80C300***
  - ***64 bit Multicast Filter***
  - ***Reports Status of "SQE" During Transmits***
  - ***Transmit No CRC Mode***
  - ***Transmit No Preamble Mode***
  - ***Transmit Packet Autopadding Mode***
  - ***Receive CRC Mode***
  - ***Disable Self-Receive on Transmit Mode***
  - ***Disable Further Transmissions when Both Transmit Status Registers are Full***
  - ***Disable Loading the Transmit Status for Successfully Transmitted Packets***
  - ***Disable the Receive Interrupts Independent of the Receive Command Register Setting***
  - ***Successful Packet Transmit Completion Feature***
- ***Full Duplex Operation***
  - ***Provides 20/200 Mbps Bandwidthfor Switched Networks***
  - ***Supports AutoDUPLEX Mode for Automatic Full Duplex Operation***
- ***Transmit Status on a Per Packet Basis Reports the Following***
  - ***Occurrence of a Transmit FIFO Underflow***
  - ***Transmit Collision Occurrence***
  - ***16 Collision Occurrence***
  - ***Carrier Sense Error During Transmission***
  - ***10/100 Mbit/sec Transmit Clock Detect***
  - ***Late Collision Occurrence***
  - ***Transmission Successful***
  - ***Transmission Deferred***

- *Management Counters for*
  - *Alignment Errors*
  - *FCS Errors*
  - *Runt Receive Frames*
  - *Short Receive Events*
  - *Oversized Receive Packets*
  - *Transmit Collisions*
  - *Receive Collisions*
  - *Very Long Transmit Events*
  - *Excessive Transmit Deferral*
  - *Late Transmit Collisions*
  - *Transmit Excessive Collisions*
  - *Symbol Errors (100 MBit/sec Ethernet Only)*
  - *Total Octets Received*
  - *Total Octets Transmitted*
  - *Receive FIFO Overflows*
  - *Total Rx Multicast, Unicast and Broadcast Frames*
  - *Total Tx Multicast, Unicast and Broadcast Frames*
  - *Tx Defer Count*
  - *Number of Retransmit Attempts*
- *128 Pin PQFP package*

Table of Contents

Illustrations

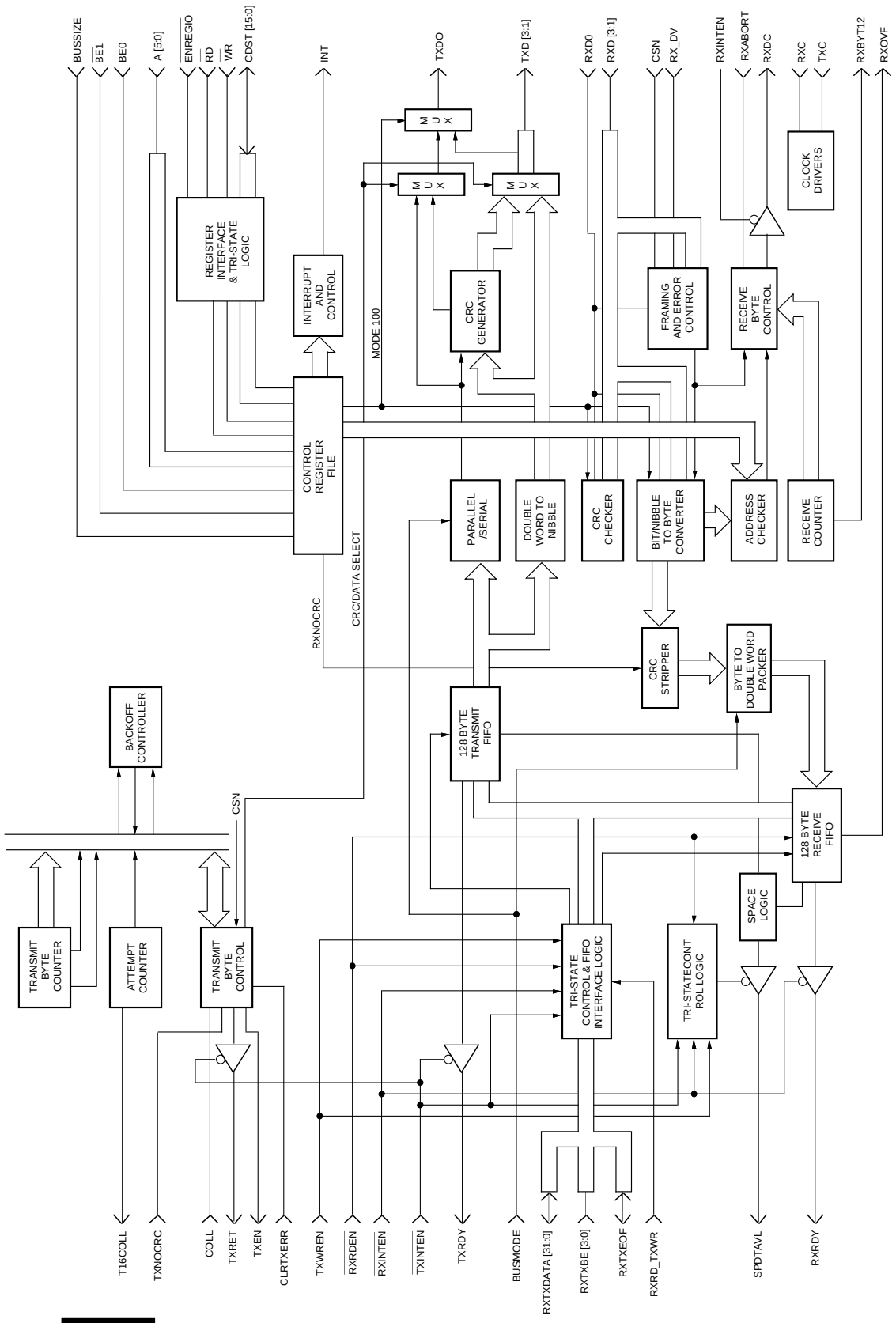


Figure 1. 80C300 Functional Block Diagram

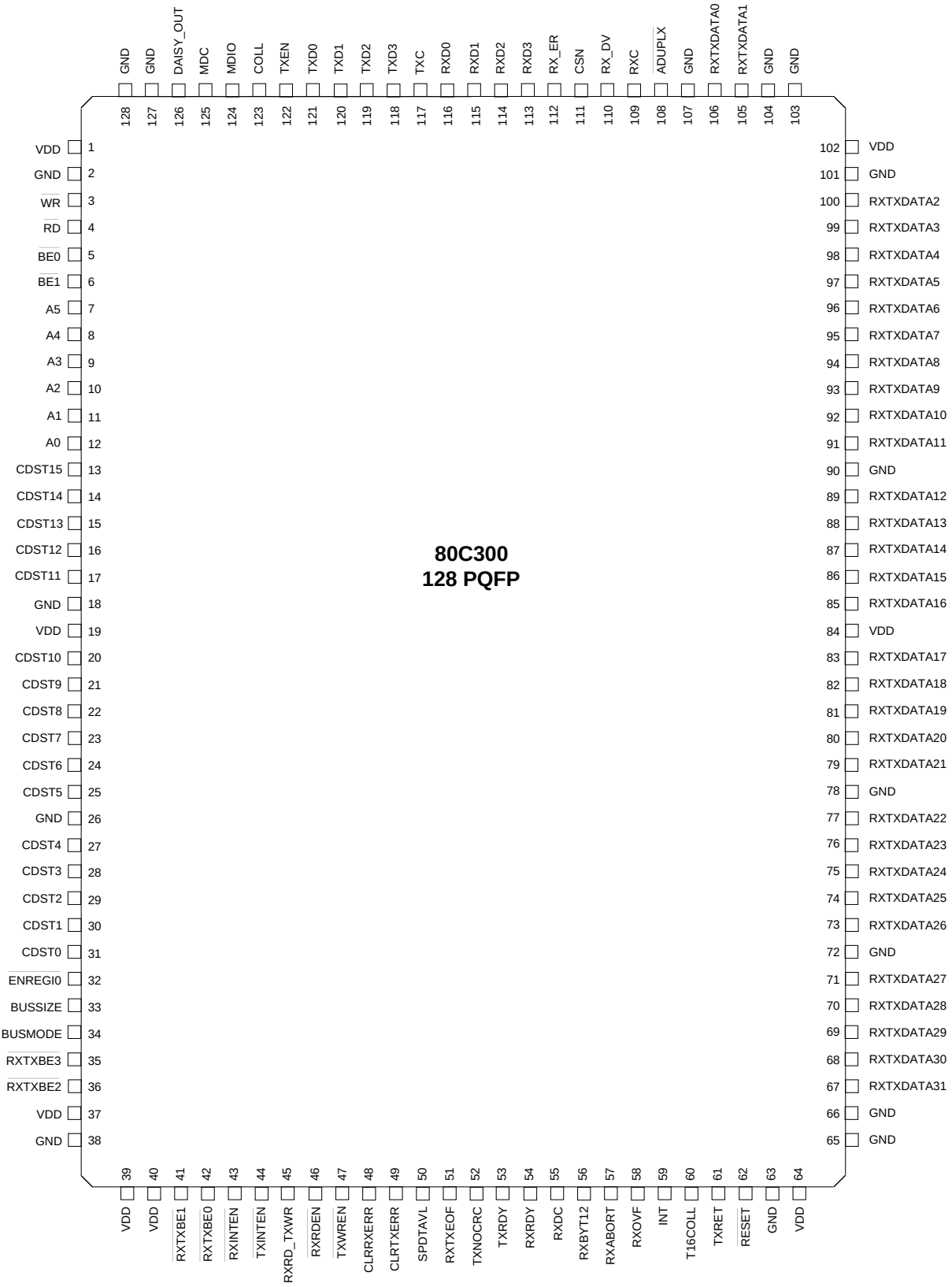


Figure 2. 80C300 Pin Configuration

1.0 PIN DESCRIPTION

Pin	Pin Name	I/O	Description
Chip Register Interface			
32	$\overline{\text{ENREGIO}}$	I	This active low input enables the chip for register operations. This input must be low before any registers can be written or read.
3	$\overline{\text{WR}}$	I	Within the chip, this input acts as a write strobe for one of the registers and the register is addressed through the A[5:0] address inputs. The data being written appears on the CDST[15:0] data lines and must be set up relative to the rising edge of the write strobe. This input is active low.
4	$\overline{\text{RD}}$	I	This input acts as a read strobe for one of the registers and the register is addressed through the A[5:0] address inputs. When the read strobe is active low, the output drivers for CDST[15:0] data bus are enabled. Valid register data appears on the data bus a specified time before the rising edge of the read strobe.
7-12	A[5:0]	I	These inputs are the address lines used to select which register is being read or written.
33	BUSSIZE	I	This input when HIGH sets the register interface of the 80C300 to 16-bit mode and when LOW to 8-bit mode. In 8-bit mode only CDST[7:0] of the CDST data bus is used for register accesses.
6, 5	$\overline{\text{BE[1:0]}}$	I	These inputs are active LOW byte enables to enable byte wide register accesses when the 80C300's register interface is in 16-bit mode.
13-17 20-25 27-31	CDST[15:0]	I/O	These bidirectional lines carry register data to or from the internal registers of the chip. These lines are nominally high impedance until their output drivers are enabled by the $\overline{\text{RD}}$ and $\overline{\text{ENREGIO}}$ input pins being driven low.
59	INT	O	This output is driven high by a variety of transmit and receive interrupt conditions. It remains high until the status register containing the reason for the interrupt is read.
62	$\overline{\text{RESET}}$	I	This input is an active low asynchronous chip reset. After reset, all registers except the Hash and Station Address registers are reset to zero, all FIFOs are cleared, all counters are reset to zero.
Receive and Transmit FIFO Interface			
34	BUSMODE	I	This input when HIGH sets the FIFO interface to Big Endian mode. In this mode data written to the transmit FIFO is transmitted most significant byte first. When this pin is LOW, Little Endian transmit order is used.
43	$\overline{\text{RXINTEN}}$	I	This is an active low input that acts as a chip enable to enable the receiver interface. Driving this pin active, enables the output drivers for the RXDC, RXRDY pins. Also, this pin must be driven active before receive FIFO reads can be performed.
44	$\overline{\text{TXINTEN}}$	I	This is an active low input that acts as a chip enable to enable the transmitter interface. Driving this pin active enables the output drivers for the TXRET, TXRDY pins. Also, this pin must be driven active before transmit FIFO writes can be performed.
46	$\overline{\text{RXRDEN}}$	I	This is an active low input that, when driven active with the $\overline{\text{RXINTEN}}$ pin, enables read operations from the receive FIFO within the chip.
47	$\overline{\text{TXWREN}}$	I	This is an active low input that, when driven active with the $\overline{\text{TXINTEN}}$ pin, enables write operations to the transmit FIFO within the chip.
45	RXRD_TXWR	I	This is the system clock acting as the chip's read/write strobe to receive/transmit FIFO. With the $\overline{\text{TXINTEN}}$ and $\overline{\text{TXWREN}}$ inputs active low, this input becomes the write strobe for writing transmit data to the chip's transmit FIFO. Similarly, with the $\overline{\text{RXINTEN}}$ and $\overline{\text{RXRDEN}}$ inputs active low, this input becomes the read strobe for reading receive data from one of the chip's receive FIFO. This input must be connected to a continuous clock whose maximum frequency can be 33 MHz.

1.0 PIN DESCRIPTION (cont.)

Pin	Pin Name	I/O	Description
Receive and Transmit FIFO Interface			
35,36,41 42	RXTXBE [3:0]	I	These are active low inputs that determine which bytes of the double word for a receive FIFO read are driven with valid data or which bytes of a double word being written to a transmit FIFO contain valid data.
53	TXRDY	O	This is an active high tristate output. When enabled, this output functions as a flag that indicates whether the transmit FIFO has enough space available to meet the threshold value programmed in the FIFO threshold register. When enabled, a high value on the output indicates that the transmit FIFO has greater than or equal to the threshold number of double word spaces available in the FIFO and a low value indicates it does not. The tristate driver for the output is enabled by a low value on the TXINTEN input pin.
54	RXRDY	O	This is an active high tristate output. When enabled, this output functions as a flag that indicates whether the receive FIFO has enough data available to meet the threshold value programmed in the FIFO threshold register. When enabled, a high value on the output indicates that the receive FIFO has greater than or equal to the threshold number of double words available in the FIFO or has a completed receive packet in the FIFO as indicated by the packets status double word being in the FIFO. The tristate driver for the output is enabled by a low value on the RXINTEN input pin.
50	SPDTAVL	O	This is an active high output that can be used for validating reads from the receive FIFO during a read operation and preventing over writes to the transmit FIFO during a write operation. For further details, please refer to the Transmit Data Write Timing and the Receive Data Read Timing diagrams.
51	RXTXEOF	I/O	This is a bidirectional pin that is used to signal the last double word of a transmit or receive packet. During receive FIFO reads this pin is enabled as an output and when detected high indicates that the last double word, and or the status double word, of a receive packet has been read from the receive FIFO. During transmit FIFO writes this pin is an input and when asserted high during a write, it indicates that this is the final double word of a transmit packet. In the transmit FIFO write case, the value of this signal is stored as the 33rd bit in the FIFO. In the receive FIFO read case, the value of this signal is read out as the 33rd bit of the receive FIFO.
52	TXNOCRC	I	This active high input is used to control appending of a CRC to a transmit packet. A transmit packet can be made to exclude appending a CRC value if this input is held high during the first double word write of transmit data to the transmit FIFO.
67-71 73-77 79-83 85-89 91-100 105, 106	RXTXDATA[31:0]	I/O	This is the bidirectional 32 bit data bus for reads or writes to the chip's receive or transmit FIFOs. For receive FIFO reads, it is enabled as an output with the assertion of the RXINTEN, RXRDEN, and a low value on the RXRD_TXWR input strobe. Otherwise, it is used as an input.

1.0 PIN DESCRIPTION (cont.)

Pin	Pin Name	I/O	Description
Transmit and Receive Exception Indicators			
61	TXRET	O	This is an active high tristate output. This output pin is driven by tristate drivers enabled by an active low being driven onto the TXINTEN input pin. Once enabled, a high value on this input indicates that the chip could not complete transmission of a packet due to one of the following conditions and that a retransmission of the packet is requested: 1. A late collision occurred during transmission. 2. Carrier sense never went high or dropped out during transmission. 3. During a transmission attempt, a transmit FIFO underflow error occurred. 4. 16 attempts to transmit the packet all resulting in transmit collisions. Internally, the TXRET signal will remain high until it is cleared by the CLRTXERR pin, (See the text on clearing error conditions). As long as the internal TXRET signal remains high, the transmit FIFO will remain cleared and no new transmissions can occur.
60	T16COLL	O	This is an active high tristate output. This output pin is driven by tristate drivers enabled by an active low being driven onto the TXINTEN input pin. Once enabled, a high value on this input indicates that 16 attempts to transmit a packet failed due to 16 collisions. This output is cleared the same way the TXRET is cleared by using the CLRTXERR input.
55	RXDC	O	This is an active high tristate output. This pin driven by tristate drivers enabled by a low value being driven onto the RXINTEN input pin. Once enabled, a high value on the input indicates that the chip discarded reception of a packet due to one of the possible receive discard conditions. Internally, a RXDC signal will remain high until it is cleared by the CLRRXERR pin, (See the text on clearing error conditions). As long as the internal RXDC signal remains high, the receive FIFO will remain cleared and no new packets will be received.
Special Purpose Pins			
49	CLRTXERR	I	This active high input is used to clear transmit retry flags within the chip. See the "Detecting and Clearing a Transmit Retry Condition" section for how this input is used.
48	CLRRXERR	I	This active high input is used to clear Receive Discard flags within the chip. See the "Detecting and Clearing a Receive Discard Condition" section for how this input is used.
57	RXABORT	I	This input when pulsed high causes the chip to abort reception of a receive frame and clear the Receive FIFO. It can be asserted at any time during the reception of a frame.
108	$\overline{\text{ADUPLX}}$	I	This active low input is used to set the chip into Full Duplex Mode. In this mode the transmitter will not defer to an active carrier sense signal.
56	RXBYT12	O	This is an active high output that when high indicates that at least the first 12 bytes of a packet are available within the receive FIFO. Internally this signal is reset to 0 upon Carrier Sense, or Receive Data Valid in MII mode, going low. This is synchronized with respect to the high going edge of the RXRD_TXWR clock.
58	RXOVF	O	This is an active high output that when pulsed high indicates that the receive FIFO overflowed. This signal is pulsed high for one byte time, (two RXC clock periods in MII mode or eight RXC clock periods in 10 Mbit/sec serial mode).
126	DAISY_OUT	NC	This pin should not be connected to any signals externally

1.0 PIN DESCRIPTION (cont.)

Pin	Pin Name	I/O	Description
Media Independent Interface (MII)			
117	TXC	I	This is the transmit clock input for the chip. In standard 10 Mbit/sec Serial Mode, this is a 10 Mhz, 50% duty cycle transmit clock used to synchronize the transmit data from the chip to the encoder. In this mode, transmit data appears serially on the TXD0 output and all transitions of transmit data and the TXEN output occur from the falling edge of the clock. In MII mode, this is a 2.5/25 Mhz, 50% duty cycle clock, and the transmit data appears on the TXD0 through TXD3 outputs. In this mode transitions of transmit data and the TXEN output occur from the rising edge of the clock.
112	RXER	I	This input is internally connected to an 8 bit counter used for counting symbol errors coming in from the PHY.
118-121	TXD[3:0]	O	In standard 10 Mbit/sec Serial Mode, TXD0 is the serial transmit data output from the chip to the encoder. In MII mode, these outputs drive a nibble of transmit data every leading edge of the TXC clock from the chip to the encoder.
122	TXEN	O	This output from the chip is used to activate the encoder. In standard 10 Mbit/sec Serial Mode, it becomes active when the first bit of the Preamble is transmitted and inactive when the last bit of the frame is transmitted. In 100 Mbit/sec mode, this output becomes active when the first nibble of the Preamble is transmitted and inactive when the last nibble of the frame is transmitted. This output is active high.
109	RXC	I	In standard 10Mbit/sec Serial Mode, this input is a 10Mhz, 50% duty cycle nominal receive clock which is used to synchronize incoming data from the decoder to the chip. In 10Mbit/sec Serial Mode CSN and RXD0 are assumed to transition from the leading edge of this clock. In MII mode this clock is a 2.5/25 Mhz, 50% duty cycle receive clock that synchronizes incoming nibble wide data from the decoder to the chip. In MII mode data and the RXDV signal are assumed to transition from the falling edge of the clock.
113-116	RXD[3:0]	I	In standard 10 Mbit/sec Serial Mode, RXD0 is the serial input data to the chip from the decoder. In this mode pins RXD[3:1] can have any logic value but must not be left floating. In MII mode, these inputs are driven with a nibble of receive data every falling edge of the RXC clock from the encoder to the chip.
111	CSN	I	This is the carrier sense input which indicates there is traffic on the transmission medium. Carrier sense becomes active with the first bit of the Preamble received, and inactive one bit time after the last bit of the frame is received. This is an active high input.
110	RX_DV	I	In MII mode this input is receive data valid. Receive data valid becomes active with the first nibble of synchronized and decoded Preamble or SFD appearing on the RXD[3:0] lines and goes inactive one clock time after the last nibble of the frame is received. In 10 Mbit/sec Serial Mode this input is not used and can have any logic value but must not be left floating. This is an active high input.
123	COLL	I	This input indicates that a transmission contention has occurred on the transmission medium. Collision is indicated to the chip by an active high pulse on the COLL input and is automatically reset at the end of the JAM sequence transmission.

