# M82592
# ADVANCED CSMA/CD LAN CONTROLLER WITH 16-BIT DATA PATH

*Military*

- Supports Industry Standard LANs
  - Ethernet and Cheapernet (IEEE 802.3 10BASE5 and 10BASE2)
  - Broadband Ethernet (IEEE 802.3 10BROAD36)
  - StarLAN (IEEE 802.3 1BASE5)
- Integrates Physical and Data Link Layers of OSI Model
  - Complete CSMA/CD Medium Access Control (MAC) Functions
  - Manchester, Differential Manchester, or NRZI Encoding/Decoding
  - On-Chip, Logic-Based Collision Detection
  - IEEE 802.3 or HDLC Frame Delimiting
- Two Modes of Operation
  - Bit Rates Up to 4 Mb/s with On-Chip Encoder/Decoder (High-Integration Mode)
  - Bit Rates Up to 20 Mb/s with External Encoder/Decoder (High-Speed Mode)
- High-Performance System Interface
  - 16-MHz Clock, 2 Clocks per Transfer
  - 64 Bytes of Configurable FIFO
- Efficient Memory Use via Buffer and Frame Chaining
- DMA Interface for Retransmission and Continuous Reception Without CPU Intervention
  - $\overline{EOP}$ Signal Generation for M82380
- Supports 8- or 16-Bit DMA Transfers
- Local and Remote Power-Down Modes
- Deterministic Collision Resolution
- 24-Bit General Purpose Timer
- On-Chip Jabber Inhibit Function
- Network Management and Diagnostics
  - Monitor Mode
  - CRC, Alignment, and Short Frame Error Detection
  - Three 16-Bit Event Counters
  - Short or Open Circuit Localization
  - Self-Test Diagnostics
  - Internal and External Loopback Operation
  - Internal Register Dump
- High-Speed CHMOS III Technology
- Military Temperature Range: $-55°C$ to $+125°C$ ($T_C$)
- Available in 68-Pin CQFP

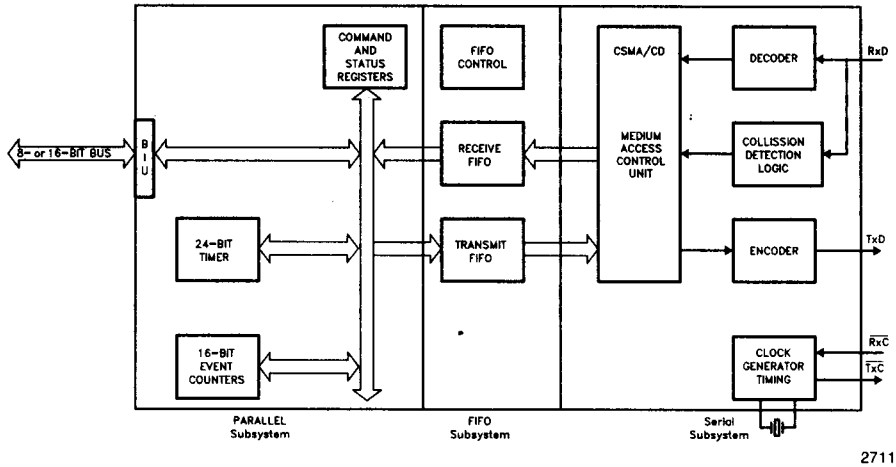



**Figure 1. M82592 Block Diagram**



January 1990
Order Number: 271120-001

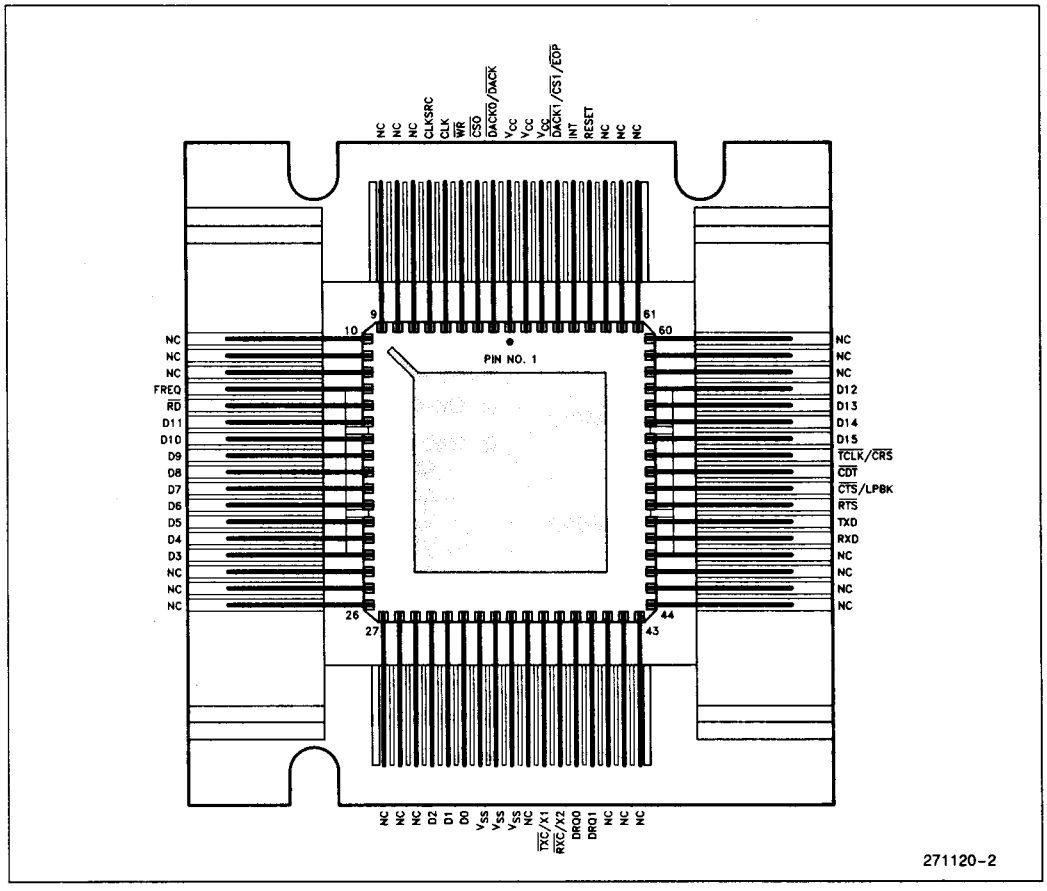



**Figure 2. M82592 68-Lead Flatpack in Carrier**

The M82592 is a second-generation, 16-bit data path CSMA/CD controller. Its system interface enables efficient operation with a wide variety of Intel microprocessors (e.g., M80186, M80286, or M80386) and industry standard buses (such as the IBM PC I/O channel or Personal System/2 Micro Channel). The M82592 can be configured to support a wide variety of industry standard networks, including StarLAN, IBM PC Network, and Ethernet/Cheapernet.

The M82592 is ideal for integrated LAN on motherboard solutions. The M82592 architecture offers low cost, high performance and minimal real estate requirements. The M82592's Tightly Coupled Interface mode allows it to use host DMA without local buffering. An integrated M82592 Ethernet motherboard LAN will occupy less than five percent of the total motherboard area. The CHMOS M82592 can be used in low power or no-fan systems such as diskless workstations and laptop PCs. The M82592 provides two power-down modes for these environments.

The M82592 is available in a 68-pin CQFP. It is fabricated with Intel's reliable CHMOS III technology.