1.0 PIN DESCRIPTION (cont.)

Pin	Pin Name	I/O	Description
Management Interface			
125	MDIC	O	This is the clock source from the 80C300 to the PHY and acts as a timing reference for transfer of information on the MDIO signal.
124	MDIO	I/O	This is a bidirectional pin used to transfer data to and from the PHY. Data is transferred out to the PHY synchronously with respect to the MDC.
1, 19, 37, 39 40, 64 84, 102	V_{DD}	–	Positive Supply 5V ± 5%
2, 18 26, 38 63, 65 66, 72 78, 90 101, 103 104, 107 127, 128	GND	–	Ground 0 Volts

2.0 Introduction

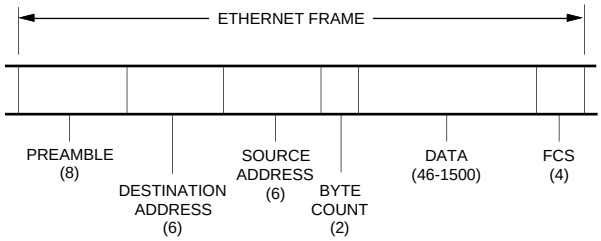
The 80C300 is an Ethernet Media Access Controller (MAC) with a rich set of operating modes and features. It is manufactured as a single-chip VLSI device to simplify and enhance the development of multi-port Ethernet embedded systems such as bridges, switches, and routers.

Two input/output paths are provided for interfacing to physical layer devices. In IEEE-standard MII mode, the 80C300 provides an industry standard interface supporting both 10Mbit/sec and 100Mbit/sec data rates. This interface will directly connect with physical layer devices such as SEEQ's 80C240 100Base-T4 PHY without additional glue logic. In Serial mode, the chip supports the standard Ethernet CSMA/CD protocol via a serial interface for transmit and receive data. Support for both Half and Full Duplex operation is provided in all interface modes.

The 80C300 is feature compatible with SEEQ's 84C300A Ethernet Media Access Controller. These features include: 64 bit Multicast filter, Transmit no CRC, Transmit no Preamble, Transmit Packet Autopadding, Receive CRC, Receive Own Transmit Disable, Receive Group Address Mode, Fast Receive Discard Mode, and Full Duplex Mode. Additionally, the 80C300 supports: programmable defer time between transmit packets, appending value of FCS on a packet-by-packet basis, and pin-controllable per-port receive packet abort.

A high-bandwidth universal system interface is provided which is compatible with many microprocessor or system busses, easing the integration of the 80C300 into many system architectures. Its 32-bit data path width is provided to provide the bandwidth necessary to maintain Full Duplex wire speed communications. Two 128 byte FIFOs are provided to ease bus multiplexing and interfacing to different clock domains.

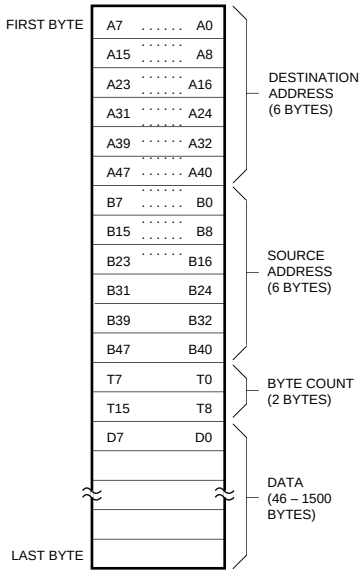
3.0 Functional Description



NOTE:
Field length in bytes in parentheses.

3.1 FRAME FORMAT

On an Ethernet communication network, information is transmitted and received in packets or frames. An Ethernet frame consists of a preamble, two address fields, a byte-count field, a data field and a frame check sequence (FCS). Each field has a specific format which is described in detail below. An Ethernet frame has a minimum length of 64 bytes and a maximum length of 1518 bytes exclusive of the preamble. The Ethernet frame format is shown in the figure above.



Typical Frame Buffer Format for Byte-Organized Memory

Preamble: *The preamble is a 64-bit field consisting of 62 alternating "1"s and "0"s followed by a "11" End-of-Preamble indicator.*

Destination Address: *The Destination Address is a 6-byte field containing either a specific Station Address, a Broadcast Address, or a Multicast Address to which this frame is directed.*

Source Address: *The Source Address is a 6-byte field containing the specific Station Address from which this frame originated.*

Byte-Count Field: *The Byte-Count Field consists of two bytes providing the number of valid data bytes in the Data Field, 46 to 1500. This field is uninterpreted at the Data Link Layer, and is passed through the EDLC chip to be handled at the Client Layer.*

Data Field: *The Data Field consists of 46 to 1500 bytes of information which are fully transparent in the sense that any arbitrary sequence of bytes may occur.*

Frame Check Sequence: *The Frame Check Sequence (FCS) field is a 32-bit cyclic redundancy check (CRC) value computed as a function of the Destination Address Field, Source Address Field, Type Field and Data Field. The FCS is appended to each transmitted frame, and used at reception to determine if the received frame is valid.*

3.2 PACKET TRANSMISSION

The transmit data stream consists of the Preamble, four information fields, and the FCS which is computed in real time by the chip and automatically appended to the frame at the end of the data. The Preamble is also generated by the chip and transmitted immediately prior to the Destination Address. Destination Address, Source Address, Type Field and Data Field are prepared in the buffer memory prior to initiating transmission. The chip encapsulates these fields into an Ethernet frame by inserting a preamble prior to these information fields and appending a CRC after the information fields. The chip can be programmed to exclude inclusion of the preamble and/or the FCS from the transmit data stream. In this case, it is assumed that the preamble and FCS are provided as part of the data written to the channel.

3.2.1 Controlling Transmit Packet Encapsulation

As was mentioned in the previous paragraph, the chip can be programmed for exclusion of the FCS and/or the preamble when transmitting a packet. To program the chip for transmitting a packet without creating a preamble, bit #2 of the Configuration Register #1 should be written high. Once this bit is set, all packets transmitted by the chip will not include a preamble pattern unless it is part of the data written to the transmit FIFO by the system. Similarly, the chip can be prevented from appending an FCS value to a packet by setting bit #4 of the Configuration Register #1. As long as this bit is high, any packet transmitted will not include an FCS value unless it is written as part of the transmit data written to the transmit FIFO. Appending of a FCS value can be controlled on a packet per packet basis by using the TXNOCRC pin as long as the TXNOCRC Tx-Rx Configuration register bit has not been set high. If the TXNOCRC pin is held high (or) if the TXNOCRC bit is set anytime during the duration of a packet write to the transmit FIFO, that particular packet will not be appended with a CRC value.

Transmit No CRC		CRC Appendage
H/W Pin 52	S/W Bit 4 of Config 1	To the Packet
0	0	Yes
0	1	No
1	0	No
1	1	No

Please note that both the H/W pin and the software bit should be kept deasserted during the entire duration of the packet write to the transmit FIFO in order to transmit a packet with CRC.

3.2.2 Transmission Initiation in Full Duplex and CSMA/CD Networks

Packet transmission begins with one of the following conditions:

1. *Data in the transmit FIFO meets or exceeds a user-defined threshold value or,*

2. *An EOF is asserted with the last data word written into the transmit FIFO.*

The transmit threshold value is controlled by programming bits 7 through 4 of the Configuration Register #3. The default value is 0 (zero), which enables the MAC to begin packet transmission with as little as one double word in the FIFO. The threshold, measured in double words, is equal to the number programmed into Transmit Control Register times 2. Thus, if the upper four bits of the register contain the value 3hex, then the transmission is defered until there are at least 6 double words of data in the FIFO.

Packet transmission initiation is also dependent upon whether the 80C300 is in Full Duplex or CSMA/CD mode. If the Chip is in CSMA/CD mode, transmission may also be prevented or delayed due to activity on the shared network medium. If the network is not busy due to other data traffic,

transmission will begin after the appropriate defer time (from end of previous traffic) has expired. Otherwise, transmission is delayed until after current data transfers are complete, and the defer time requirements have been satisfied. Following the IEEE 802.3 specifications, the minimum defer time is split into two periods. The beginning of the defer time occurs upon the transmitter sensing carrier sense going LOW. Once this case occurs then if carrier sense is reasserted during the 1st period of the defer time, the transmitter will reset its defer time counter and restart the total defer timeout period from 0. If carrier sense is reasserted during the 2nd period of the total defer time interval, the transmitter will ignore carrier sense and start transmission as soon as the defer time is met. The 1st period of the total defer time is programmable through use of the transmit defer register. The second period of the defer time interval is either 3.2 μs or 0.32 μs depending on whether the chip is in 10Mbit/sec or 100Mbit/sec mode. The total default defer time for 10Mbit/sec serial mode is 9.6 μs as measured from TXEN going LOW to TXEN going High assuming the transmit defer register is at 00 hex and assuming that the TXEN going LOW to CSN going LOW delay of the physical device is less than 5 TXC clock periods. When the chip is in Full Dulex mode, transmission of data onto the network occurs independent of whether carrier sense indicates a busy network condition or not.

Because of the variability in delays given for TXEN going LOW to CSN going LOW for different 100Mbit/sec physical devices, the default defer time in 100 Mbit/sec MII mode has been set assuming full duplex conditions where carrier sense is not monitored by the transmitter. In this case the default is 0.96 μs from TXEN going LOW to TXEN going HIGH. To adjust the defer time to some other value, the programmable defer register can be set using the formulas given in the section describing the defer register. When transmission begins, the chip activates the transmit enable (TXEN) line concurrently with the transmission of the first bit, or first nibble in the MII case, of the Preamble and keeps it active for the duration of the transmission.

3.2.3 Collision on transmit

On the occurrence of a transmit collision condition that does not represent the 16th transmission attempt for the packet or does not occur after 64 byte times into the transmission, the controller will automatically attempt to retransmit the packet. First, the controller will halt the transmission of data from the FIFO and begin transmitting a Jam pattern consisting of 55555555 hex. The controller will also reset the Transmit FIFO read address pointer back to the beginning of the transmit packet within the FIFO. At the end of transmitting the Jam pattern the controller will then begin the Backoff wait period. Once the backoff period is finished, the controller will automatically retransmit the packet. If a packet reaches 16 retransmission attempts without success due to collisions, or if a collision occurs later than 64 Byte times after the beginning of a transmission, this is considered to represent a serious network error. Upon any one of these two error conditions occurring the Transmit FIFO will be cleared and the TXRET output will be driven HIGH. If the TXRET signal was driven HIGH due to 16 transmission attempts, the T16COLL signal will also be driven HIGH. When either of the two above error conditions occurs, retransmission of any packets that were in the transmit FIFO requires first clearing the TXRET error condition and then reloading the packet or packets in the Transmit FIFO.

Scheduling of retransmission is determined by a controlled randomization process called Truncated Binary Exponential Backoff. The chip waits a random interval between 0 and 2^K slot times (51.2 μs per slot time for 10 Mbit Ethernet or 5.12 μs per slot time for 100 Mbit Ethernet) before attempting retransmission, where "K" is the current transmission attempt number (not to exceed 10).

TXRET conditions that will reset the backoff counter

Case 1: During packet transmission, a carrier sense dropout occurs but no collision occurs.

Case 2: 16 attempts to transmit a packet failed due to collisions.

3.2.4 Transmit Termination Conditions

The chip will terminate transmission under the following conditions.

***Normal:** The frame has been transmitted successfully without contention. Loading of the last data byte into a Transmit FIFO is signaled to the chip by activation of its RxTxEOF signal concurrently with the last double word of data loaded into the Transmit FIFO. This line acts as the thirty-third bit in the Transmit FIFO. After the last valid byte of the last double word has been transmitted, if the chip is not in Transmit No CRC mode, then the CRC is appended and transmitted concluding frame transmission. The Transmission Successful bit of the Transmit Status Register will be set by a normal termination.*

***Collision:** Transmission attempted by two or more Ethernet nodes. The Jam sequence is transmitted, the Collision status bit is set, transmit Collision Counter is updated, the Backoff interval begun, and the Transmit FIFO address is set to point to the beginning of the packet for retransmission automatically.*

***Underflow:** Transmit data is not ready when needed for transmission. Once transmission has begun, the chip on average requires one transmit double word every 3200 ns for 10 Mbit Ethernet or 320 ns for 100 Mbit Ethernet in order to avoid Transmit FIFO underflow (starvation). If this condition occurs, it terminates the transmission, issues a TXRET signal, and sets the Transmit-Underflow status bit.*

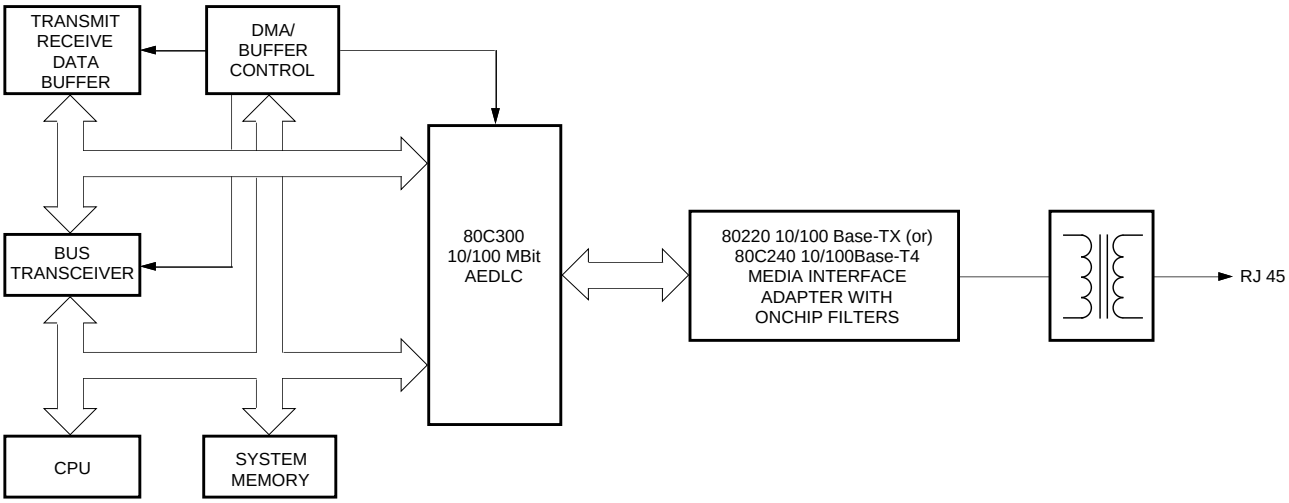


Figure 3. Typical Application Example

16 Transmission Attempts: *If a Collision occurs for the sixteenth consecutive time, the 16-Transmission-Attempts status bit is set, the Collision status bit is set, the TXRET and TI6COLL signals are generated, and the Backoff interval is begun. The counter that keeps track of the number of collisions is modulo 16 and therefore rolls over on the 17th collision. The Transmit Attempt Register described on page 30 allows the user to determine how many transmission attempts were required to successfully transmit the packet.*

Late Collision: *If a Collision occurs greater than 64 byte times after the transmission begins, this is considered a late collision error. Upon this condition, the transmission is terminated, the TXRET output is driven HIGH, and the late collision status bit is set.*

At the completion of every transmission or retransmission, new status information is loaded into the Transmit Status Register. Dependent upon the bits enabled in the Transmit Command Register, an interrupt will be generated for the just completed transmission.

3.2.5 Error Conditions That Will Cause TXRET to go HIGH

Detection of a HIGH value on the TXRET pin indicates that the chip could not complete transmission of a packet due to one or more of the following conditions:

1. *A transmit FIFO underflow occurred while transmitting the packet.*
2. *A late collision occurred while transmitting the packet.*
3. *Carrier sense never went active during transmission or went from an active to inactive state during transmission.*
4. *16 attempts to transmit the packet all resulting in transmit collisions.*

Any of the above conditions will cause the chip to flush the transmit FIFO and initiate a transmit retry request. With initiation of a transmit Retry Request, the TXRDY output will go low and stay low until the TXRET flag is cleared. Similar to the receive discard signal, the transmit retry signal going to the external TXRET pin is latched upon a transmit retry condition and held high until cleared. Until the transmit retry signal is cleared, no new transmit packets can be written to the transmit FIFO.

3.2.6 Detection and Clearing a Transmit Retry Condition

To enable the output driver for the TXRET pin, the $\overline{TXINTEN}$ input is driven low. Once a Tx retry condition is detected, the internal Tx retry signal should be cleared. Then by driving the $\overline{TXINTEN}$ input low and then pulsing the CLRTXERR input high for a minimum of one RXRD_TXWR clock cycle, the TXRET signal is cleared. The $\overline{TXINTEN}$ input must not change during the high time of the CLRTXERR input.

3.3 Packet Reception

The chip continuously monitors the network. When activity is recognized via the Carrier Sense (CSN) signal in 10 Mbit/sec Serial Mode, or through the Receive Data Valid (RX_DV) signal in MII mode, the chip will then synchronize itself to the incoming data stream through recognition of the Start Frame Delimiter (SFD) at the end of Preamble. The destination address field of the frame is then examined. Depending on the Address Match Mode specified, the chip will either recognize the frame as being addressed to itself in a general or specific fashion or abort the frame reception.

3.3.1 Preamble Processing

The chip recognizes activity on the Ethernet via its Carrier Sense (CSN) line in 10 Mbit/sec Serial Mode or through its Receive Data Valid (RX_DV) line in MII mode. In 10 MBit/sec Serial Mode, the end of preamble is detected by a double 1 serial receive data pattern preceded by 6 bits of alternating 1's and 0's. In MII mode, the end of preamble is recognized by the following nibble pattern:

	Logic Values
RXD3	0 1
RXD2	1 1
RXD1	0 0
RXD0	1 1

In 10 MBit/sec Serial Mode, detection of a double 0 pattern 16 bit times after CSN goes high and before a proper Start Frame Delimiter pattern is received, will prevent reception of the packet by the receiver. In MII mode, when RX_DV goes high the RXD[3:0] lines must be driven with 0 or more bytes of proper preamble pattern followed by one byte of proper SFD.

3.3.2 Address Matching

Ethernet addresses consist of two 6-byte fields. The first bit of the address signifies whether it is a Station Address or a Multicast/Broadcast Address.

First Bit	Address
0	Station Address (Physical)
1	Multicast/Broadcast Address (logical)

Address matching occurs as follows:

Station Address: All destination address bytes must match the corresponding bytes found in the Station Address Register. If Group Address mode is enabled, the last 4 bits of the station address are masked out during address matching.

After computing the FCS on the first six bytes of the address field (Destination address), the chip uses bits 0 thru 5 as an address to its Multi-cast address filter register. Bit 0 of the FCS is assumed to be where receive data enters the FCS generation circuitry. If the corresponding bit addressed in the Multicast address filter register is a '1' the chip will receive the frame, otherwise it will discard the frame. Addressing of the Multicast address filter register occurs using bits 0 thru 2 to determine which byte is selected and bits 3 thru 5 to determine which bit according to the following tables:

FCS Bits 0 1 2	Byte Selected	FCS Bits 3 4 5	Bit Selected
0 0 0	Byte 0	0 0 0	Bit 0
0 0 1	Byte 1	0 0 1	Bit 1
0 1 0	Byte 2	0 1 0	Bit 2
0 1 1	Byte 3	0 1 1	Bit 3
1 0 0	Byte 4	1 0 0	Bit 4
1 0 1	Byte 5	1 0 1	Bit 5
1 1 0	Byte 6	1 1 0	Bit 6
1 1 1	Byte 7	1 1 1	Bit 7

Multicast Address: If the first bit of the incoming address is a 1 and the chip is programmed to accept Multicast Addresses without using Hash filtering, the frame is received. The chip can also be programmed to use the hash filter for determining acceptance of multicast addresses.

Broadcast Address: The six incoming destination address bytes must all be FF hex. If the chip is programmed to accept Broadcast or Multicast Addresses, the frame will be received.

If the incoming frame is addressed to the chip specifically (Destination Address matches the contents of the Station Address Register), or is of general or group interest (Broadcast or Multicast Address), the chip will pass the frame exclusive of Preamble and FCS to the CPU buffer and indicate any error conditions at the end of the frame. If, however, the address does not match, as soon as the mismatch is recognized, the chip will terminate reception and discard the packet. All packet discards due to address mismatches are handled within the chip and will not cause the RXDC pin to go HIGH. This is true because determination of an address mismatch condition occurs before any portion of a receive packet is written to the receive FIFO.