**Table 1. Pin Description**

| Symbol | Pin No. (CQFP) | Type | Name and Function |
|---|---|---|---|
| D15<br>D14<br>D13<br>D12<br>D11<br>D10<br>D9<br>D8<br>D7<br>D6<br>D5<br>D4<br>D3<br>D2<br>D1<br>D0 | 54<br>55<br>56<br>57<br>15<br>16<br>17<br>18<br>19<br>20<br>21<br>22<br>23<br>30<br>31<br>32 | I/O | **DATA BUS:** The Data Bus lines are bidirectional, three-state lines connected to the CPU's Data Bus for transfers of data, commands, status, and parameters. |
| $\overline{RD}$ | 14 | I | **READ:** Together with $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$, Read controls data or status transfers out of the M82592. |
| $\overline{WR}$ | 4 | I | **WRITE:** Together with $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$, Write controls data or command transfers into the M82592. |
| $\overline{CS0}$ | 3 | I | **CHIP SELECT (PORT 0):** When LOW, the M82592 is selected by the CPU for command or status transfer through PORT 0. |
| RESET | 64 | I | **RESET:** A HIGH signal on this pin causes the M82592 to terminate current activity. This signal is internally synchronized and must be held HIGH for at least four Clock (CLK) cycles.<br>When the Clock signal is provided internally (CLKSRC is strapped HIGH), the RESET signal must be held HIGH for at least 50 $\mu$s. |
| INT | 65 | O | **INTERRUPT:** A HIGH signal on this pin notifies the CPU that the M82592 is requesting an interrupt. |
| DRQ0 | 39 | O | **DMA REQUEST (CHANNEL 0):** This pin is used by the M82592 to request DMA transfer. DRQ0 remains HIGH as long as the M82592 requires DMA transfers. Burst transfers are thus possible.<br>When the M82592 is programmed for Tightly Coupled DMA Interface, the M82592 notifies the DMA controller of the status of transmission or reception, using this pin together with $\overline{EOP}$. |
| DRQ1 | 40 | O | **DMA REQUEST (CHANNEL 1):** This pin is used by the M82592 to request DMA transfer. DRQ1 remains HIGH as long as the M82592 requires DMA transfers. Burst transfers are thus possible.<br>When the M82592 is programmed for Tightly Coupled DMA Interface, the M82592 notifies the DMA controller of the status of transmission or reception, using this pin together with $\overline{EOP}$. |

**Table 1. Pin Description** (Continued)

| Symbol | Pin No. (CQFP) | Type | Name and Function |
|---|---|---|---|
| $\overline{DACK0}$/ $\overline{DACK}$ | 2 | I | **DMA ACKNOWLEDGE (CHANNEL 0):** When LOW, this input signal from the DMA controller notifies the M82592 that the requested DMA cycle is in progress. This signal acts similarly to Chip Select for data and parameter transfers, using DMA channel 0.<br>**DMA ACKNOWLEDGE (CHANNELS 0 AND 1):** When the $\overline{DACK1}$/$\overline{CS1}$/$\overline{EOP}$ pin is programmed to $\overline{CS1}$/$\overline{EOP}$, this pin provides a DMA acknowledge for both channels 0 and 1. Two DMA acknowledge signals from the DMA controller, $\overline{DACK0}$ and $\overline{DACK1}$, must be externally ANDed in this mode of operation. |
| $\overline{DACK1}$ $\overline{CS1}$/$\overline{EOP}$ | 66 | I<br><br><br><br>I/O | This is a multifunction, bidirectional pin which can be programmed to $\overline{DACK1}$ or $\overline{CS1}$/$\overline{EOP}$ during configuration. When it is configured for $\overline{EOP}$, it provides an open-drain output.<br>**DMA ACKNOWLEDGE (CHANNEL 1):** When LOW, this input signal from the DMA controller notifies the M82592 that the requested DMA cycle is in progress. This signal acts similarly to Chip Select for data and parameter transfers, using DMA channel 1.<br>**CHIP SELECT (PORT 1):** When LOW, the M82592 is selected by the CPU for command or status transfer through PORT 1.<br>**END OF PROCESS:** A LOW output signal requests the DMA controller to terminate the active DMA service. |
| CLK | 5 | I/O | **CLOCK:** This pin can be a clock input or output, depending on the state of CLKSRC. If CLKSRC is strapped LOW, this pin is a clock input which provides timing for the M82592 parallel subsystem. If CLKSRC is strapped HIGH, the clock for the M82592 parallel subsystem is generated from the internal clock generator. The CLK pin is then a clock output and provides a clock signal whose frequency can be one-half of, or identical to, the frequency of the internally generated parallel subsystem clock, depending on the state of FREQ.<br>Note that the maximum frequency of the clock signal supplied by the CLK pin is 8 MHz. |

| CLKSRC | FREQ | CLK | | Clock for the Parallel Subsystem |
|---|---|---|---|---|
| | | Type | Signal | |
| 0 (LOW) | Don't Care | I | Clock | Clock as provided on the CLK pin. |
| 1 (HIGH) | 1 | O | Internal Parallel Subsystem Clock Divided by Two | Prescaled clock generated from the internal clock generator. |
| 1 | 0 | O | Internal Parallel Subsystem Clock | Prescaled clock generated from the internal clock generator |

| | | | |
|---|---|---|---|
| CLKSRC | 6 | I | **CLOCK SOURCE:** When strapped LOW, a clock signal on the CLK pin provides timing for the parallel subsystem. When strapped HIGH, timing for the parallel subsystem is internally generated from the clock generator provided in the serial subsystem. The internal prescaler is programmed during configuration to determine the frequency of the clock for the parallel subsystem. |

**Table 1. Pin Description** (Continued)

| Symbol | Pin No. (CQFP) | Type | Name and Function |
|---|---|---|---|
| FREQ | 13 | I | **FREQUENCY:** When strapped LOW, CLK has an output freqency equal to that of the parallel subsystem clock. When strapped HIGH, CLK has an output frequency one-half that of the parallel subsystem clock. The state of this pin is relevant only when CLKSRC is strapped HIGH. |
| X1/X2 | 37/38 | I | **HIGH INTEGRATION MODE OSCILLATOR INPUTS:** These inputs may be used to connect a quartz crystal which controls the internal clock generator for the serial subsystem. When CLKSRC is strapped HIGH, the clock generator also provides a clock for the parallel subsystem.<br>X1 may also be driven by a MOS-level clock whose freqency is 8, 10, 16, or 18 times the bit rate of Transmit/Receive data. X2 must be left floating if X1 is connected to an external MOS clock. |
| $\overline{TxC}$ | 37 | I | **HIGH SPEED MODE TRANSMIT CLOCK:** This signal provides the fundamental timing for the serial subsystem. The clock is also used to transmit data synchronously on the TxD pin. For NRZ encoding, data is transferred to the TxD pin on the HIGH to LOW clock transition. For Manchester encoding, the transmitted bit center is aligned with the LOW to HIGH transition. |
| $\overline{RxC}$ | 38 | I | **RECEIVE CLOCK:** This clock is used to synchronously sample data on the RxD pin. Only NRZ data format is supported for reception. The state of the RxD pin is sampled on the HIGH to LOW transition. |
| $\overline{TCLK}/\overline{CRS}$ | 53 | I<br>O | **CARRIER SENSE:** In High-Speed Mode this pin is Carrier Sense, $\overline{CRS}$, and is used to notify the M82592 that the serial link is active.<br>**TRANSMIT CLOCK:** In High-Integration Mode this pin is Transmit Clock, $\overline{TCLK}$. |
| $\overline{CDT}$ | 52 | I | **COLLISION DETECT:** This input notifies the M82592 that a collision has occurred. In High-Speed Mode a collision is sensed by this pin only when the M82592 is configured for external Collision Detect (external means are then required for collision detection). In High-Integration Mode collisions are sensed by this pin regardless of the internal or external Collision Detect configuration of the M82592. |
| RxD | 48 | I | **RECEIVE DATA:** This pin receives serial data. It must be HIGH when not receiving. |
| TxD | 49 | O | **TRANSMIT DATA:** This pin transmits data to the serial link. It is HIGH when not transmitting. |
| $\overline{RTS}$ | 50 | O | **REQUEST TO SEND:** When this signal is LOW the M82592 notifies the channel that it has data to transmit. It is forced HIGH after a reset or when transmission is stopped. |
| $\overline{CTS}$/LPBK | 51 | I/O | **CLEAR TO SEND:** An active LOW signal which enables the M82592 to start transmitting data. Asserting this signal HIGH stops the transmission.<br>**LOOPBACK:** This pin, in conjunction with a pull-down resistor, can be programmed to provide an active HIGH loopback signal to the external interface device. |
| $V_{CC}$ | 1<br>67<br>68 | | **POWER:** +5V ±10%. |
| $V_{SS}$ | 33<br>34<br>35 | | **GROUND:** 0V. |

# FUNCTIONAL DESCRIPTION

## Internal Architecture

The M82592 consists of a parallel subsystem, a serial subsystem, and a FIFO subsystem (see Figure 1).

### PARALLEL SUBSYSTEM

The parallel subsystem consists of a bus interface unit (BIU), command and status registers, a 24-bit general purpose timer, and three 16-bit event counters.

The BIU provides an 8- and/or 16-bit interface to the external system bus. It handles all data transfers to and from memory (at speeds up to 16 Mbytes/sec.), accepts commands from the CPU, and provides status to the CPU. There are two separate 8-bit I/O ports, Port 0 and Port 1; and two separate 8- or 16-bit DMA channels, Channel 0 and Channel 1. The 8-bit I/O ports are interfaced to the CPU via the data lines $D_0$–$D_7$. The DMA channels can be configured for an 8- or 16-bit data path during initialization, and are typically interfaced to an external DMA controller. When the M82592 is reset by hardware or software, the DMA channels are initialized for an 8-bit data path. The CPU can then configure the M82592 for a 16-bit data path if desired. Once the DMA channels are configured for a 16-bit data path all subsequent DMA transfers are performed on the data lines $D_0$–$D_{15}$. The two DMA channels are independent and can be used for high-performance operations such as simultaneous transmission and reception.

The 24-bit timer consists of a 24-bit maximum count register, a 24-bit count register, and associated control bits in the command registers. Its clock source can be the transmit clock or the parallel subsystem clock. The timer can be programmed to halt or continue on a terminal count with or without causing an interrupt.

The three 16-bit event counters can be programmed to count valid frames, collided frames, and errored (CRC or Alignment) frames. When these event counters are used in Monitor mode, the M82592 is capable of maintaining the network statistics by itself; i.e., without requesting DMA services or causing interrupts to the CPU.

### SERIAL SUBSYSTEM

The serial subsystem consists of a CSMA/CD unit, a data encoder and decoder, collision detect and carrier sense logic, and a clock generator.

The M82592's CSMA/CD unit is highly flexible in implementing the CSMA/CD protocol. It can operate in a variety of IEEE 802.3 and other CSMA/CD LAN environments, including 1BASE5 (StarLAN), 10BASE5 (Ethernet), 10BASE2 (Cheapernet), and the IBM PC Network (Baseband and Broadband). The programmable parameters include:

- Framing (IEEE 802.3 Framing or HDLC Framing)
- Address Field Length
- Station Priority
- Interframe Spacing
- Slot Time
- CRC-32 or CRC-16

The CSMA/CD unit also has a mode of operation which implements deterministic collision resolution (DCR). The DCR algorithm is fully compatible with the MULTIBUS™ II Serial System Bus (SSB) specifications.

The encoder and decoder in the serial subsystem is capable of NRZI, Manchester, and Differential Manchester encoding and decoding at bit rates up to 4 Mb/s in High-Integration Mode, and Manchester encoding at bit rates up to 20 Mb/s in High-Speed Mode. A digital phase-lock loop is used in High-Integration Mode to decode the receive data and to generate the synchronous receive clock.

The collision detect and carrier sense logic generate the internal collision detect and carrier sense signals for the CSMA/CD unit.

The M82592 implements several different internal, logic-based collision detect mechanisms. Two of these, Code Violation and Bit Comparison, have been used in a variety of applications. The Code Violation method defines a collision where a transition edge occurs outside the area of normal transitions (as specified by the data encoding method). For example, if there are no mid-bit cell transitions in the Manchester encoded data, this method interprets that condition as a collision. The Bit Comparison method compares the signature of the transmitted frame to the signature of the received frame. If the signatures are different, a collision is assumed to have occurred. Two other internal collision detect methods are Source Address Comparison and StarLAN CPS (Collision Presence Signal) Recognition. The Source Address Comparison compares the source address field of the transmitted frame to the source address field of the received frame. If the source addresses are different, it assumes that a collision has occurred resulting in data corruption in the source address field. The StarLAN CPS Recognition method looks for the specific collision presence signal defined by the IEEE 802.3 1BASE5

standard. Other abnormal circumstances, such as no carrier for more than one-half slot time in the receive channel during transmission, are interpreted as collisions by the M82592.

In addition to these internal, logic-based collision detection methods, an external means of collision detection can be used in parallel by using the $\overline{CDT}$ input pin.

The clock generator in the serial subsystem is available only in High-Integration Mode and provides timing for the serial subsystem. The clock signal can also be routed to the parallel subsystem, if so desired. The oscillator circuit is designed for use with an external, parallel resonant, fundamental mode crystal. The crystal frequency should be selected at 8X, 10X, 16X, or 18X the required serial bit rate.

#### FIFO SUBSYSTEM

The FIFO subsystem is located between the parallel subsystem and the serial subsystem. It consists of a transmit FIFO, a receive FIFO, and FIFO control logic. The transmit and receive FIFOs are independent of each other and individually provide optimal interfaces between the two subsystems which may have different speeds. There is a total of 64 bytes that can be used for the two separate FIFOs. During configuration these 64 bytes can be divided into one of four possible combinations: 16 and 16 bytes, 16 and 48 bytes, 32 and 32 bytes, or 48 and 16 bytes for the transmit and receive FIFO respectively. The FIFO threshold is also programmed during configuration.

## Programming Model—Register Overview

Figure 3 shows the M82592 internal registers that are directly accessible through the 8-bit I/O ports: Port 0 and Port 1.

Figure 4 shows the Port 0 commands, and Figure 5 shows the Port 1 commands. The two separate I/O ports can be accessed at two different addresses selected by $\overline{CS0}$ and $\overline{CS1}$, or at one address selected by $\overline{CS0}$. When the hardware does not support two chip select signals, port switch commands are used to access both ports alternately at one address. If the SWT-TO-PORT-1 command is executed while in Port 0, the port logically becomes Port 1. Software overhead associated with port switching is eliminated if two chip select signals are supplied in hardware.

The M82592 can be configured to have 4 or 6 bytes of status registers in Port 0 (see Figures 3 and 6). When configured to 4 bytes of status registers the first three status registers (STATUS 0 through 2) contain the information about the last command executed or the last frame received. The last status register, STATUS 3, contains the state of the M82592. When the M82592 is configured to 6 bytes of status registers, the two additional bytes are used to report a more complete status of the most recently received frame.

The status of the timer and event counters is available in the Port 1 status registers as shown in Figure 7.

## M82592 and Host Interaction

The CPU interacts with the M82592 through the system's memory and the M82592's on-chip registers. The CPU creates a data structure in memory, programs the external DMA controller with the start address and byte count of the memory block, and issues a command to the M82592.