The chip may be programmed via the Match Mode bits of the Receive Command Register to ignore all frames (Disable Receiver), accept all frames (Promiscuous mode), accept frames with the proper Station Address or the Broadcast Address (Station/Broadcast), or accept all frames with the proper Station Address, the Broadcast Address, or all Multicast Addresses (Station/Broadcast/Multicast).

3.6.1 Internal Channel Register Addressing Table

Config. 3 Bit 0	Register Address					Register Description[2]	
						BUSSIZE = 1, BE1 = 0, BE0 = 0	
						BUSSIZE = 1, BE1 = 0, BE0 = 1	BUSSIZE = 1, BE1 = 1, BE0 = 0
	A5	A4	A3	A2	A1	BUSSIZE = 0, A0 = 1	BUSSIZE = 0, A0 = 0
0	0	0	0	0	0	Transmit Late Collision Counter	Transmit Excessive Defer Counter
0	0	0	0	0	1	Over Flow Rx Error Counter	Attempt Counter
0	0	0	0	1	0	Unused	Unused
X	0	0	0	1	1	Tx Command/Status Register[1]	Rx Command/Status Register[1]
X	0	0	1	0	0	Hash Register 1	Hash Register 0
X	0	0	1	0	1	Hash Register 3	Hash Register 2
X	0	0	1	1	0	Hash Register 5	Hash Register 4
X	0	0	1	1	1	Hash Register 7	Hash Register 6
X	0	1	0	0	0	Product I.D. Register[4]	Configuration Register #3
X	0	1	0	0	1	Configuration Register #2[3]	FIFO Threshold Register
X	0	1	0	1	0	Defer Count Register	Configuration Register #1
0	0	1	0	1	1	Tot Rx Multicast Hi Byte	Tot Rx Multicast Lo Byte
0	0	1	1	0	0	CRC Error Counter Hi Byte	CRC Error Counter Lo Byte
0	0	1	1	0	1	Runt Frame Counter Hi Byte	Runt Frame Counter Lo Byte
0	0	1	1	1	0	Short Event Counter Hi Byte	Short Event Counter Lo Byte
0	0	1	1	1	1	Tot Rx Broadcast Hi Byte	Tot RX Broadcast Lo Byte
0	1	0	0	0	0	Alignment Error Counter Hi Byte	Alignment Error Counter Lo Byte
0	1	0	0	0	1	Receive Collision Counter Hi Byte	Receive Collision Counter Lo Byte
0	1	0	0	1	0	MII Management Interface Data Register	
0	1	0	0	1	1	MII Management Interface Command/Status Register	
0	1	0	1	0	0	Tot Rx Unicast Hi Byte	Tot Rx Unicast Lo Byte
0	1	0	1	0	1	Transmit Collision Counter Hi Byte	Transmit Collision Counter Lo Byte
0	1	0	1	1	0	Receive Oversize Frame Counter	Very Long Event Counter
0	1	0	1	1	1	Tx Defer Count Hi Byte	Tx Defer Count Lo Byte
0	1	1	0	0	0	Tx Unicast Pkt Count Hi Byte	Tx Unicast Pkt Count Lo Byte
0	1	1	0	0	1	TX Excessive Collision Counter	RXERR Error Counter
0	1	1	0	1	0	Tx Broadcast Pkt Count Hi Byte	Tx Broadcast Pkt Count Lo Byte
0	1	1	0	1	1	Tx Mcast Pkt Count Hi Byte	Tx Mcast Pkt Count Lo Byte
0	1	1	1	0	0	Rx Byt Count Byte 1	Rx Byt Count Byte 0
0	1	1	1	0	1	Rx Byt Count Byte 3	Rx Byt Count Byte 2
0	1	1	1	1	0	Tx Byt Count Byte 1	Tx Byt Count Byte 0
0	1	1	1	1	1	Tx Byt Count Byte 3	Tx Byt Count Byte 2
1	0	0	0	0	0	Station Addrs Reg 1	Station Addrs Reg 0
1	0	0	0	0	1	Station Addrs Reg 3	Station Addrs Reg 2
1	0	0	0	1	0	Station Addrs Reg 5	Station Addrs Reg 4

Notes: 1. These registers when written are the Transmit & Receive Command Registers and when read are the Transmit & Receive Status Registers.
2. All the counter registers are read only. All other registers except for the Transmit and Receive Command/Status Registers and the MII Management Interface Registers are both readable and writable.
3. Bit 0 of this register is write only and will always read back as a ZERO.
4. The product I.D. Register is an 8 Bit read only register (please refer to Revision Comparison MD400168 for details). It can be read as described in the above table.

3.3.3 Conditions of Receive Termination

Reception is terminated when either of the following conditions occur:

Carrier Sense or Receive Data Valid Inactive: *Indicates that traffic is no longer present on the Ethernet cable.*

Overflow: *The host node for some reason is not able to empty the Receive FIFO as rapidly as it is filled, and an error occurs as frame data is lost. On an average, the Receive FIFO must be serviced every 3200 ns for 10 Mbit Ethernet or 320 ns for 100 Mbit Ethernet to avoid this condition.*

3.3.4 Using RXABORT pin to Terminate Reception

By pulsing the RXABORT pin high for a minimum of 1.5 RXC cycles, reception of a packet can be terminated. When reception of a packet is terminated this way, the Receive FIFO will be cleared and will stay cleared until carrier sense in 10 MBit Serial Mode or Receive Data Valid in MII mode, transitions from high to low or from low to high indicating either the end of the packet being aborted or the beginning of a new receive packet. It is important to note that RXABORT will cause the RXDC pin to go HIGH based on the conditions described in the section "Conditions that Cause the RXDC Pin to go High".

The assertion of RXDC is done so that an external processor will always have an indication of a packet abortion irrespective of whether its aborted by the user or by an external PHY. However, the assertion of the RXDC signal can be avoided by setting bit 4 of configuration register #2. This will enable the reception of any packet irrespective of whether the RXABORT pin goes HIGH during packet reception.

3.3.5 Receive Discard Conditions

Receive packets can be discarded for not meeting the minimum IEEE 802.3 requirements for a good packet, for address mismatches when the chip is not in promiscuous mode, and by either user intervention or symbol errors occurring from a 100 Mbit/sec physical device. in the case of discards due to oversized packets, address mismatches, or the assertion of the RXABORT pin during packet reception, further writing of receive packet data to the receive FIFO is halted once the mismatch, receive abort or oversized packet condition is determined.

Except for discards due to address mismatches and oversized packets, all packet discards occur after carrier sense, or Receive Data Valid in MII mode, deasserts. The discarding of receive packets for error conditions can be controlled through bits 0 through 3 of the receive command register, and through bit 4 of configuration register #2. Listed below are the required conditions for a receive discard to be produced:

1. *Bit 0 of the Rx command register is LOW and a receive FIFO overflow occurred during reception.*
2. *Bit 1 of the Rx command register is LOW and a packet with a CRC error was received.*
3. *Bit 2 of the Rx command register is LOW and an oversized packet is received.*
4. *Bit 3 of the Rx command register is LOW and a packet with less than 64 bytes of data was received.*
5. *The Receiver is not in promiscuous mode and a address mismatch occurs.*
6. *Bit 4 of Configuration register 2 is LOW and the RXABORT pin is driven high while CSN is high.*

Discarding of a receive packet will cause any packet data that was written to the receive FIFO to be flushed from the FIFO. If no completely received packets are in the receive FIFO at the time a receive discard occurs, the receive FIFO will be completely flushed of data. If however a completely received packet, as indicated by the packet's status double word having been written to the FIFO, is in the receive FIFO at the time of a receive discard, the FIFO will be flushed only up to the last completely received packet. To prevent a receive packet from being discarded due to an error condition, you can selectively enable the reception of errored packets as described in the section "Receive Command Register".

Conditions that Cause the RXDC Pin to go HIGH

As packets are discarded due to the receive packet error conditions given in the section "Receive Discard Conditions", the RXDC pin may or may not assert. If a receive packet's status has been written to the receive FIFO and the packet's status has not yet been read from the FIFO, discards caused by following packets with errors are handled within the chip and the RXDC pin will not go HIGH. If all status double words for all packets written to the FIFO have been read out, then the RXDC pin will go HIGH under the following condition:

1. *Enough of a receive packet has been written to the FIFO to cause RXRDY to go HIGH before the packet is discarded due to an error condition.*
2. *If there are no status double words in the receive FIFO and if RXRDY goes HIGH just before a discard condition occurs, RXRDY may go LOW again before any FIFO reads have occurred. This is due to the receive discard clearing the FIFO of any receive bytes already written to the FIFO. In this case, RXRDY is guaranteed to remain HIGH for at least one RXRD_TXWR clock cycle.*

Detecting and Clearing a Receive Discard Condition

To enable the output driver for the RXDC pin, the RXINTEN input must be driven low. Once a discard condition is detected, the receive discard can be cleared

by driving the RXINTEN input low and then pulsing the CLRRXERR input high for a minimum of one RXRD_TXWR clock cycle. The RXINTEN input must not change state for the duration of the time that the CLRRXERR input is high.

Clearing Interrupts

Both receive and transmit interrupts are combined into a single interrupt signal which then goes to the INT output pin. The interrupt signal in the chip is actually the result of the receive/transmit status register outputs and the receive/transmit command register interrupt enable bits that are set. To clear an interrupt, the status that caused the interrupt needs to be cleared. This can be accomplished by reading the transmit status register and/or the receive status register.

3.4 SYSTEM INTERFACE

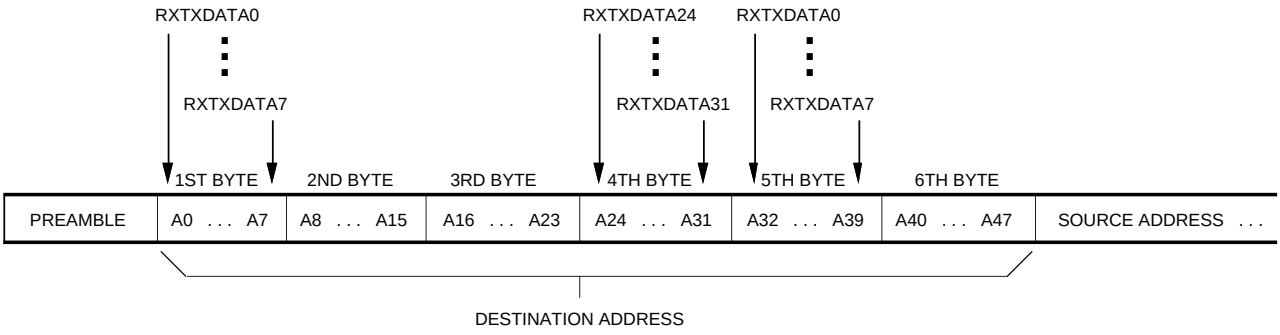
The Chip's system interface consists of one receive/transmit 32-bit bidirectional data bus, one 8/16-bit bidirectional command/status data bus, and the control signals of the respective buses. Receive FIFO data is read and Transmit FIFO data is written over the RXTXDATA[31:0] bus, and Command/Status data is written or read over the bidirectional CDST[15:0] data bus.

80C300 register accesses can be set to 16-bit or 8-bit mode by tying the BUSSIZE pin HIGH or LOW. In 16-bit mode it is still possible to perform byte wide accesses by using the BE[1:0] byte enable pins. The BUSSIZE and BE[1:0] pins are used only by the 80C300's register interface and are not part of the FIFO interface control.

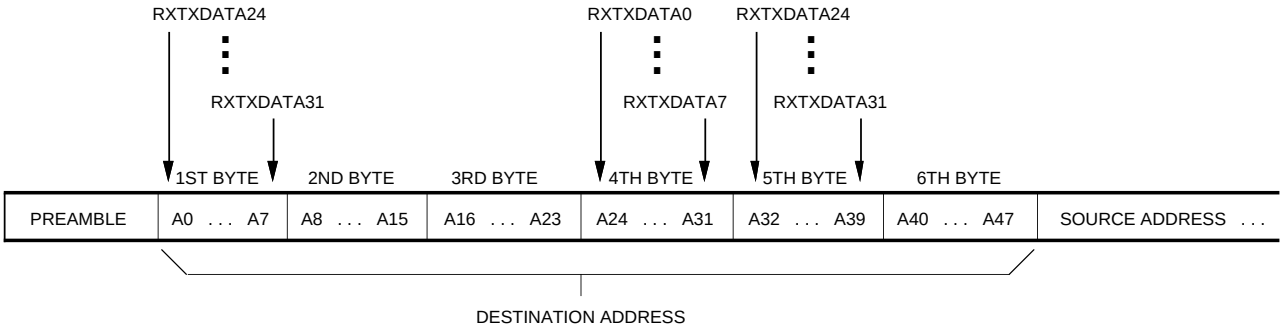
3.5 FIFO Format

3.5.1 Little Endian and Big Endian Format

The FIFO interface control includes the BUSMODE pin which sets the 80C300 FIFO interface to Big Endian or Little Endian byte transmit/receive data order. In Big Endian mode, data written to the transmit FIFO is transmitted most significant byte of the RXTXDATA bus first and least significant byte of the RXTXDATA bus last. In Little Endian mode, the least significant byte of each double word is transmitted first and the most significant byte of each double word is transmitted last. The figure shown below depicts the difference in byte transmission order for Big Endian verses Little Endian mode. On the receive side, if Big Endian mode is in effect then the first data bytes received are assumed to be the most significant bytes of



Bit Serialization/Deserialization for Little Endian Format



Bit Serialization/Deserialization for Big Endian Format

the double word and appear on the most significant portion of the RXTXDATA bus for receive FIFO reads. The receiver reverses this order if the chip is in Little Endian mode. The value of the BUSMODE pin has no effect on the operation of the 80C300's register interface.

3.5.2 Transmit FIFO Interface

To determine if the transmit FIFO has reached its threshold number of double words of space available, the TXRDY output can be enabled by driving the TXINTEN input low. The TXRDY output will be high if there is enough space available in the transmit FIFO to meet or exceed the programmed threshold value.

Once the TXRDY output is determined to be high, the transmit FIFO can be written. To write to the Transmit FIFO, the TXWREN and TXINTEN inputs must be asserted low and at least one of the RXTXBE byte enables must be low for each write cycle. All of the above inputs are clocked into the chip on the high going edge of the RXRD_TXWR clock input which also acts as the FIFO write strobe. Because of this pipe lining, the actual FIFO write will occur one RXRD_TXWR cycle after the assertion of the Transmit FIFO interface control signals. Valid combinations of the RXTXBE inputs for transmit FIFO writes are given below:

RXTXBE3	RXTXBE2	RXTXBE1	RXTXBE0
0	0	0	0
1	0	0	0
0	0	0	1
1	1	0	0
1	0	0	1
0	0	1	1
1	1	1	0
1	1	0	1
1	0	1	1
0	1	1	1

The TXRDY output will remain high until the chip's transmit FIFO no longer has enough double word space to meet the programmed threshold value.

While transmit FIFO writes are occurring, the SPDTAVL output will remain high until the highgoing edge of the write to the second to the last remaining double word space in the FIFO. Because transmit FIFO writes are pipelined, there will always be one more FIFO write after TXWREN is deasserted externally.

Using the 80C300 in 8 bit or 16 bit mode.

The transmit and the receive FIFO are 128 bytes deep organized as double word (32 bits) rows. During writes to the transmit FIFO, the FIFO pointer gets incremented on every write to the FIFO, irrespective of whether all the four byte enables are asserted or not. Hence, during non double word writes to the FIFO, one entire row of the FIFO gets filled irrespective of whether all the bytes are valid or not. The 80C300 automatically ignores the invalid bytes when the data gets transmitted from the FIFO.

Effect of Auto Retransmission Upon TXRDY Behavior

As a packet is read out of the Transmit FIFO by the transmitter for transmission onto the network, the TXRDY signal will not reflect any reads that have occurred to the FIFO until enough bytes of data have been transmitted to get past the normal collision window of less than 64 byte times. This means that if TXRDY goes low during the writing of a packet to the Transmit FIFO, it will not go HIGH again until both of the following conditions are true:

1. *The packet has been completely transmitted or a point 64 byte times from the beginning of the transmission has been reached.*

2. *The number of bytes taken out of the transmit FIFO for transmission subtracted from the number of bytes written to the FIFO leaves the FIFO with enough double word space available to meet the threshold setting.*

It is important to note that until the packet is completely transmitted or until enough of the packet is transmitted to get past the normal collision window, the TXRDY output will only reflect how many writes have occurred and will not reflect how much of the FIFO data has been read out for transmission. Because of this it is important to insure enough packet data has been written to prevent FIFO underflows if there exists a large latency between the TXRDY output being determined HIGH and the writing of more data to the FIFO.

3.5.3 Receive FIFO Interface

To determine if the receive FIFO has reached its threshold number of double words of data, the RXRDY output can be enabled by driving the RXINTEN input low. The RXRDY output for the chip will be high under one of the following conditions:

1. *There are enough double words of data in the channel's receive FIFO to meet or exceed the programmed threshold value.*

2. The status double word for a receive packet with an end of frame value of HIGH is in the receive FIFO.

Once the RXRDY output is determined to be high, the receive FIFO can be read. To read from the Receive FIFO, the RXRDEN and RXINTEN inputs must be asserted low and the RXTXBE byte enables must be low for each read cycle. Similar to the Transmit FIFO interface, all of the above Receive FIFO interface control signals are clocked into the chip on the high going edge of the RXRD_TXWR clock input which also acts as the FIFO read strobe. Because of this pipe lining the actual FIFO read will occur one RXRD_TXWR cycle after the assertion of the Receive FIFO interface control signals.

Using the 80C300 in 8 Bit or 16 Bit Mode

On the receive side, two different modes are possible.

On burst reads (Rxrden being asserted for multiple clock cycles), if the first read is not a double word read, the second read will always increment the FIFO pointer irrespective of whether all the byte enables are enabled or not. In this mode, 16 bit reads are possible by muxing the LSB and the MSB of the data bus. 8 bit reads are not possible.

On single reads (Rxrden being asserted for only one clock cycle), the FIFO pointer will get incremented only on a double word read. In this mode, the different bytes of the data bus can be muxed to perform multiple 8 bit or 16 bit reads. But, all the reads of the bytes belonging to one row should be terminated with a double word read to increment the FIFO pointer.

When the chip is being read, the RXRDY output will remain high until the high going edge of the read that results in one of the following conditions:

1. The FIFO no longer has enough data to meet the threshold setting.
2. A packets status double word with its associated HIGH end of frame value is read out.

In the case of RXRDY being driven LOW upon condition two given above, it will remain LOW for 8 RXRD_TXWR clock cycles and then goes back HIGH if one of the conditions for RXRDY being HIGH is met.

During reads from the FIFO, the SPDTAVL output will remain high until the high going edge of the read that causes one of the following conditions to occur:

1. The read that empties the FIFO completely.
2. The read that reads a packets status double word from the FIFO.

In the case of SPDTAVL being driven low upon the high going edge of the read that meets one of the above conditions, the SPDTAVL output will remain low for a period of 8 RXRD_TXWR clock cycles. For the time that SPDTAVL remains low, further reads are blocked within the chip even if external reads continue. This allows overreading the receive FIFO by a few cycles without, internal to the chip, reading an empty FIFO or reading new packet data before the present packet is processed. It is up to the processor doing the FIFO reads to determine on which read cycle the SPDTAVL went low and thereby which read cycles are over reads containing invalid data.

3.5.4 Special Conditions on the RXRD_TXWR input

This input is required to be tied to a continuous clock signal whose maximum clock frequency can be 33Mhz. The number of read or write cycles occurring to the chip is controlled through the TXWREN and RXRDEN inputs. All transitions of the TXRDY, RXRDY, RXTXEOF, SPDTAVL, RXDC, RXTXDATA[31:0], T16COLL, RXBXT12 and TXRET outputs are synchronized internally to the RXRD_TXWR clock and are clocked to the output drivers on the highgoing edge of the clock.

3.6 REGISTER INTERFACE

Writing of Command, Configuration, and Station Address registers, and reading of status registers and management counters is controlled by the BUSSIZE, BE[1:0], ENREGIO, RD, WR, and A[5:0] pins. The ENREGIO signal is used as general register interface enable and must be active low before any register operations can occur. Setting the BUSSIZE control pin to HIGH or LOW sets the register interface to 16-bit mode or 8-bit mode respectively. The BE[1:0] pins are used to allow byte wide register accesses when the chip is in 16-bit mode. To facilitate programming when in 16-bit mode, 8-bit registers within the chip are organized in pairs so that two 8-bit registers can be accessed with a single 16-bit read or write or by using BE[1:0] pins to access them singly. In 16-bit mode the BE[1:0] pins control whether registers are accessed a byte at a time or a 16-bit word at a time and which portion of the CDST[15:0] is used. In 8-bit mode the BE[1:0] pins are not used and all accesses drive only the CDST[7:0] portion of the CDST data bus. The table below shows the states of the BUSSIZE and BE[1:0] pins, which portion of a two 8-bit register pair or single 16-bit register is accessed, and which part of the CDST[15:0] data bus is used.

BUSSIZE	$\overline{\text{ENREGIO}}$	$\overline{\text{BE0}}$	$\overline{\text{BE1}}$	A0	CDST[15:8]	CDST[7:0]
X	1	X	X	X	---	---
0	0	X	X	0	---	LOBYTE
0	0	X	X	1	---	HIBYTE
1	0	0	0	X	HIBYTE	LOBYTE
1	0	0	1	X	----	LOBYTE
1	0	1	0	X	HIBYTE	---
1	0	1	1	X	---	---

Addressing of registers within the chip is controlled directly through the A[5:0] address pins. Initiation of a register read is controlled by the RD pin and initiation of a register write is controlled by the WR pin. Even though the registers can be accessed at any time, it is recommended that writing to the command register, be done only during interframe gaps.

With the exception of the two Match Mode bits in the Receive Command Register, all bits in both command registers are interrupt enable bits. Changing the interrupt enable bits during frame transmission does not affect the frame integrity. Asynchronous error events, however, e.g., overflow, underflow, etc., may cause chip operation to vary, if their corresponding enable bits are being altered at the same time.

Reading the status registers may also occur at any time during transmission or reception.

Status Registers and all management counters are read only registers. The Rx and Tx Command Registers are write only and all other registers are writable and readable. Access to these registers is via the CPU interface: Control signals ENREGIO, RD, WR, and the Command/Status Data Bus CdSt [15:0].

3.6.2 Station Address Register

The Station Address Register is 6 bytes in length. The contents may be written in any order, with bit "0" of byte "0" corresponding to the first bit received in the data stream, and indicating whether the address is physical or logical. Bit 7 of station address byte 5 is compared to the last bit of the received destination address. The Station Address should be programmed prior to enabling the receiver.

3.6.3 Transmit Command Register

The transmit command register is an 8 bit write only register. Bits 0 through 3 of the Transmit Command Register function as interrupt mask bits, which provide for control of the conditions allowed to generate transmit interrupts. Each of the four bits may be individually set or cleared. When set, the occurrence of the associated condition will cause an interrupt to be generated. The four specific conditions for which interrupts may be generated are:

1. *A transmit FIFO underflow occurred while transmitting the packet.*
2. *A collision occurred while transmitting the packet.*
3. *A transmit error condition occurred i.e, (Carrier sense never went active during transmission or went from an active to inactive state during transmission, 16 collisions occurred for a transmit packet, or a late transmit collision occurred.*
4. *The packet was transmitted successfully.*

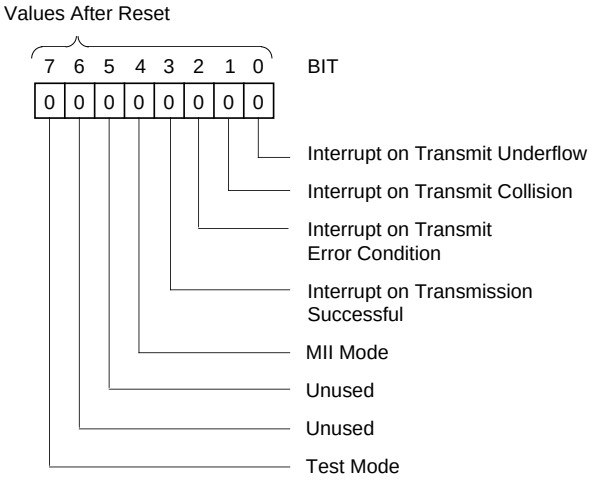
Interrupts are cleared by following the procedure given in the section entitled "Clearing Interrupts".

Bit 4 of the transmit command register is used to set the chip into MII mode. When this bit is written high the chip will use the MII interface for transmit and receive data.

Bits 5 and 6 are unused.

Bit 7 is used for testing purposes and should not be written high under normal circumstances

Transmit Command Register Format



3.6.4 Receive Command Register

The Receive Command Register has two primary functions, it specifies the Address Match Mode, and it specifies which types of receive frames will be received and if an associated interrupt will be produced. To set interrupt conditions, the Receive Command Register uses bits 5 through 0 in conjunction with bit #7 of configuration register #1.

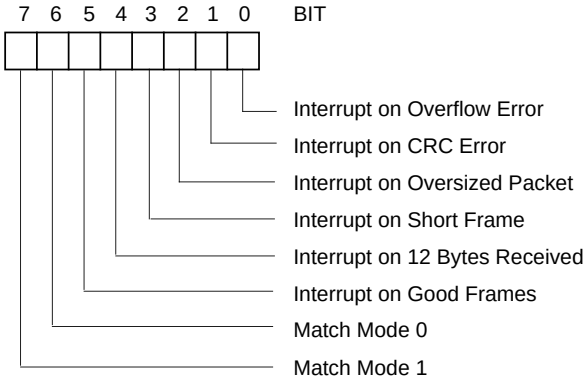
Bit 7 of configuration register #1 is a general receive interrupt disable. Setting this bit HIGH disables all receive interrupt conditions even if one of the bits 1 through 6 in the receive command register is set HIGH. This allows enabling reception of receive packets with errors without an interrupt being produced. With the general receive interrupt bit LOW, a receive interrupt can be produced on one or more of the following conditions by setting its associated interrupt enable bit in the receive command register:

1. *Interrupt on good frames (Rx Command Bit 5)*
2. *Interrupt on receiving the 12 bytes of data for a packet. (Rx Command Bit 4)*
3. *Interrupt on reception of a short frame. (Rx Command Bit 3)*
4. *Interrupt on reception of a oversized packet (>1518). (Rx Command Bit 2)*
5. *Interrupt on reception of a frame with a CRC error. (Rx Command Bit 1)*
6. *Interrupt on a receive FIFO overflow error. (Rx Command Bit 0)*

For conditions 3, 4, 5 and 6 above, the associated interrupt enable bit also acts as a receive enable bit. This means for example that if bit 1 of the receive command register is written high, packets with CRC errors will be received and no receive discard will be asserted. By using a combination of the general receive interrupt disable bit 7 in configuration register #1 and bits 0 through 3 of the receive command register, a port can be programmed to accept packets with error conditions without the generation of an interrupt.

Bits 6 and 7 of the Receive Command Register are the receive match mode bits.

Receive Command Register Format



Bits 0-5 specify Interrupt and Frame-of-Interest when set.

	Match Mode 1	Match Mode 0	Function
0	0	0	Receiver Disable
1	0	1	Receive All Frames
2	1	0	Receive Station or Broadcast Frames
3	1	1	Receive Station, Broadcast/Multicast Frames

Match Mode Definition

Changing the receive Match Mode bits during frame reception may change chip operation and give unpredictable results.

Interrupt Enable and Frames-of-Interest

Bits 0-5 when set specify interrupt generation on occurrence of the corresponding frame reception condition.

3.6.5 Transmit Status Register

Within the transmit section there are 2 transmit status registers. These registers give the appearance of a single register to an external CPU. With each transmission at-

tempt, whether successful or not, one of the status registers is written with the transmit status for that packet and bit 7 of that register is set to a 0 until both registers are full. When both registers are full, no new transmit status can be written until one of the registers is read. To an external CPU, both transmit status registers appear as a single register. If the CPU reads a LOW value for bit 7 of the transmit status register, this indicates that either one or both of the internal transmit registers contains new status. A delay time after the highgoing edge of the read operation that reads new transmit status, one of the internal transmit status registers will be cleared and made available for new transmit status. Following are the types of transmit status given through status register:

Bit 0 - Transmit FIFO Underflow Occurred.

Bit 1 - Collision during transmission occurred.

Bit 2 - 16 collisions occurred while attempting to transmit a packet.

Bit 3 - Packet transmitted successfully.

Bit 4 - Carrier Sense error during transmission attempt.

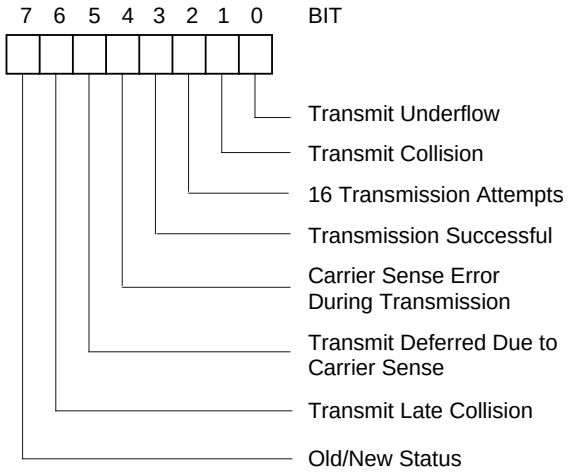
Bit 5 - Transmit Deferred Due to Carrier Sense

Bit 6 - A Late Transmit Collision Occurred.

Bit 7 - Old/New Status.

The chip can be programmed so that if both transmit registers are full, no new transmissions will occur until at least one of the registers is cleared by reading it. To program this feature, bit #1 of configuration register #2 needs to be written to a 1 value. Also, the chip can be programmed so that no transmit status is loaded if the Transmission is successful. This feature is programmed by setting bit #0 of configuration register #2 to a 1 value.

Transmit Status Register Format



3.6.6 Receive Status Register

Within the receive section, there is a receive status register that is written with the status of each receive packet whether it is discarded or not. Once the receive status register is written, bit 7 of the register is set to a 0 and the register is write protected from being overwritten with new status until it is read. Reading the receive status register clears the register and enables it to be written with new status. The following packet status is reported in the receive status register:

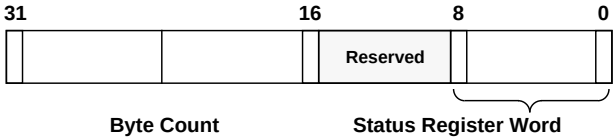
Bit 7 - Old/New status
Bit 6 - 12 bytes of a frame received.
Bit 5 - Frame received without errors.
Bit 4 - Oversized frame received.
Bit 3 - Short frame error.
Bit 2 - Frame with dribble bits or nibbles.
Bit 1 - Frame with CRC error.
Bit 0 - Receive FIFO overflow error.

Receive packet status is also included as part of the final double word of receive data for a packet that is not discarded. The final double word of a packet as read from the receive FIFO contains the status and the byte count for that packet with the status appearing as the least significant word of the double word and the byte count appearing in the two most significant bytes of the double word. The status read through the FIFO has the same bit values as the receive status register except for the following:

Bit 7: RXABORT During Reception
Bit 8: Read Error Condition

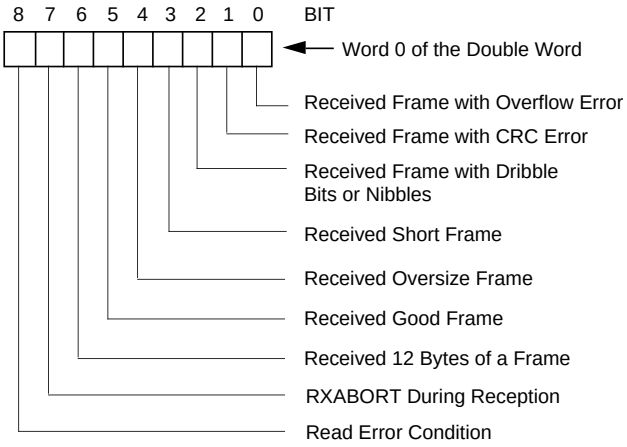
Bit 7 is an indication that the RXABORT pin was pulsed HIGH while CSN was HIGH for the packet. Bit 8 Indicates that some type of error has occurred in the receive FIFO control circuitry with a result that the number of double words written to the FIFO as indicated by the byte count portion of the status double word does not equal the number of double words read from the FIFO for the packet. This type of error can only be caused by some type of noise glitch or other unusual occurrence within the receive section. Any packet read from the FIFO with Bit 8 of the status set HIGH should be considered to have bad data. This condition should never occur in a properly designed application. If status is ever read with Bit 8 being HIGH, the receive section will automatically reset itself to provide a clean starting point for further packet reception.

Format of the Status Double Word



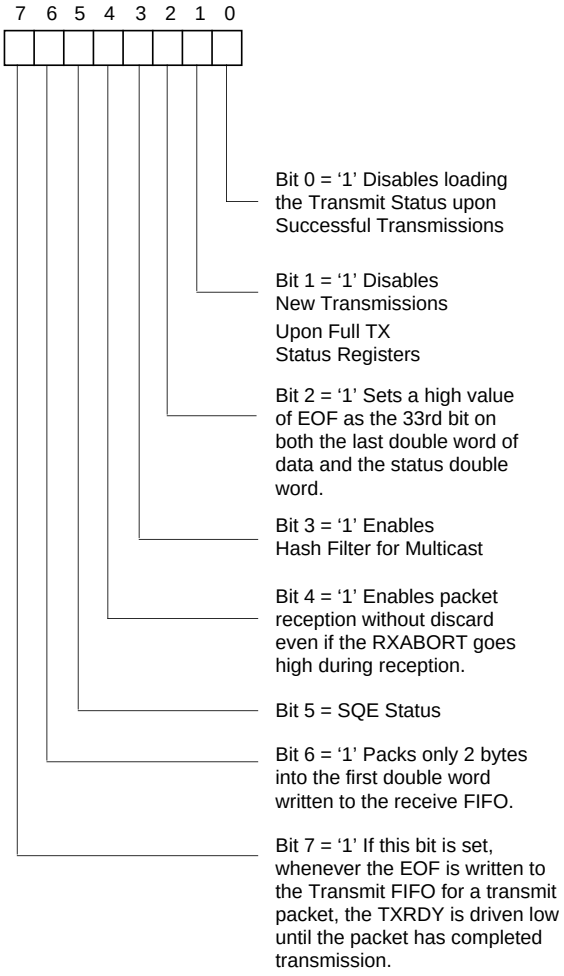
Note: This status double word gets appended to the packet in same format for both Little and Big Endian modes.

Status Register Word



3.6.7 Configuration Register #2

Allows for the following programmable features given below:



Don't Load Tx Status Upon Successful Transmit Mode
If bit #0 of configuration register #2 is set, then a packet that has been transmitted successfully will not have its status loaded into either of the two internal transmit status registers.

Disable Further Transmission Upon Full Tx Status Register Mode
If bit #1 of configuration register #2 is set, then whenever both Tx status registers have been filled, no new transmissions will occur until one of the TX status registers is cleared, even if the Transmit FIFO has transmit data.

EOF Bit Setting Control
If bit #2 of configuration 2 is set, the EOF bit is set both at the end of the last double word of ***data*** *and the* ***status*** *respectively. If this bit is not set, then the EOF bit is set only at the end of the status double word.*

Multicast Mode
The chip has a 64 bit multicast address filter register which can be accessed as shown in the Internal Channel Register Addressing Table (page 15). When it is programmed to receive multicast frames (match mode 3), after computing the CRC on the address field of the receiving frame (first 6 bytes), it will index to the multicast address filter register depending on bits 0 to 5 of the CRC. If the corresponding bit is a '1' it will receive the frame, otherwise it will discard the frame.

Reception of Packets with RXABORT Errors
If bit #4 of configuration #2 is set, any packet can be received without discard even if RXABORT goes high during reception.

SQE Status Bit
After transmitting a frame if the chip does not receive a collision with in a 4.0 µs period this bit will be set. Once set, this will stay set until cleared. This can be read and cleared as explained in the register section. The SQE test is only available in 10 Mbit/sec Serial Ethernet Mode.

Successful Packet Transmission Complete Feature
This feature is programmable by setting bit 7 of configuration register #2 to a "1" value. If this bit is set then, independent of the FIFO threshold setting, the TXRDY pin will go LOW once the final double word of data for a transmit packet is written to the transmit FIFO. Once TXRDY has been driven LOW due to this condition, TXRDY will remain LOW until the packet has completed transmission without error or until a transmission exception condition causing the TXRET pin to go HIGH is cleared. This allows the user to determine when a packet has completed successful transmission by detecting when TXRDY goes HIGH after the final double word of the packet has been written. After TXRDY goes LOW due to a double word write with the RXTXEOF pin HIGH, further writes to the transmit FIFO are allowed as long as the SPDTAVL pin indicates there is still space available within the transmit FIFO.

3.6.8.1 FIFO Threshold Register Settings Table

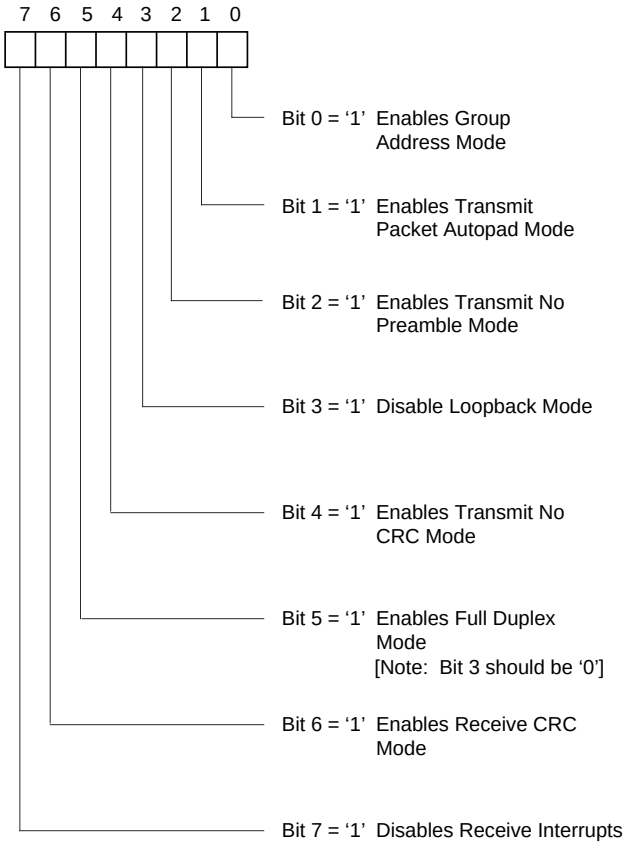
Fifo Threshold Register Bits								Minimum # of Double Words of Data for RXRDY High	Minimum # of Double Word Spaces for TXRDY High
7	6	5	4	3	2	1	0		
0	0	0	0	0	0	0	0	1	1
0	0	0	1	0	0	0	1	2	2
0	0	1	0	0	0	1	0	3	3
0	0	1	1	0	0	1	1	4	4
0	1	0	0	0	1	0	0	5	5
0	1	0	1	0	1	0	1	6	6
0	1	1	0	0	1	1	0	7	7
0	1	1	1	0	1	1	1	8	8
1	0	0	0	1	0	0	0	9	9
1	0	0	1	1	0	0	1	10	10
1	0	1	0	1	0	1	0	11	11
1	0	1	1	1	0	1	1	12	12
1	1	0	0	1	1	0	0	13	13
1	1	0	1	1	1	0	1	14	14
1	1	1	0	1	1	1	0	15	15
1	1	1	1	1	1	1	1	16	16

Pack Only Two Valid Bytes in First Receive Double Word

This feature is programmable by setting bit 6 of configuration register #2 to a "1" value. If this bit is set then the first double word of data written to the receive FIFO for a receive packet will have only two valid bytes. When this first double word is read out of the receive FIFO , which two bytes are valid depends upon whether the BUSMODE pin is tied HIGH or LOW. For the first double word read, only RXTXDATA[15:0] are valid if Busmode is HIGH, otherwise only RXTXDATA[31:16] are valid. All subsequent double words of data read from the receive FIFO will contain 4 valid bytes except for the last double word which may not have all 4 bytes valid.

Configuration Register #1

Allows for control of various transmit and receive features. Set to all 0's after reset.



Group Address Mode

In this mode, the last 4 bits of the serial receive data stream for the destination address are masked out in address comparison. This means that when the destination address is compared against the value programmed in the station address register, the packet will not be rejected due to incorrect address even if its last 4 bits did not match.

Transmit Packet Autopad Mode

This feature automatically pads packets to be transmitted with less than 60 bytes of data out to a minimum IEEE 802.3 standard packet length of 60 bytes excluding FCS. Padding is done with bytes of 00 hex in 10 Mbit/sec Serial Mode and 55 hex in MII mode.

Transmit No Preamble Mode

This mode prevents the transmitter from adding a preamble pattern at the beginning of data to be transmitted.

Disable Loopback Mode

Description on the Loopback mode (Bit #3 of Config 1)

The following description assumes that a transceiver is connected to the MAC.

Configuration Register #1

Bit 3	Bit 5	Mode	Functional Description
1	0 (Default)	Half Duplex	In this mode, the transmit data looped back from the transceiver is ignored by the controller. The data does not get written into the receive FIFO and the Rxrdy does not reflect the incoming data.
0 (Default)	1	Full Duplex	In this mode, the transceiver (In Full Duplex mode), will not loopback the transmitted data. However, since data reception is possible during transmission, bit 3 should be written with '0' so that the data gets written to the Receive FIFO.
0 (Default)	0 (Default)	Loopback Mode	In normal half duplex operation, the PHY loops back the transmitted data back to the MAC. In other words, the PHY always loops back the transmitted data in half duplex mode. As far as the controller is concerned, it knows that the data coming back is it's own transmitted packet and since bit 3 is not set, the transmitted packet gets written into the receive FIFO.