The chip select and interrupt lines are used to communicate between the M82592 and the CPU as shown in Figure 9. The interrupt signal is used by the M82592 to attract the CPU's attention. The chip select signal is used by the CPU to attract the M82592's attention. Note that the M82592 does not have any address lines.

There are two kinds of transfers over the bus: command/status and data transfers. The 8-bit command/status transfers are always performed by the CPU. The 8- or 16-bit data transfers are requested by the M82592, and are usually performed by a DMA controller. Table 2 shows the command/status and data-transfer control signals. The CPU writes commands to the M82592 using the $\overline{CS0}$ (or $\overline{CS1}$) and $\overline{WR}$ signals, and reads status using the $\overline{CS0}$ (or $\overline{CS1}$) and $\overline{RD}$ signals. When data transfers are performed, $\overline{DACK0}$ or $\overline{DACK1}$ must be asserted by the DMA controller instead of the Chip Select.

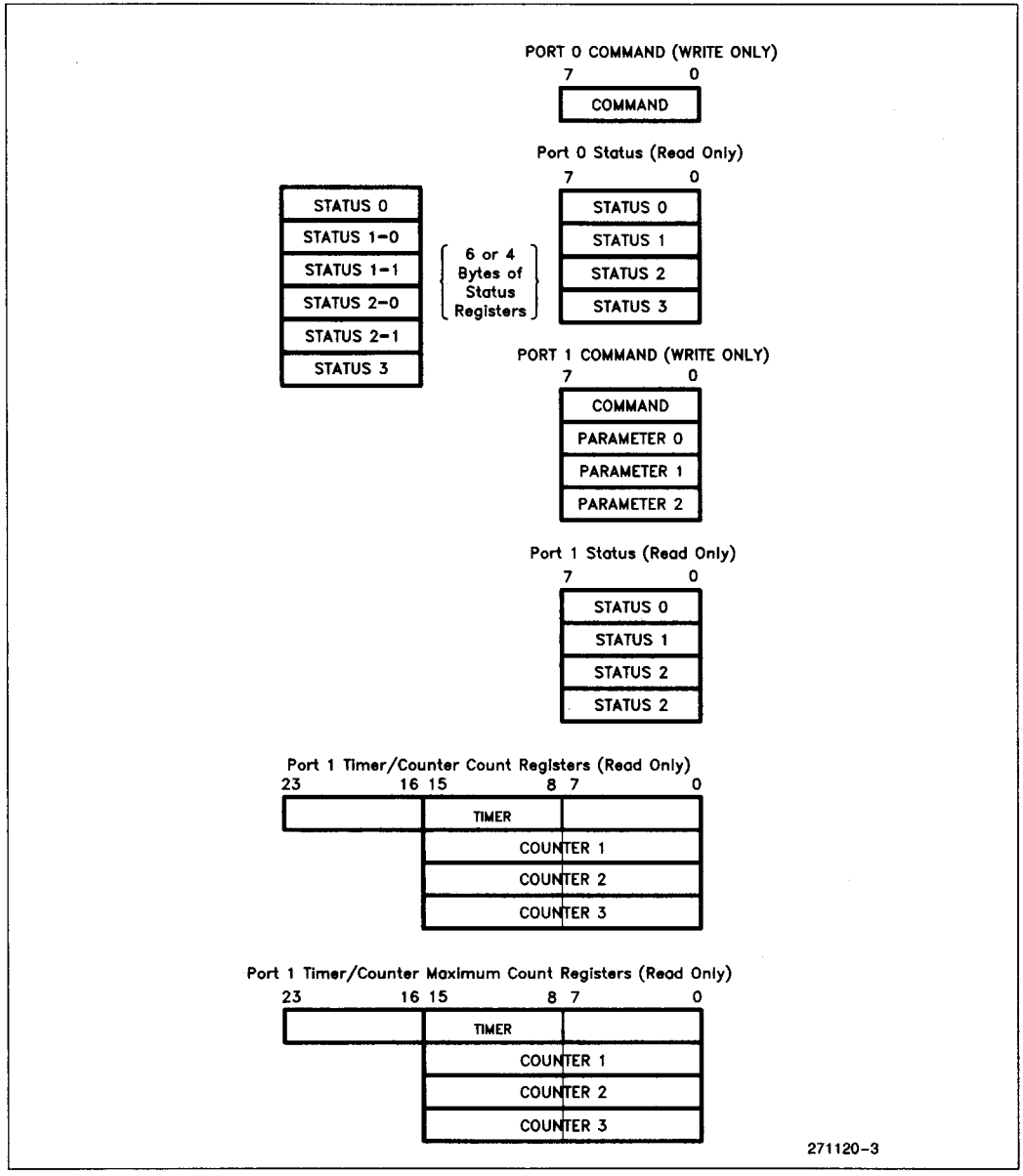



**Figure 3. Programming Model—Directly Accessible Registers (Accessible Through 8-Bit I/O Port[s])**

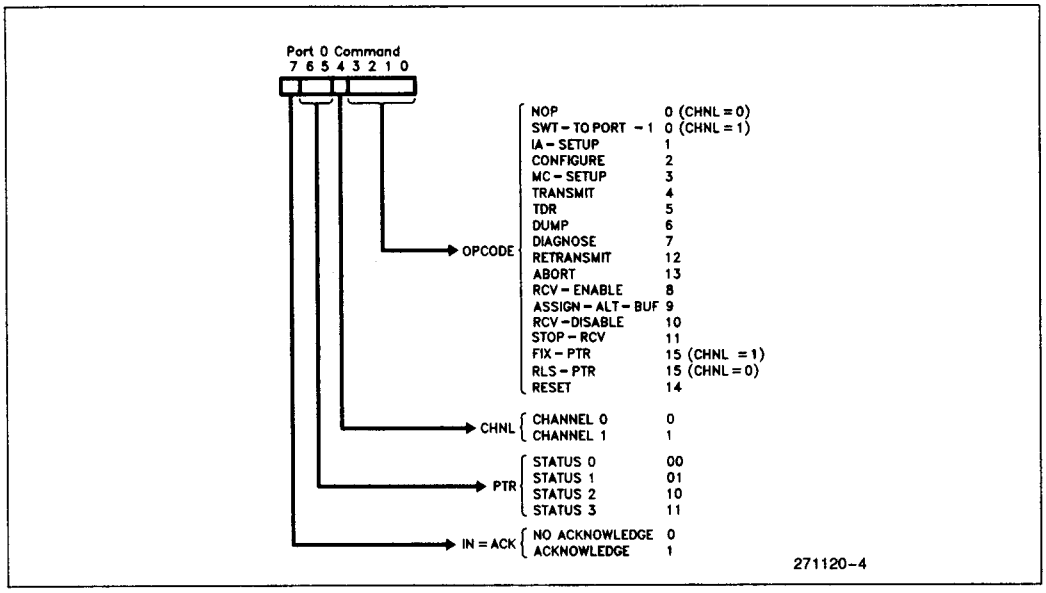



**Figure 4. Port 0 Commands**

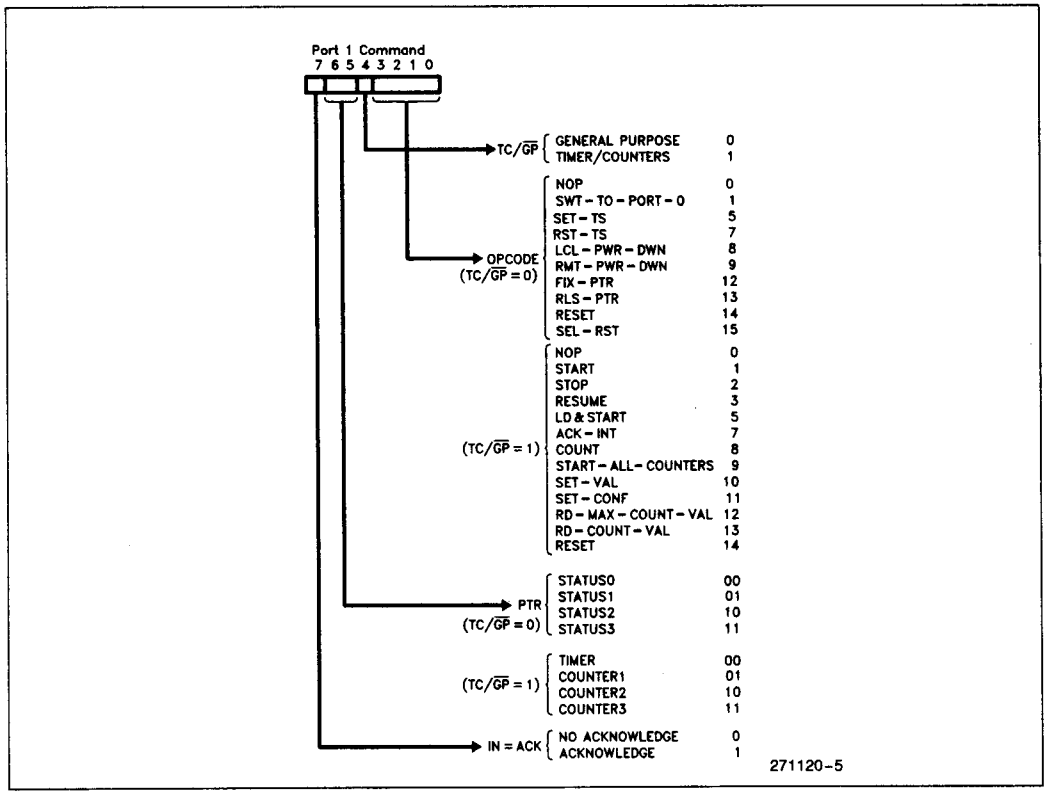
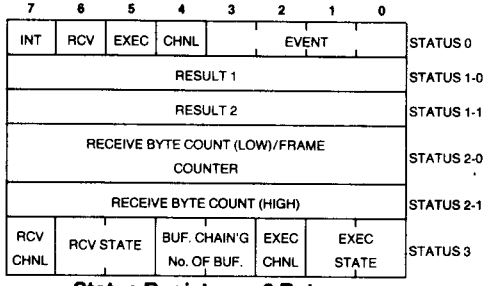
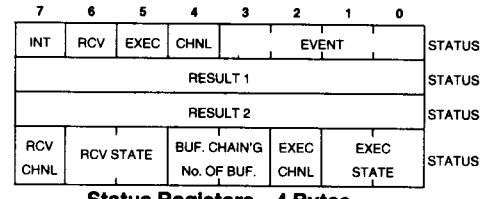



**Figure 5. Port 1 Commands**




**Status Registers—6 Bytes**

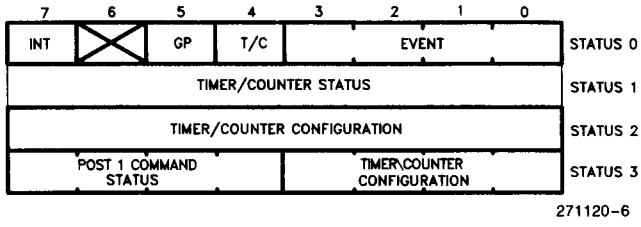



**Status Registers—4 Bytes**

| Events | Value (Status 0) |
|---|---|
| CMOS* | 0 (CHNL = 1) |
| IA-Setup-Done | 1 |
| Configure-Done | 2 |
| MC-Setup-Done | 3 |
| Transmit-Done | 4 |
| TDR-Done | 5 |
| Dump-Done | 6 |
| Diagnose-Passed | 7 |
| End-of-Frame | 8 |
| Request-Next-Buffer | 9 |
| Reception-Aborted | 10 |
| Retransmit-Done | 12 |
| Execution-Aborted | 13 |
| Diagnose-Failed | 15 |

*Available only after Hardware or Software Reset.

**Figure 6. Port 0 Status Registers**




| Timer/Counter Events (T/C = 1) | Value* (Status 0) |
|---|---|
| Timer Expired | Bit 0 = 1 |
| Counter 1 Expired | Bit 1 = 1 |
| Counter 2 Expired | Bit 2 = 1 |
| Counter 3 Expired | Bit 3 = 1 |
| **General Purpose Event (GP = 1)** | **Value* (Status 0)** |