Note: There is no internal loopback within the MAC. Loopback is dependent on a PHY connected to the MAC.

Transmit No CRC Mode
This mode prevents the transmitter from appending transmit data with an FCS.

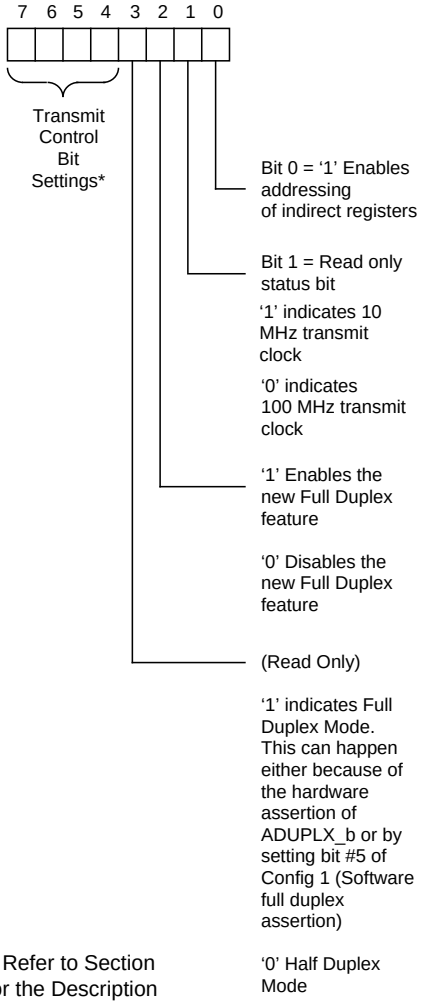
Full Duplex Mode
In this mode, the transmitter will ignore carrier sense and will not defer to it if it is ready to transmit a packet.

The software bit setting and the hardware setting (pin #108) have an OR relationship. This means that either the hardware or software setting will enable Full Duplex.

Receive CRC Mode
In this mode, the receiver loads the 4 bytes of FCS into the receive FIFO along with the data allowing the FCS value to be read out.

Disable Receive Interrupts
With this bit set, the receiver is disabled from producing receive interrupts.

Configuration Register #3
Provides indirect register addressing and speed detect status.



*Please Refer to Section 3.2.2 for the Description

New Full Duplex Feature: *In revisions A, & B, it is necessary to reset bit 3 of Configuration 1 whenever Full Duplex mode is selected. The new Full Duplex feature makes the setting of bit 3 automatic based on the mode selected. This new feature is present only in revisions C and onwards.*

3.6.8 FIFO Threshold Register
This register allows programming of the threshold of Space Available and/or Data Available double word counts that cause assertion of the TxRDY and/or RxRDY signals respectively. Bits 4 through 7, when written with a binary value, indicates the minimum number of double words necessary in the receive FIFO before RxRDY is asserted. Similarly, bits 0 through 3, when written with a binary value, indicate the minimum number of double word wide spaces necessary in the transmit FIFO for TxRDY to be asserted. Table 3.6.8.1 shows the number of double words of space/data that are required to cause the TXRDY/RXRDY signal to go high for each threshold setting.

3.6.9 Defer Register Calculations for the 80C300

Defer Time Definitions
In the standard Half Duplex Mode, Defer time is defined as the time from the falling edge of carrier sense to the rising edge of TXEN. In full duplex mode, the defer time is measured as the time from the falling edge of TXEN to the next rising edge of TXEN. The binary value programmed into the defer count register is used to determine how many byte times the defer time will be set to. The algorithms below illustrate how the defer time is calculated.

The defer time is split into two periods. The first period is the first 2/3 and the second period is the second 1/3 of the defer time. The defer time calculated by the following algorithms are for the first 2/3 of the defer period only. For further details, please refer to the section 3.2.2.

Algorithm for Defer Time Calculations for MII

Defer Time = Int{{Int (Delay /40) + 5 + DefRegSet}/2}+2

Defer Time = The transmit defer time in byte times

Delay = Delay from the falling edge of TXEN to the falling edge of CSN. (Half Duplex)
= 0 (Full Duplex)

DefRegSet = The transmit defer register setting

Int = Using the Whole Number Portion

Example Calculations

To find out the value that needs to be programmed into the defer register for a defer time of 960 ns, the following steps need to be taken

Assume Delay = 340 ns
Desired Defer Time = 960 ns = 12 byte times
Note: The desired defer time should be a multiple of 80

Step 1: Calculation of the Actual Defer Time

Let's assume a Defer Register Setting Value of 10

Defer Time = Int{ {Int (Delay / 40) + 5 + DefRegSet} /2} + 2

= Int { { Int (8.5) + 5 + 10 }/2} + 2

= Int { 11.5 } + 2

= 11 + 2 = 13 byte times

Step 2: Calculation of the Actual Defer Register Setting

Since we know that the value derived from the previous step is 1 byte time greater than what is desired we will decrement the assumed defer register setting by 3 and do the calculations again.

Let's assume a Defer Register Setting Value of 7

Defer Time = Int { {Int(Delay/40) + 5 + DefRegSet} /2} + 2

= Int { { Int(8.5) + 5 + 7 } /2 } + 2

= Int { 10 } + 2

= 12 byte times

Note: Please note that you might have to do this process several times before you can get the actual defer register setting for a desired defer time based on your delays.

Algorithm for Defer Time Calculations for 10 Mbit Serial Mode

Defer Time = Int {{Int (Delay/100)+17+DefRegSet}/8}+2

Defer Time = The transmit defer time in byte times

Delay = Delay from the down going edge of TXEN to the down going edge of CSN. (Half Duplex)
= 0 (Full Duplex)

DefRegSet = The transmit defer register setting

Int = Using the Whole Number Portion

Example Calculations

To find out the value that needs to be programmed into the defer register for a defer time of 9600 ns, the following steps need to be taken

Assume Delay = 3400 ns
Desired Defer Time = 9600 ns = 12 byte times
The desired byte times should be a multiple of 800

Step 1: Calculation of the Actual Defer Time

Let's assume a Defer Register Setting Value of 21

Defer Time = Int {{ Int (Delay / 100) + 17 + DefRegSet} /8} + 2

= Int { { Int (34) + 17 + 21}/8} + 2

= Int {9} + 2

= 9 + 2 = 11 byte times

Step 2: Calculation of the Actual Defer Register Setting

Since we know that the value derived from the previous step is 1 byte time lower than what is desired we will increment the assumed defer register setting by 8 and do the calculations again.

Let's assume a Defer Register Setting Value of 29

Defer Time = Int {{Int (Delay / 100)+17+DefRegSet} /8} + 2

= Int { { Int (34) + 17 + 29 } /8} + 2

= Int { 10 } + 2

= 10 + 2 = 12 byte times

Please note that you might have to do this process several times before you can get the actual defer register setting for a desired defer time based on your delays.

3.7 MANAGEMENT INTERFACE OF MII
Using the 80C300's MII Management Interface to Perform Register Operations to the Ethernet Physical Signaling Device (PHY)

The MII specification provides for a serial management interface to access registers within the PHY. This interface follows a specific serial data signaling protocol that is described in the IEEE approved MII specification. The management interface portion of the MII requires two signals MDC and MDIO. MDC is used as the serial data clock and MDIO is used for the serial data. All register

accesses to the PHY take place over these two signal lines. According to the MII specification two 5-bit addresses are required for uniquely addressing a register within a PHY. In the case of multiple PHY's, both a 5-bit physical address for the PHY and a 5-bit register address for a register within the PHY are used to perform reads and writes to registers within a PHY.

To simplify the task of using the management interface portion of the MII for register operations to or from a PHY, the 80C300 provides two special registers, the MII Management Interface Data Register and the MII Management Interface Command/Status Register. These registers can be accessed by an external CPU in the same fashion as any of the 80C300's other registers. A state machine within the 80C300 uses writes and reads to the registers to initiate writes and reads over the MDC and MDIO lines following the management interface protocol as specified in the MII specification. It also performs serial to parallel or parallel to serial conversion of I/O data between the MDIO pin and the 16 bit MII Management Interface Data Register.

To perform a PHY register write using the 80C300, the register data is first written to the MII Management Interface Data Register. Then to initiate the write operation, bit 13 of the MII Management Interface Command/Status Register is written HIGH. At the same time bits 0 through 4, bits 8 through 12, bits 5 through 7, and bit 14 of this same register should be written with the PHY physical address, PHY register address, the RXRD_TXWR clock multiplier, and the I/O complete interrupt enable bit respectively. To determine when a write operation is complete on the MII side of the 80C300, bit 15 of the MII Command/Status Register can be polled until it is read HIGH.

To initiate a PHY register read using the 80C300, bit 13 of the MII Management Interface Command/Status Register is written LOW. At the same time the remaining bits of the MII Management Interface Command/Status Register should be written similar to the description above for initiating a PHY register write operation. To determine when a read has been completed on the MII side of the 80C300 and when valid data is available in the MII Management Interface Data Register, bit 15 of the MII Command/Status Register can be polled until it is read HIGH.

Setting bit 14 of the MII Management Interface Command/Status Register enables an interrupt to be produced upon completion of either a read or a write operation.

Clearing of an interrupt due to completion of a management interface I/O operation requires reading or writing bit 15 - 8 of the Command/Status Register.

The MDC becomes active only when there is an management interface I/O operation.

3.8 COUNTERS (Note: Please check the latest Revision Comparison MD400168 for details regarding feature updates on counters)

CRC Error Counter

This is a 16 bit read only counter that counts the number of frames received or discarded with CRC errors but no framing errors. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Runt Frame Counter

This is a 16 bit read only counter that counts the number of frames received or discarded where CSN or RX_DV was active for greater than the "ShortEventMaxTime" (74-82 bit times), but less than the minimum valid frame time (64 bytes). Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Short Receive Event Counter

This is a 16 bit read only counter that counts the number of frames received or discarded with CSN or RX_DV activity less than the "ShortEventMaxTime" (74-82 bit times). Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Alignment Error Counter

This is a 16 bit read only counter that counts the number of frames received or discarded with a framing error and a CRC error both. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Transmit Collision Counter

This is a 16 bit read only counter. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Receive Collision Counter

This is a 16 bit read only counter that counts the number of collisions other than transmit collisions that occur. Collisions due to the SQET test are not counted. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Very Long Event Counter

This is a 8-bit counter that counts the number of times the transmitter is active for greater than the MAU Jabber Lockup Protection Timer allows ([4-7ms] → 10 MBit & [0.4 – 0.75ms] → 100 Mbit). Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise, completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the Receive Oversize Frame counter. By using the BE[1:0] byte enables in 16-bit mode, this counter can be read separately from the Receive Oversize Frame counter if so desired.

Receive Oversize Frame Counter

This is a 8-bit counter that counts the number of receive frames with greater than the 1518 byte maximum frame size of data. Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise, completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the Very Long Event counter. By using the BE[1:0] byte enables in 16-bit mode, this counter can be read separately from the Very Long Event counter if so desired.

Transmit Excessive Defer Counter

This is a 8-bit counter that counts the number of times the transmitter had to defer for greater than the MaxDefer-Time, (3036 byte times). Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise, completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the Transmit Late Collision counter. By using the BE[1:0] byte enables in 16-bit mode, this counter can be read separately from the Transmit Late Collision counter if so desired.

Transmit Late Collision Counter

This is a 8-bit counter that counts the number of collisions that occur greater than 512 bit times after a transmission has started, i.e. TXEN goes HIGH. Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise, completing the read will

reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the Transmit Excessive Defer counter. By using the BE[1:0] byte enables in 16-bit mode, this counter can be read separately from the Transmit Excessive Defer counter if so desired.

RXERR Error Counter

This is a 8-bit counter that counts the number of times RXERR is asserted by the Ethernet PHY, (the physical signaling portion of the Ethernet). Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read . Otherwise, completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a bye read is not used, reading this counter will also end up reading the Transmit Excessive Collision Counter. RXERR is only used by 100 MBit/sec Ethernet and is an indication of a symbol coding error for receive data.

MII Management Interface Data Register

This 16-bit register is used to read or write data from or to registers within the Ethernet PHY, (the chip used to implement the physical signaling portion of the Ethernet). For PHY register writes, the 80C300 takes data from this register and serializes it for transmission over MDIO data line. For PHY register reads, the 80C300 takes serial data off the MDIO, converts it parallel, and then writes it to this register.

Transmit Excessive Collision Counter

This is a 8-Bit counter that counts the number of times a packet collided 16 times without successful transmission. Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise , completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the RXERR Error Counter. By using the BE[1:0] byte enables in 16-Bit mode, this counter can be read separately from the RXERR counter if so desired.

Total Rx Bytes Counter

This 32-Bit counter keeps a running total of the number of bytes received for good frames. Only those receive frames meeting the IEEE 802.3 requirements for a good frame have their bytes counted. Upon reaching its maximum count value of FFFFFFFF hex, this counter rolls over to 00000000 hex. To read this counter in 8-Bit mode, the most significant byte of the counter must be read first and the least significant byte last. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally once, the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the least significant byte of the counter causes it to be loaded with the most recent packets byte count. In 16-Bit mode the counter is reset upon reading the least significant word.

Rx Overflow Error Counter

This is an 8-bit counter that counts the number of times the receive FIFO overflowed for packets being received. Upon reaching its maximum count value of FF hex, this counter will stop counting. During reading of this counter the count value will be frozen to prevent incrementing while being read. Should the 80C300 attempt to increment the counter while it is frozen, then the counter will be loaded with 01 hex upon completion of the read. Otherwise, completing the read will reset the counter to 00 hex. If the 80C300 is in 16-bit mode and if a byte read is not used, reading this counter will also end up reading the Transmit Attempt Register.

Transmit Attempt Register

This register is written with the value of the transmit attempt count at the completion of each transmit attempt. The transmit attempt count tracks how many retransmission attempts are made due to transmit collisions occurring. The best way to use this register is to program the 80C300 to produce an interrupt whenever a transmission is successfully completed. Reading this register upon the completion of a successful transmission will tell the user how many transmission attempts were necessary for the successful transmission of the packet. If the 80C300 is in 16-bit mode and a byte read is not used, reading this register will also end up reading the Rx Overflow Error Counter.

Transmit Deferred Counter

This 16-bit counter counts the number of times the transmitter had transmit data available and was ready to transmit but had to defer transmission due to carrier sense going HIGH. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the low byte is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte has been read the counter is reset to zero. Should the 80C300 attempt to increment the counter while it is frozen, then reading the low byte of the counter causes it to be loaded with 0001 hex thereby preventing the counter from missing a count. In 16-bit mode the counter is frozen during the read and reset or preset to 0001 hex upon completion of the read.

Total Rx Multicast Frames

This 16-bit counter counts the number of frames received with multicast addresses. Only those receive frames meeting the IEEE 802.3 requirements for a good frame are counted. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

Total RX Broadcast Frames

This 16-bit counter counts the number of frames received with broadcast addresses. Only those receive frames meeting the IEEE 802.3 requirements for a good frame are counted. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

Total Rx Unicast Frames

This 16-bit counter counts the number of frames received with Unicast addresses. Only those receive frames meeting the IEEE 802.3 requirements for a good frame are counted. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

Total TX Bytes Counter

This 32-bit counter keeps a running total of the number of bytes transmitted successfully, i.e. the TXRET pin does not assert. Upon reaching its maximum count value of FFFFFFFF hex, this counter rolls over to 00000000 hex. To read this counter in 8-bit mode, the most significant byte of the counter must be read first and the least significant byte last. Upon reading the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally, once the low byte is read, the counter is reset to zero. Should the 80C300 attempt to update the counter while it is frozen, then reading the least significant byte of the counter causes it to be loaded with the most recent packets byte count. In 16-bit mode the counter is reset or preloaded upon reading the least significant word.

Total TX Multicast Frames

This 16-bit counter counts the number of multicast frames transmitted successfully, i.e. the TXRET pin does not assert. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

Total Tx Broadcast Frames

This 16-bit counter counts the number of broadcast frames transmitted successfully, i.e. the TXRET pin does not assert. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

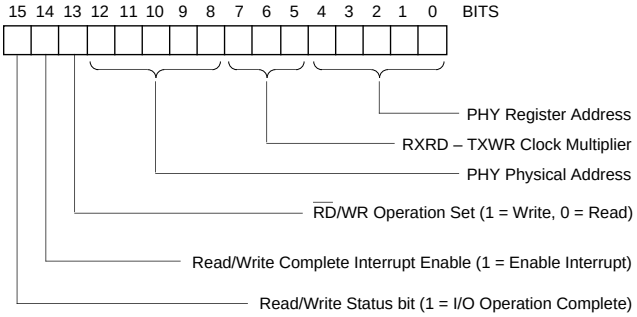
Total Tx Unicast Frames

This 16-bit counter counts the number of Unicast frames transmitted successfully, i.e. the TXRET pin does not assert. Upon reaching its maximum count value of FFFF hex, this counter will stop counting. To read this counter in 8-bit mode, the high byte of the counter must be read first followed by reading the low byte. Upon reading of the high byte, the count value of the counter is frozen to prevent the low byte count value from rolling over before it is read. Normally the counter will be reset upon reading the low byte of the counter. Should the 80C300 attempt to update the counter while it is frozen, then reading the low byte of

the counter causes the counter to be loaded with a value of 0001 hex. In 16-bit mode the counter is either reset or preloaded to 0001 hex upon being read once.

MII Management Interface Command/Status Register

This 16-bit register is used to initiate register reads or writes to a PHY connected to the 80C300. It also has a read only status indicator bit that indicates when a register operation has been completed and an interrupt enable bit to allow interrupts to be produced when the register operation is completed. The bit definitions for this register are given below:



The MDC cycle time may be set by programming the clock multiplier bits of the register. The MDC cycle time can be determined from the following formula assuming that MULTIPLIER represents the decimal equivalent of the binary value programmed for bits 7 thru 5.

MDC (cycle time) = $\overline{RXRD\_TXWR}$ (cycle time) x 2 x (MULTIPLIER + 1)

The value of bits 7 thru 5 after reset is 111 binary or 7 decimal.

ABSOLUTE MAXIMUM RATINGS

Absolute maximum ratings are limits beyond which may cause permanent damage to the device or affect device reliability. All voltages are specified with respect to GND, unless otherwise specified.

VCC Supply Voltage .....-.3V to 6.0V

All Inputs and Outputs with Respect to GND .....-.3V to VCC+.3V

Package Power Dissipation ..... 1.5 Watt @ 70 °C
Storage Temperature .....-65 to +150°C
Temperature Under Bias..... -10 to +80°C
Lead Temperature (Soldering, 10 Sec) ..... 260°C
Body Temperature (Soldering, 30 Sec) .....220°C

4.0 DC Characteristics T_A = 0° C to 70°C, V_{CC} = 5 V ± 5%

Symbol	Parameter	Limits[1] Min.	Typ.	Max.	Units	Condition
I_{IN}	Input Leakage Current			10	µA	V_{IN} = 0.45 V to 5.25 V
I_O	Output Leakage Current			10	µA	V_{OUT} = 0.45 V to 5.25 V
I_{CC}	V_{CC} Current		90	130	mA	
V_{CH}	Clock Input High Voltage	3.5			V	
V_{CL}	Clock Input Low Voltage			0.8	V	
V_{IL}	Input Low Voltage			0.8	V	
$V_{IH\ 1}$	Input High Voltage	2.0			V	
V_{OL}	Output Low Voltage RXTXDATA [31:0], RXTXEOF, SPDTAVL, TXRDY, RXRDY, TXRET, RXDC			0.4	V	I_{OL} = 8 mA
V_{OH}	Output High Voltage RXTXDATA [31:0], RXTXEOF, SPDTAVL, TXRDY, RXRDY, TXRET, RXDC	2.4			V	I_{OH} = 8 mA
V_{OL}	Output Low Voltage TXD [0:3], TXEN			0.4	V	I_{OL} = 4 mA
V_{OH}	Output High Voltage TXD [0:3], TXEN	2.4			V	I_{OH} = 4 mA
V_{OL}	Output Low Voltage All Other Outputs			0.4	V	I_{OL} = 2 mA
V_{OH}	Output High Voltage All Other Outputs	2.4			V	I_{OH} = 2 mA

NOTE:
1. Typical values are for T_A = 25°C and nominal supply voltages.