| REM-PWR-UP | 9 |

*The M82592 may have more than one EVENT bit set by the time the CPU reads the Status register.

**Figure 7. Port 1 Status Registers**

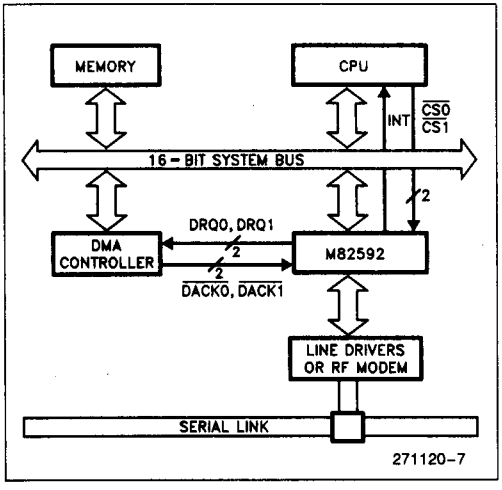



**Figure 8. M82592/Host CPU Interaction**

**Table 2. Data Bus Control Signals and Functions**

| Pin Name | | | Function |
|---|---|---|---|
| $\overline{CS0}$ $\overline{CS1}$* | $\overline{RD}$ | $\overline{WR}$ | |
| 1 | X | X | No Transfer to/from Command/Status |
| 0 | 1 | 1 | |
| 0 | 0 | 0 | Illegal |
| 0 | 0 | 1 | Read from Status Register |
| 0 | 1 | 0 | Write to Command Register |
| $\overline{DACK0}$ $\overline{DACK1}$* | $\overline{RD}$ | $\overline{WR}$ | |
| 1 | X | X | No DMA Transfer |
| 0 | 1 | 1 | |
| 0 | 0 | 0 | Illegal |
| 0 | 0 | 1 | Data Read from DMA Channel 0 (or 1) |
| 0 | 1 | 0 | Data Write to DMA Channel 0 (or 1) |

*Only one of $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$ may be active at any time.

To initiate an operation such as Transmit or Configure (see Figure 4), the command from the CPU must first be written to the M82592. Any parameters or data associated with the command are transferred from memory to the M82592 using DMA. Upon completion of the operation, the M82592 updates the appropriate status registers and sends an interrupt to the CPU.

## Frame Transmission

To transmit a frame, the CPU prepares a Transmit Data Block in memory as shown in Figure 9. Its first two bytes specify the length of the rest of the block. The next few bytes (up to six) contain the destination address of the station the frame is being sent to. The rest of the block is the data field. The CPU programs the DMA controller with the start address of the block, length of the block, and other control information and then issues a Transmit command to the M82592. Upon receiving this command, the M82592 fetches the first two bytes of the block to determine its length. If the link is free and the first data byte was fetched, the M82592 begins transmitting the preamble and concurrently fetches more bytes from the Transmit Data Block and loads them into the transmit FIFO to keep them ready for transmission.