AC Test Conditions

Output Load: 1 Schottky TTL Gate + CL = 100 pF except where specifically given otherwise in the condition column.
Input Pulse Level:0.4 V to 2.4 V
Timing Reference Level:1.5 V

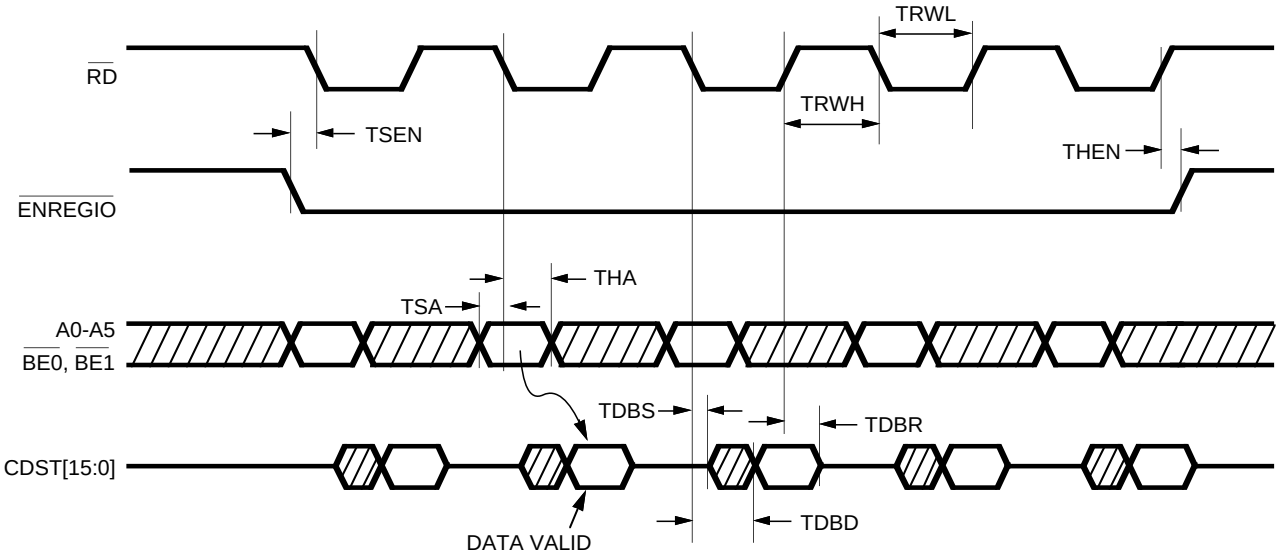
Capacitance $T_A = 25°C, F_C = 1$ MHz

Symbol	Parameter	Maximum	Condition
C_{IN}	Input Capacitance	15 pF	$V_{IN} = 0$ V
$C_{I/O}$	I/O Capacitance	15 pF	$V_{I/O} = 0$ V

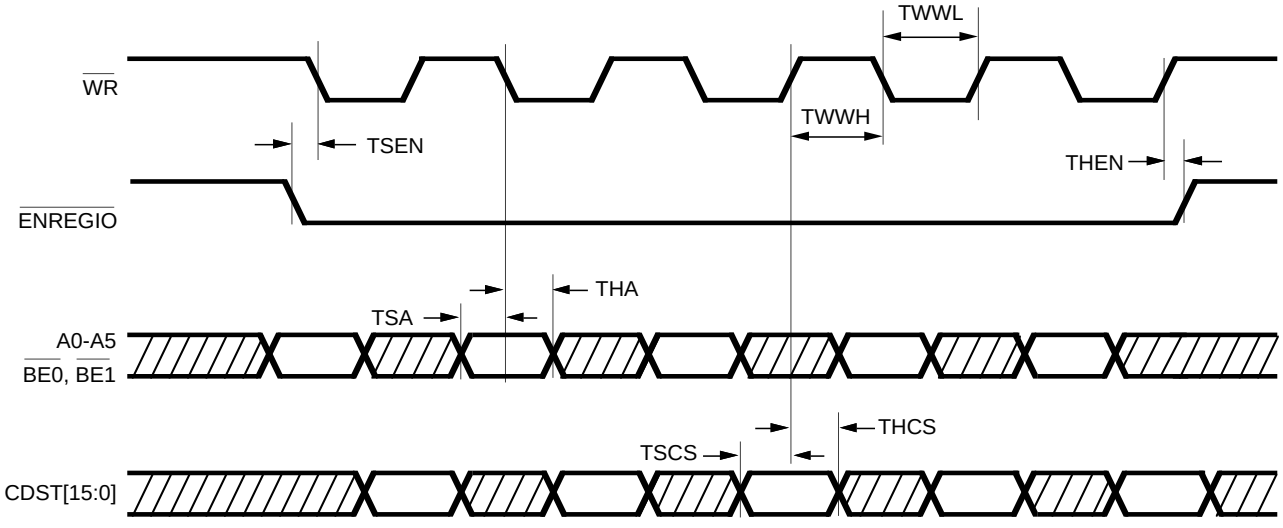
5.0 AC Characteristics $T_A = 0°$ C to 70°C, $V_{CC} = 5$ V ± 5%

Symbol[1]	Parameter	Limits			Units	Condition
		Min.	Typ.	Max.	(ns)	
COMMAND/STATUS INTERFACE READ AND WRITE TIMING						
TDBD	Receive/Transmit Command Status, and Management Counters Delay	0.5 RXC/TXC Cycles + 10 ns		1.5 RXC/TXC Cycles + 50 ns	ns	
	All Other Registers Delay	10		50	ns	
TDBR	CDST [15:0] Bus Release Delay	1.5		5.5	ns	
TDBS	CDST [15:0] Bus Siezure Delay	6			ns	
THA	A[5:0] / $\overline{BE}$ [1:0] Hold	10			ns	
TSA	A[5:0] / $\overline{BE}$ [1:0] Setup	0			ns	
TSCS	CDST Bus Setup	10			ns	
THCS	CDST Bus Hold	2			ns	
TRWH	$\overline{RD}$ High Width	1 TXC/RXC Cycle			ns	
TRWL	$\overline{RD}$ Low Width	1.5 TXC/RXC Cycles + 70 ns			ns	
TWWH	$\overline{WR}$ High Width	30			ns	
TWWL	$\overline{WR}$ Low Width	30			ns	
TSEN	$\overline{ENREGIO}$ Setup	0			ns	
THEN	$\overline{ENREGIO}$ Hold	0			ns	

5.01 Command/Status Interface Read Timing



5.02 Command/Status Interface Write Timing

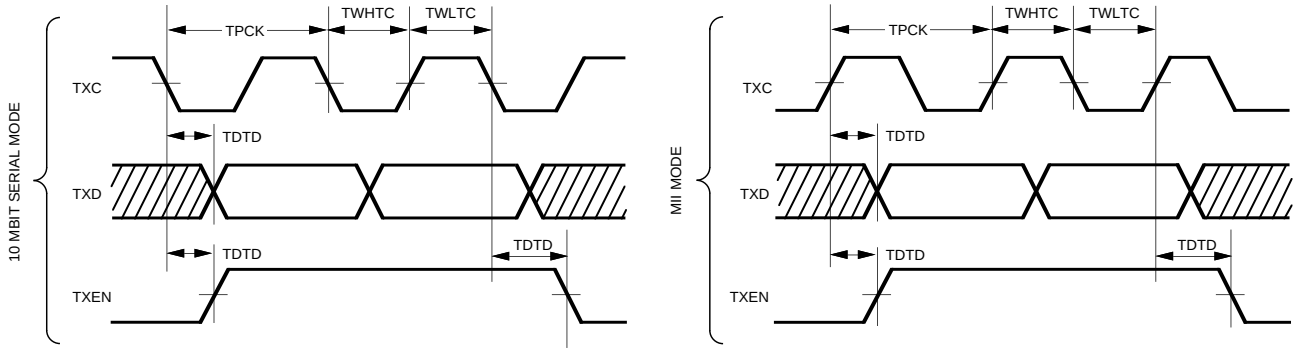


6.0 Ethernet Transmit and Receive Interface Timing

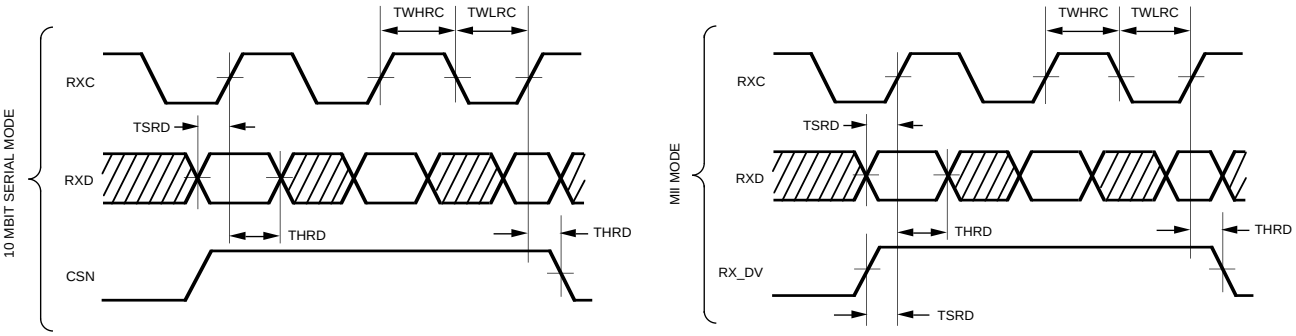
AC Characteristics T_A = 0° C to 70°C, V_{CC} = 5 V ± 5%

ETHERNET TRANSMIT INTERFACE TIMING					
		Limits			
Symbol	Parameter	Min.	Typ.	Max.	Condition
TDTD	TXD/TXEN Delay	5 ns		20 ns	
TWHTC	TXC High Width	1 TXC Cycle/2 – 5 ns			
TWLTC	TXC Low Width	1 TXC Cycle/2 – 5 ns			
ETHERNET RECEIVE INTERFACE TIMING					
THRD	RXD/CSN Hold	0 ns			
TSRD	RXD Setup	10 ns			
TWHRC	RXC High Width	1 RXC Cycle/2 – 5 ns			
TWLRC	RXC Low Width	1 RXC Cycle/2 – 5 ns			

6.01 Ethernet Transmit Interface Timing



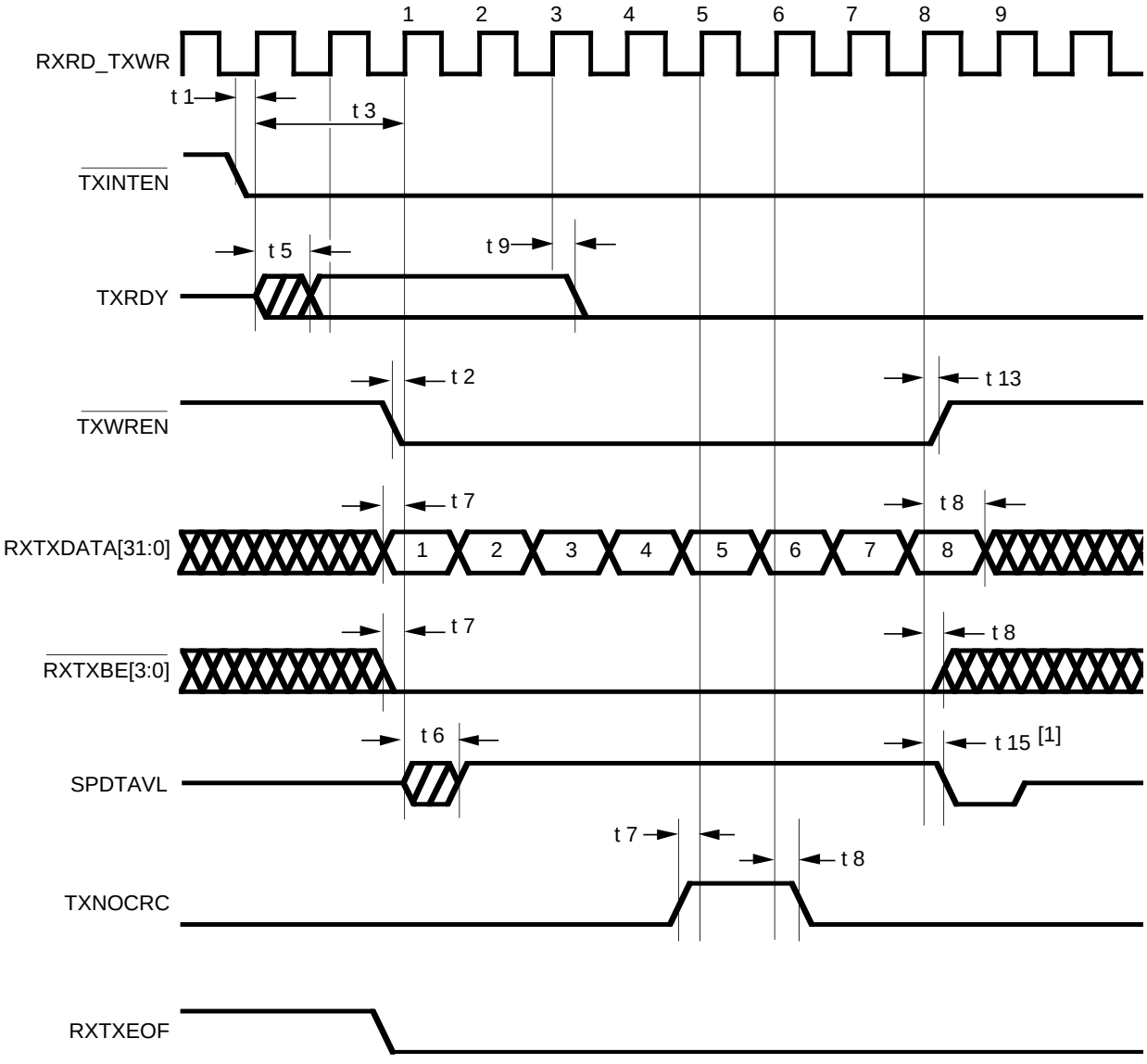
6.02 Ethernet Receive Interface Timing



7.0 Transmit Data Interface Write Timing

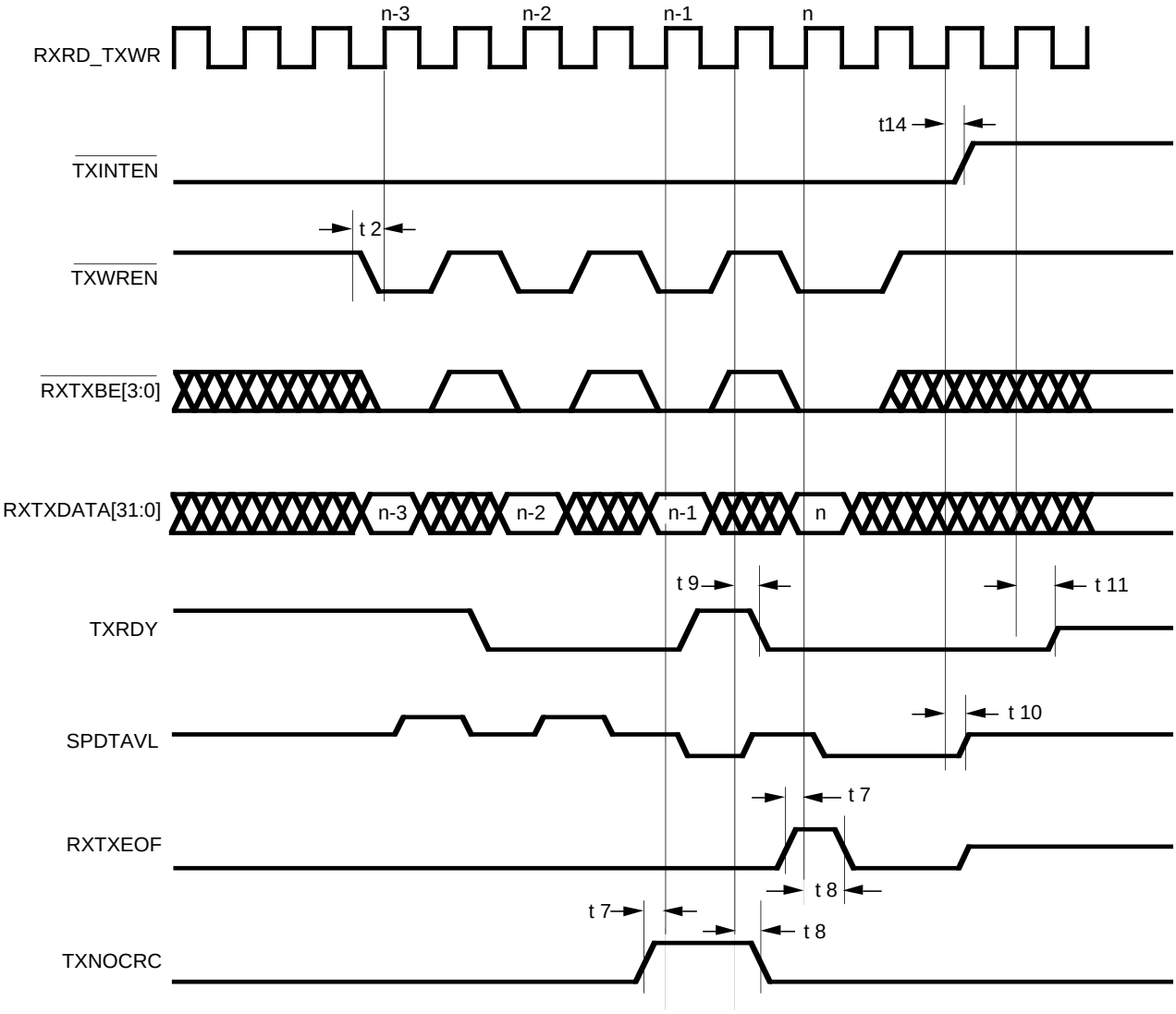
Symbol	Parameter	Min.	Typ.	Max.
t_1	Transmit Interface Enable to Clock Setup Time	5ns		
t_2	Transmit Write Enable to Clock Setup Time	5 ns		
t_3	Transmit Interface Enable to Transmit Write Enable Timing Skew	0 cycles		
t_5	TXRDY Output Enabled to Output Valid Delay	5 ns		25 ns
t_6	SPDTAVL Output Enable to Output Valid Delay	4 ns		24 ns
t_7	Transmit Data, Byte Enables, TXEOF, TXNOCRC to Clock Setup Time	5 ns		
t_8	Transmit Data, Byte Enables, TXEOF, TXNOCRC Hold Time	2.0 ns		
t_9	TXRDY Deassert Due to Threshold Being Met	5 ns		25 ns
t_{10}	SPDTAVL Output Disabled to Hi-Z Delay	3 ns		14 ns
t_{11}	TXRDY Output Disabled to Hi-Z Delay	3 ns		13 ns
t_{13}	Transmit Write Enable Hold Time	1.5 ns		
t_{14}	Transmit Interface Enable Hold Time	2.5 ns		
t_{15}	SPDTAVL Deassert Due to Transmit FIFO Reading an almost Empty Condition	4 ns		24 ns

7.01 Transmit Data Interface Write Timing 1



Notes: 1. SPDTAVL gets deasserted because of the 7th double word write to the transmit FIFO indicating that the 8th double word write will fill the FIFO completely. It is important to note that the data gets pipelined internally, hence the 7th external double word write (The 7th Clock Edge that latches in the active low TXWREN) actually happens on the 8th clock cycle internally.

7.02 Transmit Data Interface Write Timing 2



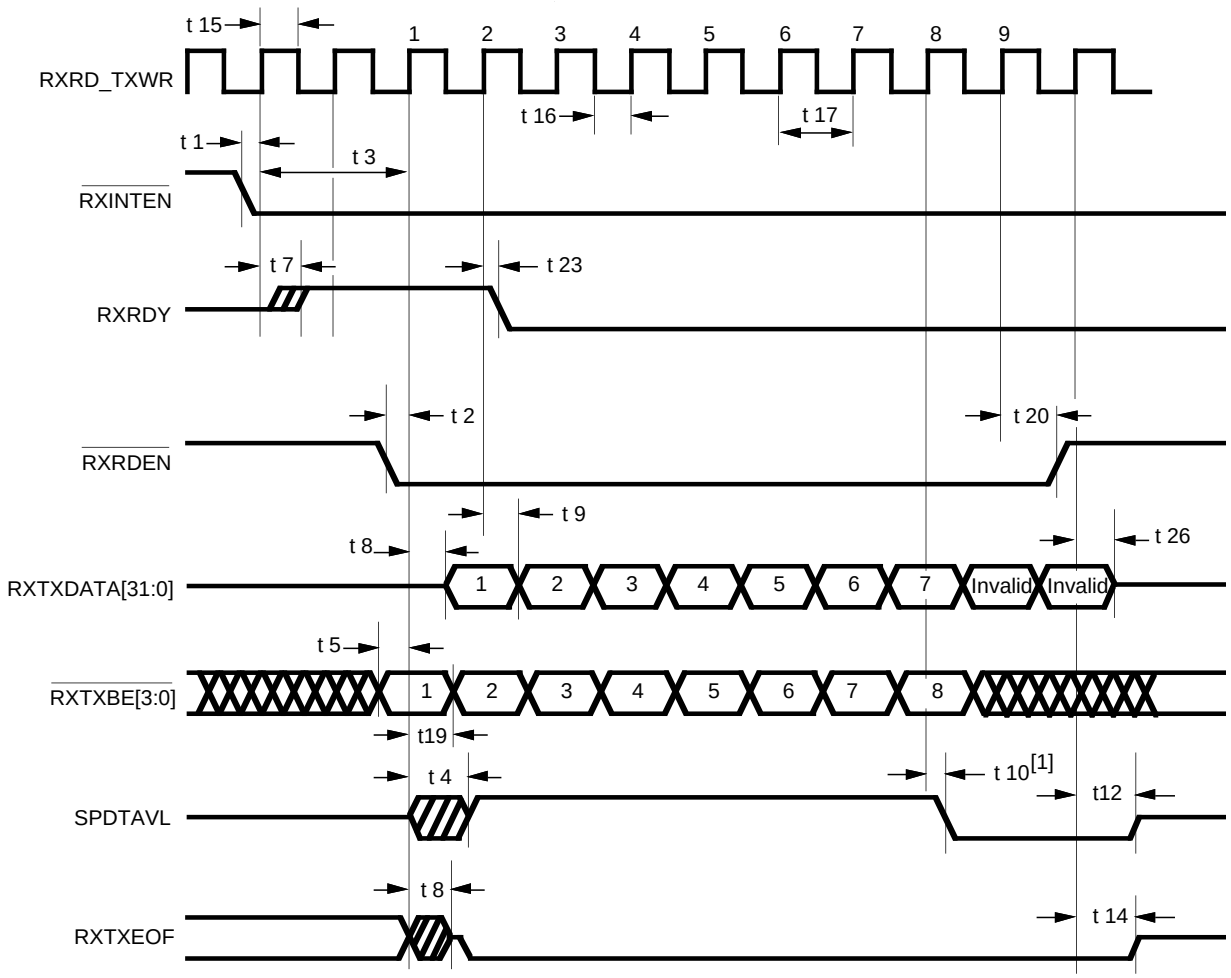
8.0 Receive Data Interface Timing

Symbol	Parameter	Min.	Typ.	Max.	Condition
t_1	Receive Interface Enable to Clock Setup Time	5ns			
t_2	Receive Read Enable to Clock Setup Time	5 ns			
t_3	Receive Interface Enable to Receive Read Enable Timing Skew	0 ns			
t_4	SPDTAVL Output Enabled to Output Valid Delay	4 ns		24 ns	
t_5	Receive Byte Enables to Clock Setup Time	3 ns			
t_7	RXRDY Output Enabled to Output Valid Delay	4 ns		26 ns	
t_8	RXTXDATA [31:0], RXTXEOF Outputs Enabled to Outputs Valid Delay	5 ns		22 ns	
t_9	FIFO Read Strobe High to RXTXEOF, RXTXDATA[31:0] FIFO Data Out Delay	5 ns		22 ns	
t_{10}	Clock to SPDTAVL Low Delay			22 ns	
	SPDTAVL Deassert to Assert Minimum Low Time	8 RXRD_TXWR Cycles			
t_{12}	SPDTAVL Output Disabled to Hi-Z Delay	3 ns		14 ns	
t_{13}	RXRDY Output Disabled to Hi-Z Delay	3 ns		12 ns	
t_{14}	Receive Data and RXTXEOF Outputs Disabled to Hi-Z Delay	3 ns		13 ns	
t_{15}	RXRD_TXWR Clock Pulse Width High	12 ns			
t_{16}	RXRD_TXWR Clock Pulse Width Low	12 ns			
t_{17}	RXRD_TXWR Clock Period	30 ns		180 ns	TXC/RXC = 10 MHz
		30 ns		50 ns	TXC/RXC = 25 MHz
		30 ns		500 ns	TXC /RXC = 2.5 MHz
t_{19}	Byte Enables Hold Time	2.5 ns			
t_{20}	Receive Read Enable Hold Time	1.5 ns			
t_{21}	Receive Interface Enable Hold Time	1.5 ns			

Receive Data Interface Timing (cont'd)

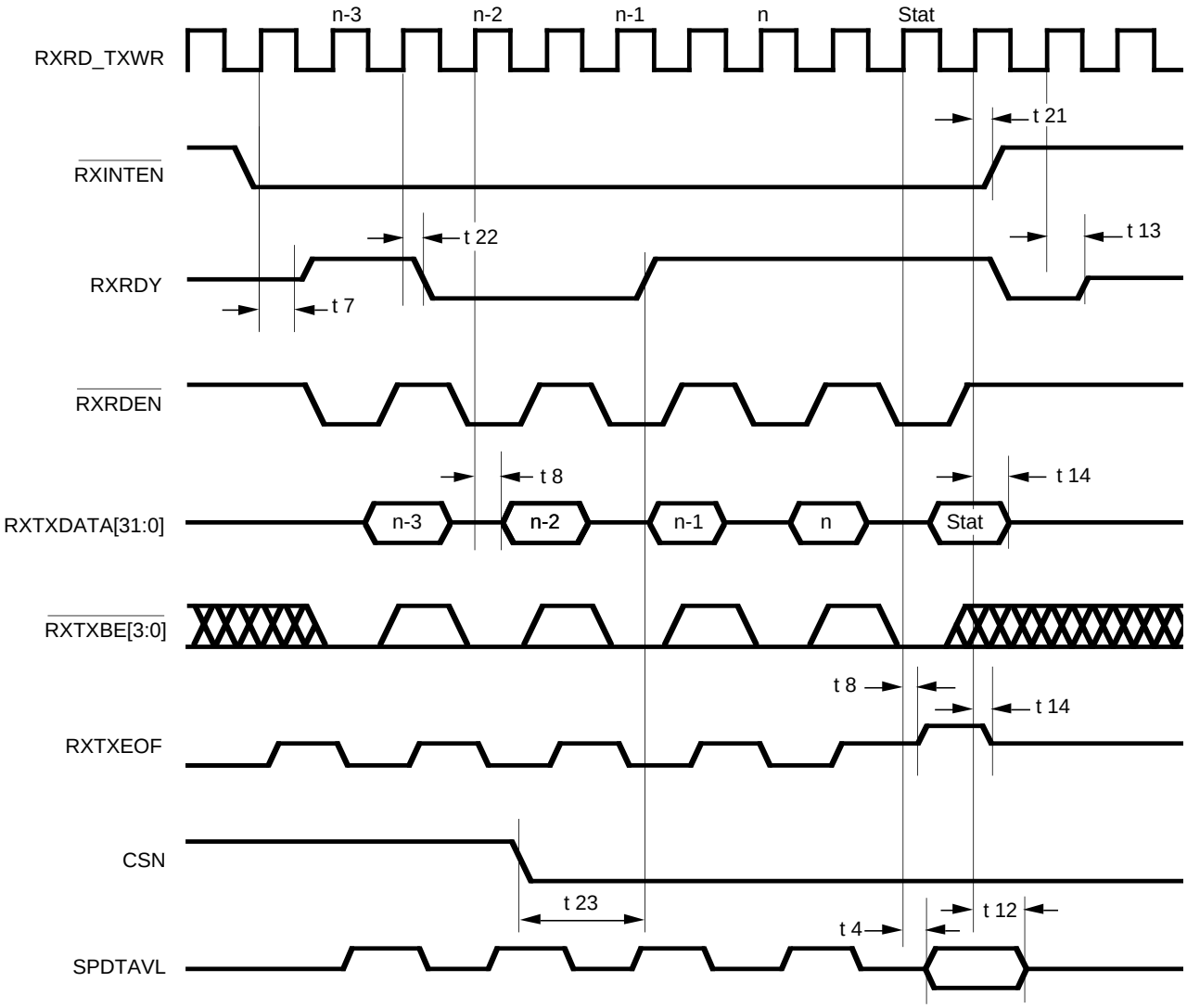
Symbol	Parameter	Min.	Typ.	Max.	Condition
t_{22}	RXRDY Deassert Due to Emptying RX FIFO Below Threshold	4 ns		26 ns	
t_{23}	RXRDY Assert from CSN Going Low Due to Status Write	9 RXC Cycles + 2.5 RXRD_TXWR Cycles + 4 ns (10MBit/sec Serial Mode)		17 RXC Cycles + 3.5 RXRD_TXWR Cycles + 22 ns (10MBit/sec Serial Mode)	
		3 RXC Cycles + 2.5 RXRD_TXWR Cycles + 4 ns (MII Mode)		5 RXC Cycles +3.5 RXRD_TXWR Cycles + 22 ns (MII Mode)	

8.01 Receive Data Interface Read Timing 1



Notes: 1. SPDTAVL gets deasserted because of the 7th double word read from the receive FIFO indicating that there is no more data available in the receive FIFO and further reads will cause invalid reads. Here, it is important to note that the 7th read is referred to the 7th clock edge that latches in the active low RXRDEN and the resultant data can be latched out on the 8th clock edge because of the pipelining effect.

8.02 Receive Data Interface Read Timing 2



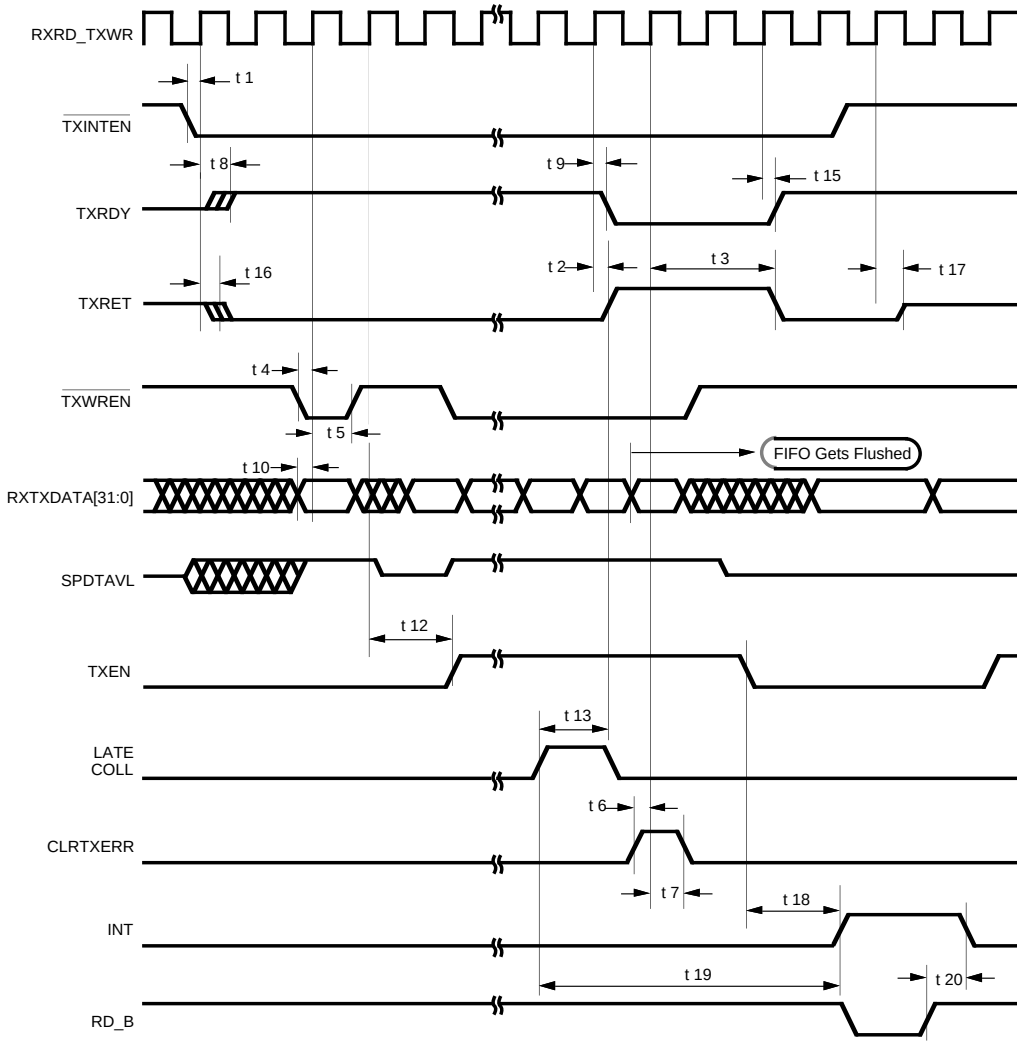
9.0 Transmit Data Interface Timing on Exception Conditions

Symbol	Parameter	Min.	Typ.	Max.
t_1	TXINTEN Setup Time	5ns		
t_2	RXRD_TXWR to TXRET Delay	9.5 ns		38 ns
t_3	TXRET Deassert from CLRTXERR	1 TXC Cycle + 1 RXRD_TXWR Cycle + 7 ns		2 TXC Cycles + 2 RXRD_TXWR Cycles + 28 ns
t_4	TXWREN Setup Time	5 ns		
t_5	TXWREN Hold Time	0 ns		
t_6	CLRTXERR Setup Time	12 ns		
t_7	CLRTXERR Hold Time	0 ns		
t_8	TXRDY Output Enabled to Output Valid Delay	5 ns		25 ns
t_9	TXRDY Deassert Due to TXRET Going HIGH Because of an Exception Condition	5 ns		1 RXRD_TXWR Cycle + 25 ns
t_{10}	RXTXDATA Setup Time	5 ns		
t_{12}	TXEN Assert from First Data Write to the Transmit FIFO (Assuming Defer Time Has Been Met)	0.75 RXRD_TXWR Cycles + 18.5 TXC Cycles + 5 ns (10 Mbit/sec Serial Mode)		0.75 RXRD_TXWR Cycles + 26.5 TXC Cycles + 20 ns (10 Mbit/sec Serial Mode)
		0.75 RXRD_TXWR Cycles + 6.5 TXC Cycles + 5 ns (MII Mode)		0.75 RXRD_TXWR Cycles + 8.5 TXC Cycles + 20 ns (MII Mode)
t_{13}	TXRET Set Delay Due to Late Collision or 16 Collisions	25 TXC Cycles + 1 RXRD_TXWR Cycle + 9.5 ns (10 Mbit/sec Serial Mode)		34 TXC Cycles +2 RXRD_TXWR Cycles + 38 ns (10Mbit/sec Serial Mode)
		7 TXC Cycles + 1 RXRD_TXWR Cycle + 9.5 ns (MII Mode)		10 TXC Cycles + 2 RXRD_TXWR Cycles + 38 ns (MII Mode)
	TXRET Set Due to Underflow	8 TXC Cycles + 1 RXRD_TXWR Cycle + 9.5 ns (10 Mbit/sec Serial Mode)		8 TXC Cycles + 2 RXRD_TXWR Cycles + 38 ns (10 Mbit/sec Serial Mode)
		2 TXC Cycles + 1 RXRD_TXWR Cycle + 9.5 ns (MII Mode)		2 TXC Cycles + 2 RXRD_TXWR Cycles + 38 ns (MII Mode)
t_{15}	TXRDY Going HIGH Due to TXRET Going Low	5 ns		25 ns
t_{16}	TXRET Output Enabled to Output Valid Delay	9.5 ns		38 ns
t_{17}	TXRET Output Disabled to Hi-Z Delay	3 ns		12 ns

9.0 Transmit Data Interface Timing on Exception Conditions (cont'd)

Symbol	Parameter	Min.	Typ.	Max.
t_{18}	INT High to TXEN Low Delay Due to Underflow	1 TXC Cycle + 15 ns		1 TXC Cycle + 40 ns
	TXEN Low to INT HIGH Delay Due to Carrier Sense Dropout	2 TXC Cycles + 15 ns		2 TXC Cycles + 40 ns
	TXEN Low to INT High Delay Due to Successful Transmission	10 ns		20 ns
t_{19}	COLL High to INT High Delay	20 TXC Cycles + 15 ns (10 Mbit/sec Serial Mode)		27 TXC Cycles + 40 ns (10 MBit/sec Serial Mode)
		8 TXC Cycles + 15 ns (MII Mode)		9 TXC Cycles + 40 ns (MII Mode)
t_{20}	INT Clear Delay	1.5 TXC Cycles + 15 ns		2.5 TXC Cylces + 40 ns

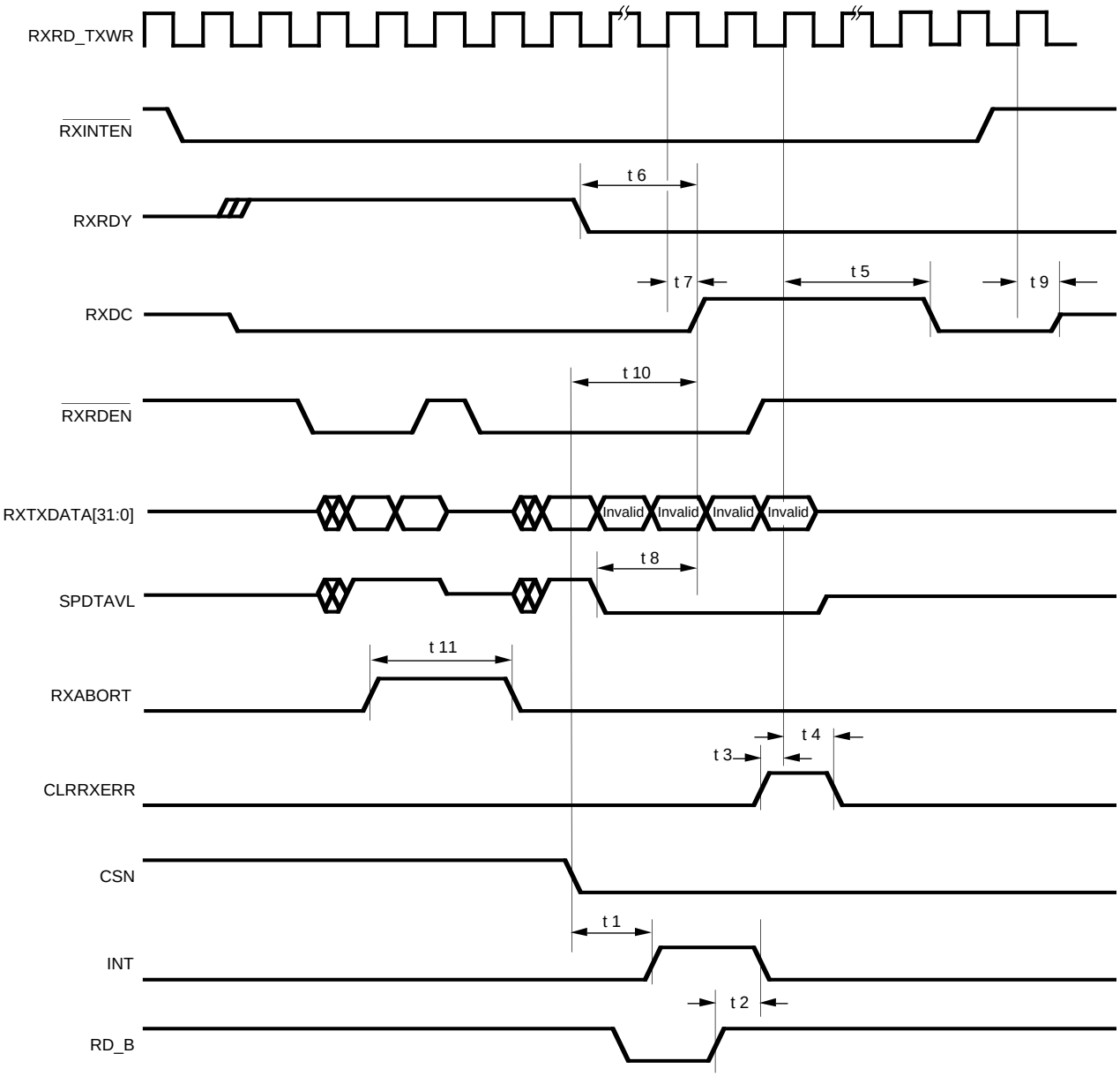
9.0 Transmit Data Interface Timing on Exception Conditions



10.0 Receive Data Interface Timing on Exception Conditions

Symbol	Parameter	Min.	Typ.	Max.	Condition
t_1	Receive INT Delay Due to Shortframe, CRC, Good Frame, or Oversized Packet	2 RXC Cycles + 15 ns		2 RXC Cycles + 40 ns	
	Receive INT Delay Due to Overflowed Packet	2 RXC Cycles + 15 ns		18 RXC Cycles + 40 ns	
t_2	INT Clear Delay	1.5 RXC Cycles + 15 ns		2.5 RXC Cycles + 40 ns	
t_3	CLRRXERR Setup Time to RXRD_TXWR	6 ns			
t_4	CLRRXERR to RXRD_TXWR Hold Time	0 ns			
t_5	CLRRXERR High to RXDC Low Delay	1 RXC Cycle + 3 RXRD_TXWR Cycles + 6 ns		2 RXC Cycles + 4 RXRD_TXWR Cycles + 27 ns	
t_6	RXRDY Deassert Due to Discard to RXDC High Delay	5 ns		1 RXRD_TXWR Cycle + 11 ns	
t_7	RXRD_TXWR to RXDC Delay	9 ns		37 ns	
t_8	SPDTAVL Deassert Due to Discard to RXDC High Delay	5 ns		1 RXRD_TXWR Cycle + 13 ns	
t_9	RXRD_TXWR to RXDC Hi-Z	3 ns		11 ns	
t_{10}	CSN Deassert to RXDC High Due to Receive Overflow Condition	2 RXC Cycles + 3 RXRD_TXWR Cycles + 9 ns (10MBit/sec Serial Mode)		18 RXC Cycles + 4 RXRD_TXWR Cycles + 37 ns (10MBit/sec Serial Mode)	
		2 RXC Cycles + 3 RXRD_TXWR Cycles + 9 ns (MII Mode)		6 RXC Cycles + 4 RXRD_TXWR Cycles + 37 ns (MII Mode)	
t_{10a}	RXDC High From Point of Detection of Receive Packet with Greater than 1518 Bytes	2 RXC Cycles + 3 RXRD_TXWR Cycles + 9 ns		2 RXC Cycles + 4 RXRD_TXWR Cycles + 37 ns	
t_{11}	RXABORT Pulse Width	1.5 RXC			RXABORT is Asynchronously Asserted with Respect to RXC
	RXABORT to RXC Setup Time	5 ns			RXABORT is Synchronously Asserted with Respect to RXC
	RXC to RXABORT Hold Time	5 ns			