The destination address is transmitted after the preamble. This is followed by the source or the station individual address, which was previously stored in the M82592 by the IA-Setup command. After this, the entire information field is transmitted, followed by a CRC field calculated by the M82592. If a collision is encountered during transmission of the frame, then the transmission is aborted after a jam pattern is sent. If the collision is detected during preamble or SFD (Start Frame Delimiter) transmission, the M82592 transmits the jam pattern after the SFD is transmitted. An interrupt is then generated to inform the CPU of the unsuccessful transmission due to a collision.

The CPU reinitializes the DMA controller and issues a Retransmit command to the M82592. Retransmission is done by the CPU exactly as the Transmit command is done, except the Retransmit command keeps track of the number of collisions encountered. When the M82592 gets the Retransmit command and the backoff timer is expired, it transmits the frame again. Retransmission is repeated until the attempt is successful, or until the preprogrammed retry number expires.

If the M82592 is programmed to generate the $\overline{EOP}$ signal to the M82380 DMA controller, or if it is used with a DMA controller which implements the Tightly Coupled Interface, retransmission is performed without CPU intervention.

## Frame Reception

The M82592 can receive frames when its receiver has been enabled. The M82592 checks for an address match for an Individual address, a Multicast address, or a Broadcast address. In the Promiscuous mode the M82592 receives all frames. When the address match is successful, the M82592 transfers the frame to memory using the DMA controller. Before enabling the receiver, it is the CPU's responsibility to make a memory buffer area available to the receiver and to properly program the starting address of the DMA controller. The received frame is transferred to the memory buffer in the format shown in Figure 10. This method of reception is called Single Buffer reception; the entire frame is contained in one continuous buffer. Upon completion of reception, the status of the reception is appended at the end of the received frame in the memory buffer, and the total number of bytes transferred to the memory buffer is loaded into the internal status registers 1 and 2. An interrupt is then generated to inform the CPU of the frame reception.

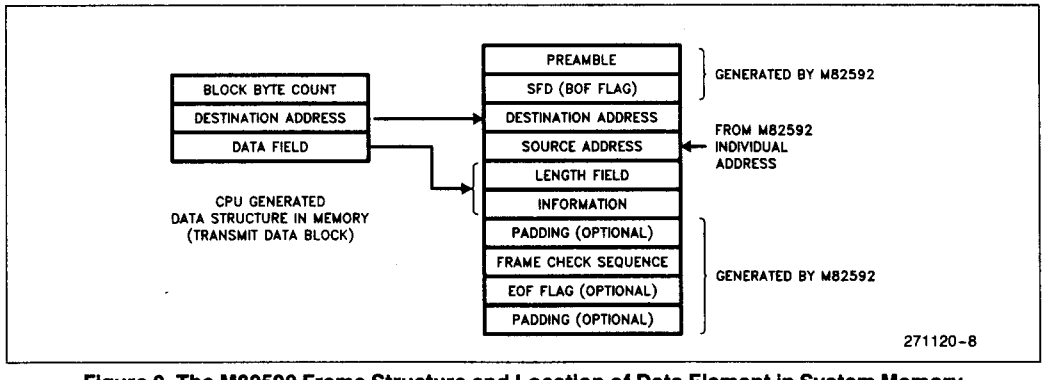



**Figure 9. The M82592 Frame Structure and Location of Data Element in System Memory**

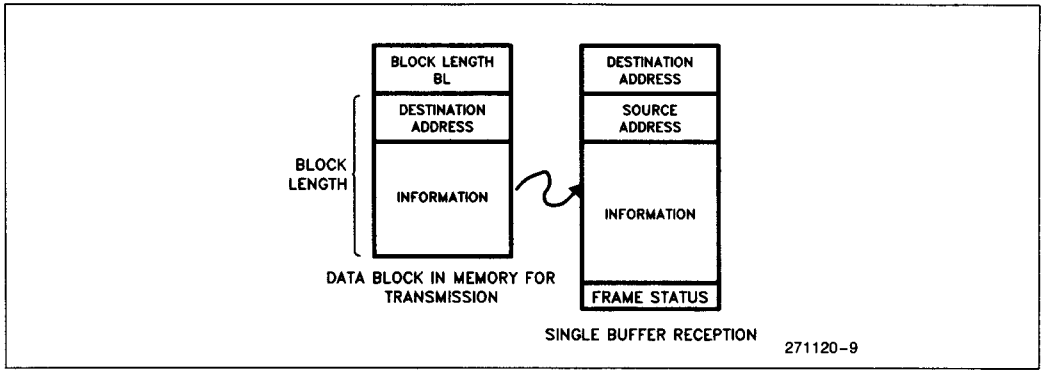



**Figure 10. Single Buffer Reception**

If the frame size is unknown, memory usage can be optimized by using Multiple Buffer reception. In this mode of operation, the CPU and DMA controller can dynamically allocate memory space as it receives frames. This method requires both DMA channels to receive the frame alternately. As frame reception begins, the M82592 interrupts the CPU and automatically requests assignment of the next available buffer. The CPU does this and loads the second DMA channel with the next buffer's information so the M82592 can immediately switch to the other channel when the current buffer is full. When the M82592 switches from the first to the second buffer it again interrupts the CPU and requests another buffer to be allocated on the previous channel. This process continues until the entire frame is received. The received frame is spread over multiple memory buffers. The link between the buffers is easily maintained by the CPU, using a buffer chain descriptor structure in memory as shown in Figure 11. This dynamic allocation of memory buffers results in efficient use of available storage when handling frames of widely differing sizes.

If the M82592 is programmed to generate the $\overline{EOP}$ signal to the M82380 DMA controller, or if it is used with a DMA controller which implements the Tightly Coupled Interface, buffer reclamation and more advanced data structures for the buffer area can significantly improve system performance.

## $\overline{EOP}$ Signal to the DMA Controller

The M82592 can be programmed to assert the $\overline{EOP}$ signal to the M82380 DMA controller when one or more of the following occurs:

- A collision during transmission
- An error (CRC or alignment) during reception
- A good frame reception

If the M82380 is programmed for Auto-initialize mode and if the M82592 is programmed to assert the $\overline{EOP}$ signal on a collision during transmission, the retransmission following a collision is done automatically by the M82592. The M82592 will retransmit the same frame from the same memory area without CPU intervention. When the M82592 is programmed for this mode it does not interrupt the CPU upon a collision, and the CPU does not need to issue a Retransmit command to the M82592. The CPU is interrupted only after a successful transmission or retransmission, or after a transmission failure, such as DMA underrun.

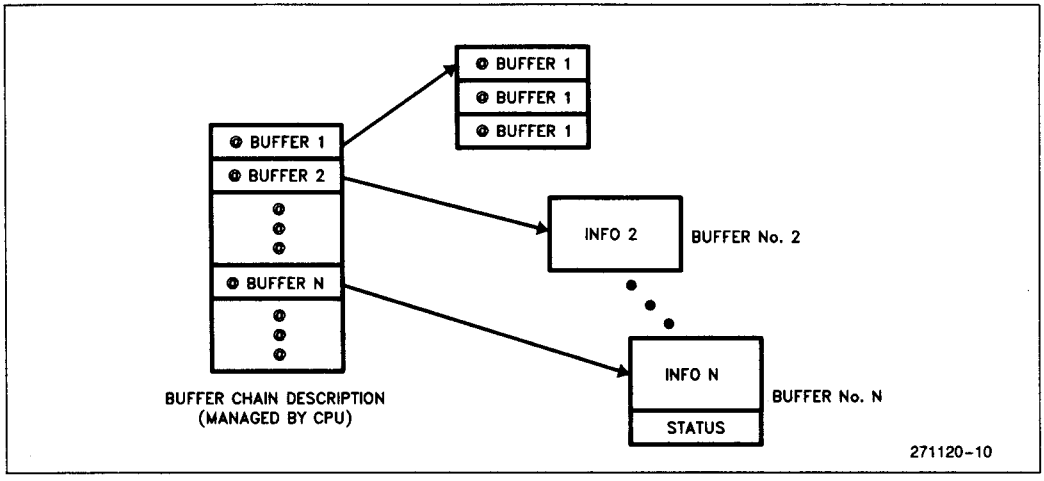



**Figure 12. Multiple Buffer Reception**

If the M82592 is programmed to assert the $\overline{EOP}$ signal when an error occurs during reception, the M82380 in Auto-initialize mode will be able to reclaim the memory area which would otherwise be wasted for the errored frame reception. If the M82592 is programmed to assert $\overline{EOP}$ at the end of a frame reception, automatic buffer switching can be accomplished by alternating the DMA channels with the M82380. When the 82380 is used, the buffer switching can be done with only one DMA channel.