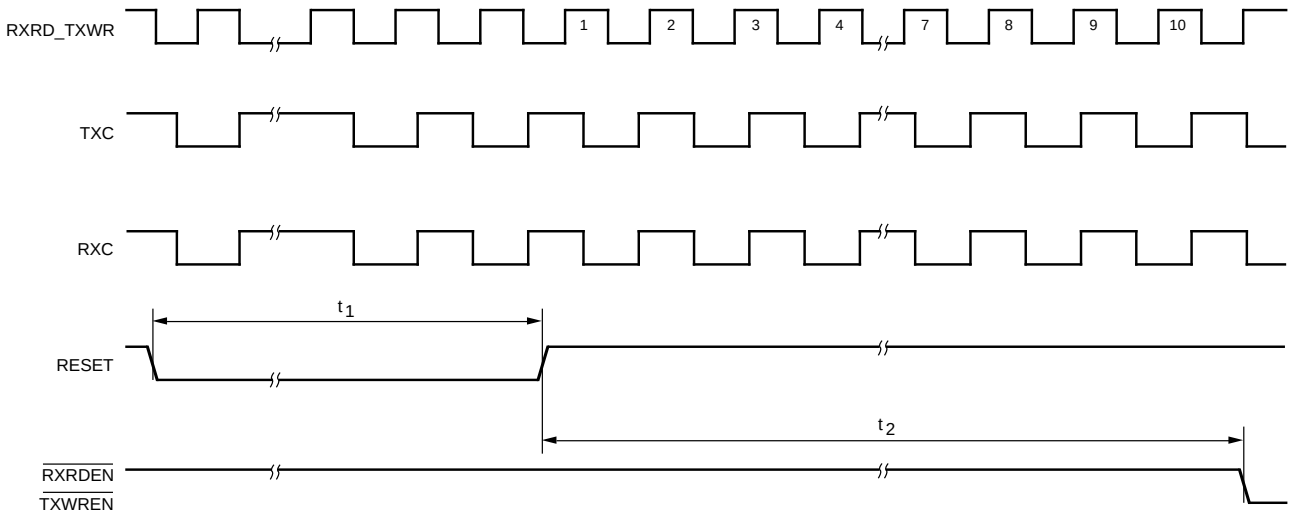
10.0 Receive Data Timing on Exception Conditions



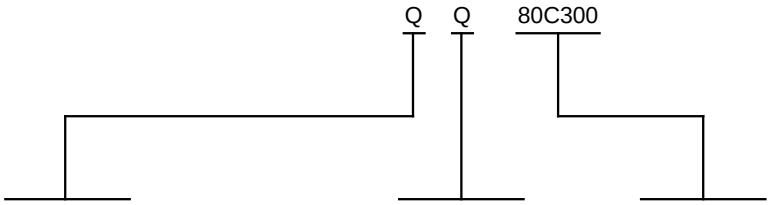
11.0 Reset Timing

Symbol	Parameter	Min.	Typ.	Max.	Condition
t_1	Asynchronous Reset Pulse Width	10 µs			All clocks must be active during this peroid of time
t_2	Reset Completion to Normal Operation Delay	10 RXDR_TXWR Cycles			

11.0 Reset Timing



Ordering Information



SEEQ Hurricane, and Full Duplex Designation

SEEQ's Hurricane family of products offer 100MBit Fast Ethernet Solutions. Symbol identifies product as a part of SEEQ's Hurricane family.

Full Duplex

Symbol identifies product as Full Duplex device.

Revision History

4/19/96

Page 18: Using the 80C300 in 8 Bit or 16 Bit Mode sub section has been added to Section 3.5.2.

Page 18, Section 3.5.3: The end of the second paragraph in this section has been replaced with the new sub section Using the 80C300 in 8 Bit or 16 Bit Mode.

Page 24: Receive Own Transmit Disable Mode has been deleted and replaced with new sub section Disable Loopback Mode and new table Configuration Register #1.

Page 25: Section 3.6.9 has been entirely replaced with new copy.

11/7/96

11/7/96 - Document Revision changed to MD400145/D

Page 5, Pin Description:
- Pin 45 Description now reads; This is the system clock acting as the chip's ...

Page 6, Pin Description:
- Pin 50 Description now reads; This is an active high output that can be used for validating reads from the receive FIFO during a read operation and preventing over writes to the transmit FIFO during a write operation. For further details, please refer to the Transmit Data Write Timing and the Receive Data Read Timing diagrams.

Revision History

11/7/96

Page 10 - Section 2.0 Introduction has been deleted and replaced with new Section 2.0 Introduction.

Page 16 - Section 3.3.5 Second paragraph, now reads; Except for discards due to address mismatches and oversized packets, all packet ...

Page 20 - Illustration, Transmit Command Register; Bit 4 has been changed from 100 Mbit Mode, to MII Mode.

Page 22 - Format of the Status Double Word, illustration has been added.

Page 25 - Configuration Register #1 Illustration has been changed; now reads, Bit 5 = '1' Enables Full Duplex Mode [Bit 3 should be '0'].
- Configuration Register #1 Table has changed, Loopback Mode has been added.

Page 26 - Configuration Register #3 Illustration has been changed; reference to Bit 2 and reference to Bit 3 have been added.

Pages 33 to 42, has been deleted and replaced with new Tables and Timing Diagrams, also the pagination has changed.

Page 33, 4.0 DC Characteristics
- All Parameter references to [1,4] have been deleted.

Page 34, AC Characteristics:
- TDBD (min) has been changed from 100 to 0.5 RXC/TXC Cycles + 10 ns.
- TDBD (max) has been changed from 200 to 1.5 RXC/TXC Cycles + 50 ns.
- TDBD, All Other Registers (min) is now 10.
- TDBR (min) has been changed from 7 to 1.5.
- TDBR (max) has been changed from 20 to 5.5.
- TDBS (min) has been changed from 10 to 6.
- TDBS (max) has been changed from 20 to 32.
- Symbol *THAR* has been changed to THA.
- THA Parameter has been changed from A_{0-2}/Reg PS[1:0] Hold to A[3:0] Hold.
- Symbol *TSAR* has been changed to TSA.
- TSA Parameter has been changed from A_{0-2}/Setup to A[3:0] Setup.
- THCS row is new.
- Symbol *TWCH* has been changed to *TRWH*.
- TRWH Parameter has been changed from RD/WR High Width, to RD High Width.
- TRWH (min) has been changed from 200 to 1 TXC/RXC Cycle.
- Symbol *TWCL* has been changed to *TRWL*.
- TRWL Parameter has been changed from RD/WR Low Width to RD Low Width.
- TRWL (min) has been changed from 200 to 1.5 TXC/RXC Cycles + 70 ns.
- TWWH row is new.
- TWWL row is new.
- All Parameter references to [7:0] have been changed to [15:0], and all Parameter references [3:0] have changed to [5:0].
- All Condition references to RXC,TXC = 10MHz have been deleted.

Page 35 - New Timing Diagrams, 5.01 Command/Status Interface Read Timing, and 5.02 Command/Status Interface Write Timing.

Page 36 - New Timing Diagrams, 6.01 Ethernet Transmit Interface Timing, and 6.02 Ethernet Receive Interface Timing.

Page 36 - New Table 6.0 Ethernet Transmit and Receive Interface Timing.

Page 37 - New Table, 7.0 Transmit Data Interface Timing.

Page 38 - New Timing Diagram, 7.01 Transmit Data Interface Write Timing 1.

Page 39 - New Timing Diagram, 7.02 Transmit Data Interface Write Timing 2.

Revision History

11/7/96

Page 40 - New Table, 8.0 Receive Data Interface Timing.

Page 41 - New Timing Diagram, 8.01 Receive Data Interface Read Timing 1.

Page 42 - New Timing Diagram, 8.02 Receive Data Interface Read Timing 2.

Page 43 - New Table, 9.0 Transmit Data Interface Timing on Exception Conditions.

Page 44 - New Timing Diagram, 9.0 Transmit Data Timing on Exception Conditions

Page 45 - New Table, 10.0 Receive Data Interface Timing on Exception Conditions.

Page 46 - New Timing Diagram, 10.0 Receive Data Timing on Exception Conditions.

Page 51 - Appendix A has been added.

1/23/97

Page 11 - Section 3.2.2 Transmission Initiation/Deferral has changed to 3.2.2 Transmission Initiation in Full Duplex and CSMA/CD Networks, and this section has been replaced with new copy.

Page 18 - Column two, third paragraph last sentence, Because transmit FIFO ... has been changed to, Because transmit FIFO writes are pipelined, there will always be one more FIFO write after $\overline{\text{TXWREN}}$ is deasserted externally.

7/21/97

7/22/97 - Document Revision changed to MD400145/E

Global Change: All references to $\overline{\text{RXRD\_TXWR}}$ have been changed to RXRD_TXWR.

Page 1, Features
- Additional Feature; Successful Packet Transmit Completion Feature, has been added to; The Following Additional Features Can Be Programmed for 80C300

Page 3, Figure 1. 80C300 Functional Block Diagram
- Reference to $\underline{\text{RXINTEN}}$ has been changed to $\overline{\text{RXINTEN}}$
- Reference to $\overline{\text{RXRD}}$ has been changed to RXRD_TXWR

Page 5, 1.0 Pin Description
- Pin # 62 Description has been changed to: This input is an active low asynchronous chip reset, After reset all registers except the Hash and Station Address registers are to reset zero, all FIFOs are cleared, all counters are reset to zero.

Revision History

Page 7, 1.0 Pin Description Continued
- Pin # 126, Daisy Out, has been added.

Page 11, Section 3.2.1 Controlling Transmit Packet Encapsulation
- First paragraph copy addition; ...FCS value can be controlled on a packet per packet basis by using the TXNOCRC pin as long as the TXNOCRC Tx-Rx Configuration register bit has not been set high. If the TXNOCRC pin is held high (or) if the TXNOCRC bit is set anytime during the duration of a packet write to the transmit FIFO, that particular packet will not be appended with a CRC value.

Transmit No CRC		CRC Appendage
H/W Pin 52	S/W Bit 4 of Config 1	To the Packet
0	0	Yes
0	1	No
1	0	No
1	1	No

Please note that both the H/W pin and the software bit should be kept deasserted during the entire duration of the packet write to the transmit FIFO in order to transmit a packet with CRC.

Page 13, Figure 3 Typical Application Example:
- Reference to 80220 10/100 Base-TX (or); has been added to illustration.

Page 26, Section 3.6.9 Defer Register Calculations for the 80C300:
- Paragraph has been added to Section; The defer time is split into two periods. The first period is the first 2/3 and the second period is the second 1/3 of the defer time. The defer time calculated by the following algorithms are for the first 2/3 of the defer period only. For further details, please refer to the section 3.2.2.

Page 33, Absolute Maximum Ratings:
- Absolute Maximum Ratings is new.

4.0 DC Characteristics:
- 4.0 DC Characteristics T_A = 0° C to 70°C, V_{CC} = 5 V to 5%, has been changed to, 4.0 DC Characteristics T_A = 0° C to 70°C, V_{CC} = 5 V ± 5%
- V_{CH} Limits (Max) is now blank.
- V_{IH1} Limits (Max) is now blank.

Page 35, Figure 5.02 Command/Status Interface Write Timing:
- Timing CDST [15:0] reference to TSCS, THCS has changed.

Revision History

10/8/97

10/8/97 - Document Revision changed to MD400145/F

Page 2, 11.0 Reset Timing has been added.

Page 11, Section 3.2.2 Transmission Initiation in Full Duplex and CSMA/CD Networks

- Copy change paragraph 2: The transmit threshold value..... has been changed to, The transmit threshold value is controlled by programming bits 7 through 4 of the Configuration Register #3. The default value is 0 (zero), which enables the MAC to begin packet transmission with as little as one double word in the FIFO. The threshold, measured in double words, is equal to the number programmed into Transmit Control Register times 2. Thus, if the upper four bits of the register contain the value 3hex, then the transmission is deferred until there are at least 6 double words of data in the FIFO.

Page 15, 3.6.1 Internal Channel Register Addressing Table

- Copy change to Note 4. The product I.D. Register is an 8 Bit read only register (please refer to Revision Comparison MD400168 for details). It can be read as described in the above table.

Page 24, 3.6.6.1 FIFO Threshold Register Settings Table, has been changed to 3.6.8.1 FIFO Threshold Register Settings Table

Page 26 , Configuration Register #3 Illustration bits 7, 6 ,5, 4 reserved has been changed to bits 7, 6, 5, 4 Transmit Control Bit Settings

- Section 3.6.8 FIFO Threshold Register; copy change, On page 23 is a table.... has been changed to... Table 3.6.8.1 shows the number of double words of space/data that are ....

Page 28, 3.8 Counters; Copy addition now reads, 3.8 Counters (Note: Please check the latest Revision Comparison MD400168 for details regarding feature updates on counters)

Page 31, Section Total Tx Unicast Frames

- Copy change; ....the number of broadcast frames transmitted.... has been changed to ...the number of Unicast frames transmitted....

Page 32, Section Total Tx Bytes Counter, has been deleted.

Page 34, 5.0 AC Characteristics

- Symbol TDBD, Parameter now reads... Receive/Transmit Command Status, and Management Counters Delay... and ... All Other Registers Delay
- Symbol TDBS (Max) is now blank
- Symbol THA Parameter A[5:0] Hold has been changed to A[5:0] / $\overline{BE}$ [1:0] Hold
- Symbol TSA Parameter A[5:0] Setup has been changed to A[5:0] / $\overline{BE}$ [1:0] Setup
- Symbol TSA (Min) has been changed from 15 to 0
- Symbol THCS (Min) has been changed from 0 to 2
- Symbol TRWH, $\overline{RD}$ / High Width has been changed to $\overline{RD}$ High Width
- Symbol TRWL $\overline{RD}$ / Low Width has been changed to $\overline{RD}$ Low Width
- Symbol TSEN row has been added
- Symbol THEN row has been added

Revision History

Page 35, 5.01 Command/Status Interface Read Timing
- Timing Wave A0-A5, BE0, BE1 has been modified
- Timing Wave CDST [15:0] has been modified

5.02 Command/Status Interface Write Timing
- Timing Wave A0-A5, BE0, BE1 has been modified
- Timing Wave CDST [15:0] has been modified

Page 36, 6.0 Ethernet Transmit and Receive Interface Timing
- Symbol THRD, Parameter has been changed from RXD Hold, to RXD / CSN Hold, and (Min) has been changed from 5 ns to 0 ns
- Symbol TSRD, (Min) has been changed from 5 ns to 10 ns

6.01 Ethernet Transmit Interface Timing
- Illustration 10 MBIT SERIAL MODE, Timing Wave TXD and TXEN have changed.
- Illustration MII MODE, Timing Wave TXD and TXEN have changed.

6.02 Ethernet Receive Interface Timing
- Illustration 10 MBIT SERIAL MODE, Timing Wave RXD and CSN have changed.
- Illustration MII MODE, is new.

Page 37, 7.0 Transmit Data Interface Write Timing
- Symbol t_3, (Min) has been changed from 0 ns to cycles
- Symbol t_8, (Min) has been changed from 1.5 ns to 2.0 ns
- Symbol t_{13}, (Min) has been changed from 0 ns to 1.5 ns
- Symbol t_{14}, (Min) has been changed from 0 ns to 2.5 ns

Page 40, 8.0 Receive Data Interface Timing
- Symbol t_5, (Min) has been changed from 5 ns to 3 ns
- Symbol t_9, (Max) has been changed from 24 ns to 22 ns
- Symbol t_{19}, (Min) has been changed from 0 ns to 2.5 ns
- Symbol t_{20}, (Min) has been changed from 0 ns to 1.5 ns
- Symbol t_{21}, (Min) has been changed from 0 ns to 1.5 ns

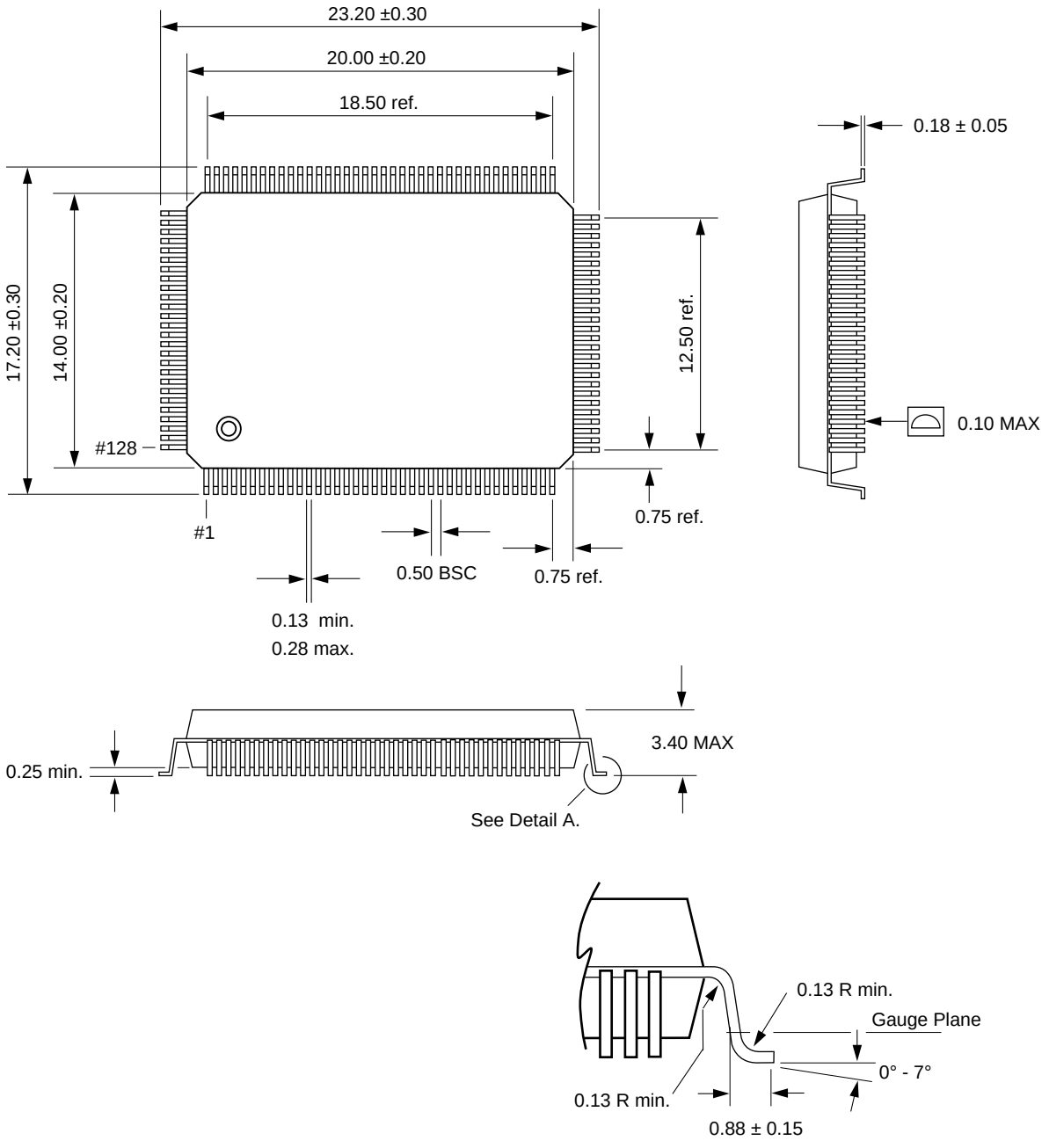
Page 47, 11.0 Reset Timing
- 11.0 Reset Timing Table is new
- 11.0 Reset Timing Illustration is new

1/12/98

1/12/98 - Document Revision changed to MD400145/G

Page 26, Configuration Register #3
- Bit 2 and Bit 3 description have been switched
- New Full Duplex Feature: reference to revisions A, B, and C has been changed to A and B. reference to revision D has been changed to C.

128 Lead PQFP



Detail A.

Notes
1. All dimensions are in millimeters.