The $\overline{EOP}$ signal must be derived from the $\overline{DACK1}$/$\overline{CS1}$/$\overline{EOP}$ pin using external logic (see Figure 12).

## M82592 Tightly Coupled Interface

The M82592 has a mode of operation called "Tightly Coupled Interface." In this mode the M82592 provides a tightly coupled interface to a DMA controller in order to execute some of the time-critical processes of the CSMA/CD protocol without any CPU intervention. By implementing the Tightly Coupled Interface in a DMA controller, operations such as automatic retransmission, continuous back-to-back frame reception, and transmit and/or receive buffer chaining can be accomplished.

The M82592 provides the status of the current active transmission or reception to the DMA controller by using the DRQ and $\overline{EOP}$ signals at the end of every DMA cycle. The status is encoded according to Table 3. As long as the M82592 generates DRQ High and $\overline{EOP}$ Floating at the rising edge of $\overline{RD}$ or $\overline{WR}$, the DMA controller repeats DMA transfers. If the transmission is completed without collisions or if the reception is good (no collision, no CRC, or no Alignment error), then DRQ and $\overline{EOP}$ both become Low at the end of a DMA transfer which follows the last DMA data transfer. If the transmission encountered a collision or if the reception had an error, DRQ becomes High and $\overline{EOP}$ becomes Low. The DMA controller must decode these signals appropriately and must reinitialize the DMA channel so it can retransmit the same frame or reclaim the otherwise wasted buffer. It is the DMA controller's responsibility to reprogram itself for the next appropriate operation.

## Network Management and Diagnostics

The M82592 provides a large set of diagnostic and network management functions including: internal and external loopback, monitor mode, optional capture of all frames regardless of destination address (Promiscuous mode), and time domain reflectometry for locating fault points in the network cable. The M82592 Dump command ensures software reliability by dumping the contents of the M82592 internal registers into the system memory.

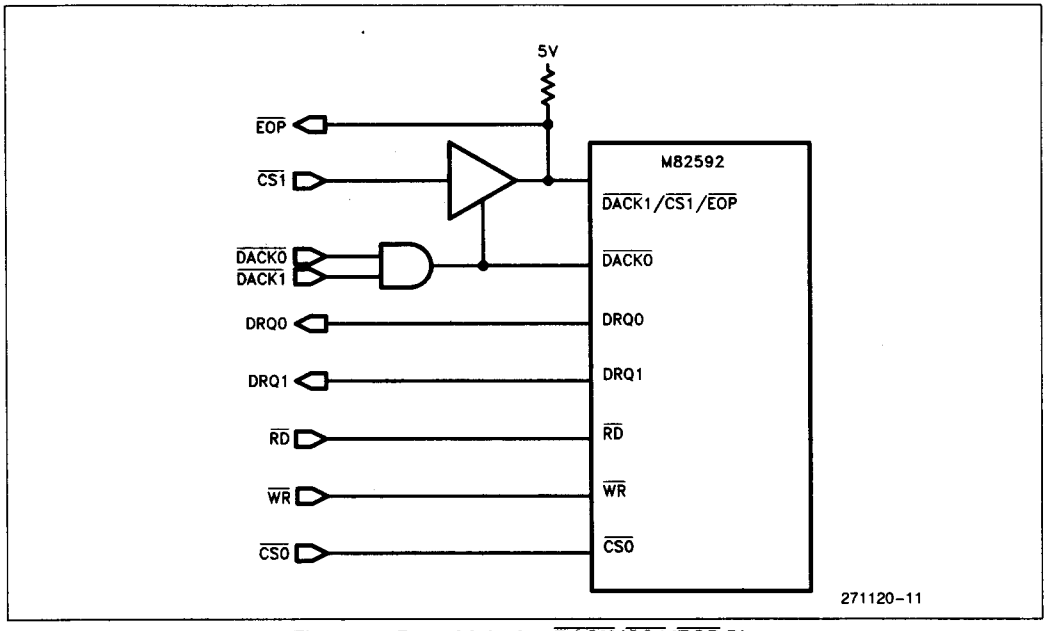



**Figure 12. Demultiplexing DACK/CS1/EOP Pin**

**Table 3. Transmit/Receive Status Encoding on DRQ and EOP**

| DRQ | EOP | Status Information |
|---|---|---|
| 0 | Hi-Z | Idle |
| 1 | Hi-Z | DMA Transfer |
| 0 | 0 | Transmission or Reception Terminated OK |
| 1 | 0 | Transmission or Reception Aborted |

## Other Enhancements

The M82592 has many functional and performance enhancements. This section lists some of these enhancements which are not covered in other sections.

1. **Multi-IA**—The M82592 implements multiple-individual address (Multi-IA) filtering. It can receive more than one IA frame in this mode.
2. **Power Down Modes**—Two power down modes, Local Power Down and Remote Power Down, are available. When the M82592 is in Remote Power Down mode, it can be powered up remotely by sending a special frame to it.
3. **Automatic Padding and IEEE 802.3 Length Field**—If a frame to be transmitted is shorter than the configured Slot Time, the M82592 automatically appends pad bytes up to the shortest frame greater than the Slot Time. If the data field of a received frame is longer than the byte count indicated in the Length field, the extra bytes are stripped automatically according to the Length field. Erroneous conditions are detected and reported by the M82592. An example of such conditions is reception of a frame which is shorter than the byte count indicated in the Length field.
4. **Automatic Retransmission on Collision During Preamble**—The M82592 can be programmed to retransmit automatically if it detects a collision during transmission of the preamble.
5. **On-Chip Jabber Inhibit Function**—The M82592 can be programmed to provide an on-chip jabber inhibit function.
6. **CRC Transfer to Memory**—The M82592 can be programmed to transfer the CRC field of a received frame into memory.
7. **Loopback Signal to the 82C501**—The M82592 can be programmed to provide an active High loopback signal to the 82C501.
8. **StarLAN**—The M82592 can be configured to recognize the IEEE 802.3 1BASE5 Collision Presence Signal (CPS). In this mode it also delays deactivation of the RTS signal at the end of a frame transmission in order to insert an end-of-frame marker as required by the standard.

## ABSOLUTE MAXIMUM RATINGS*

Case Temperature ($T_C$) Under Bias
CQFP ..... −55°C to +125°C
Storage Temperature ..... −65°C to +150°C
Voltage on Any Pin with
Respect to Ground ..... −1V to +7V
Power Dissipation ..... 550 mW

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

*NOTICE: Specifications contained within the following tables are subject to change.*

## Operating Conditions

| Symbol | Description | Min | Max | Units |
|---|---|---|---|---|
| $T_C$ | Case Temperature (Instant On) | −55 | +125 | °C |
| $V_{CC}$ | Digital Supply Voltage | 4.50 | 5.50 | V |

## D.C. CHARACTERISTICS (Over Specified Operating Conditions)

TxC and RxC have MOS levels (see $V_{MIL}$, $V_{MIH}$). All other signals have TTL levels (see $V_{IL}$, $V_{IH}$, $V_{OL}$, $V_{OH}$).

| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage (TTL) | −0.5 | 0.8 | V | |
| $V_{IH}$ | Input High Voltage (TTL) | 2.0 | $V_{CC}$ +0.5 | V | |
| $V_{OL}$ | Output Low Voltage (TTL) | | 0.45 | V | $I_{OL}$ = 2.0 mA |
| $V_{OH}$ | Output High Voltage (TTL) | 2.4 | | V | $I_{OH}$ = −400 μA |
| $V_{MIL}$ | Input Low Voltage (MOS) | −0.5 | 0.6 | V | |
| $V_{MIH}$ | Input High Voltage (MOS) | 3.9 | $V_{CC}$ +0.5 | V | |
| $I_{LI}$ | Input Leakage Current | | ±10 | μA | $0V \le V_{IN} \le V_{CC}$ |
| $I_{LIO}$ | I/O Leakage Current | | ±10 | μA | $0.45V \le V_{OUT} \le V_{CC} - 0.45V$ |
| $C_{IN}$ | Capacitance of Input Buffer | | 10 | pF | Frequency = 1 MHz |
| $C_{IO}$ | Capacitance of I/O Buffer | | 20 | pF | Frequency = 1 MHz |
| $I_{CC}$ | Power Supply Current | | 80 | mA | (Note 1) |
| | | | 5 | mA | (Note 2) |
| | | | 1 | mA | (Note 3) |
| | | | | | (Note 4) |

**NOTES:**
1. System side (CLK) at 16 MHz, serial side ($\overline{TxC}$) at 20 MHz in High-Speed Mode.
2. Remote power-up mode.
3. Power-down mode—all inputs connected to $V_{CC}$ level.
4. General formulas for current are: (a) f(CLK) × 2.9 + f($\overline{T_xC}$) × 1.8 for High Speed Mode. (b) f(CLK) × 2.9 + f($\overline{TFC}$) × 0.2 + f($\overline{TFC}$) × 1.8/SR, where SR is the sampling rate in High-Integration Mode, and f = frequency in MHz.

## A.C. CHARACTERISTICS $C_L$ on all outputs is 20 pF–125 pF unless otherwise specified.

| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **SYSTEM CLOCK INPUT PARAMETERS** | | | | | |
| $t_1$ | CLK Cycle Time | 62.5 | | ns | |
| $t_2$ | CLK Low Time | 27 | | ns | (Note 5) |
| $t_3$ | CLK High Time | 27 | | ns | (Note 5) |
| $t_4$ | CLK Rise Time | | 5 | ns | (Note 1) |
| $t_5$ | CLK Fall Time | | 5 | ns | (Note 2) |
| **SYSTEM CLOCK OUTPUT PARAMETERS** | | | | | |
| $t_{98}$ | CLK Cycle Time | 120 | | ns | (Notes 4, 7) |
| $t_{99}$ | CLK Low Time | 50 | | ns | (Note 4) |
| $t_{100}$ | CLK High Time | 50 | | ns | (Note 4) |
| $t_{101}$ | CLK Rise Time | | 15 | ns | (Notes 1, 4) |
| $t_{102}$ | CLK Fall Time | | 15 | ns | (Notes 2, 4) |
| **RESET PARAMETERS** | | | | | |
| $t_6$ | Reset Active to Clock Low | 20 | | ns | (Note 3) |
| $t_8$ | Reset Pulse Width | $4t_1$ | | ns | |
| $t_9$ | Control Inactive after Reset | | $2t_1$ | ns | |
| **INTERRUPT PARAMETERS** | | | | | |
| $t_{10}$ | CLK High to Interrupt Active | | 55 | ns | (Note 4) |
| $t_{11}$ | $\overline{WR}$ Inactive to Interrupt | | 55 | ns | (Note 4) |
| $t_{103}$ | Int Low to Int High Gap | $2t_1$ | | ns | (Note 4) |
| **WRITE PARAMETERS** | | | | | |
| $t_{12}$ | $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$ Setup to $\overline{WR}$ Low | 0 | | ns | |
| $t_{13}$ | $\overline{WR}$ Pulse Width | 55 | | ns | |
| $t_{14}$ | $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$ Hold after $\overline{WR}$ High | 0 | | ns | |
| $t_{15}$ | Data Setup to $\overline{WR}$ High | 30 | | ns | |
| $t_{16}$ | Data Hold after $\overline{WR}$ High | 3 | | ns | |
| $t_{94}$ | Write Cycle Time | $2t_1$ | | ns | |
| $t_{96}$ | $\overline{WR}$ Inactive Time | 55 | | ns | |

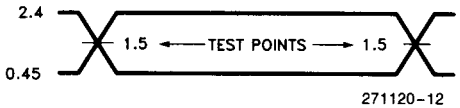
## A.C. CHARACTERISTICS $C_L$ on all outputs is 20 pF–125 pF unless otherwise specified. (Continued)

| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **READ PARAMETERS** | | | | | |
| $t_{17}$ | $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$ or $\overline{DACK1}$ Setup to $\overline{RD}$ Low | 0 | | ns | |
| $t_{18}$ | $\overline{RD}$ Pulse Width | 55 | | ns | |
| $t_{19}$ | $\overline{CS0}$, $\overline{CS1}$, $\overline{DACK0}$, or $\overline{DACK1}$ Hold after $\overline{RD}$ High | 0 | | ns | |
| $t_{20}$ | $\overline{RD}$ Low to Data Valid | | 45 | ns | |
| $t_{21}$ | Data Float after $\overline{RD}$ High | 5 | 40 | ns | |
| $t_{95}$ | Read Cycle Time | $2t_1$ | | ns | |
| $t_{97}$ | $\overline{RD}$ Inactive Time | 55 | | ns | |
| **DMA PARAMETERS** | | | | | |
| $t_{22}$ | CLK Low to DRQ0 or DRQ1 Active | | 55 | ns | (Note 4) |
| $t_{23}$ | $\overline{WR}$ or $\overline{RD}$ Low to DRQ0 or DRQ1 Inactive | | 45 | ns | (Note 4) |
| $t_{104}$ | $\overline{WR}$ or $\overline{RD}$ High to DRQ0 or DRQ1 Inactive—82560 Retransmit or Receive Buffer Reclaim | 2.5 | 65 | ns | (Note 4) |
| $t_{105}$ | $\overline{WR}$ or $\overline{RD}$ Low to $\overline{EOP}$ Active | | 45 | ns | (Note 6) |
| $t_{106}$ | $\overline{EOP}$ Float after $\overline{DACK0}$ or $\overline{DACK1}$ Going Inactive | | 40 | ns | (Note 6) |

**NOTES:**
1. 0.8V to 2.0V.
2. 2.0V to 0.8V.
3. To guarantee recognition on the next clock.
4. $C_L$ = 50 pF.
5. Measured at 1.5V.
6. Open drain I/O pin.
7. None of the A.C. parameters are related to the CLK output pin.

### A.C. TESTING INPUT/OUTPUT WAVEFORMS

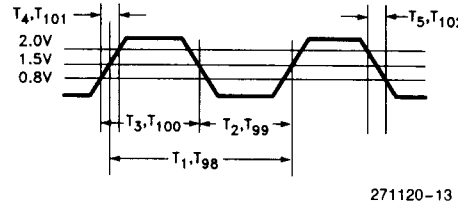
**TTL Input/Output Voltage Levels for Timing Measurements**




271120-12

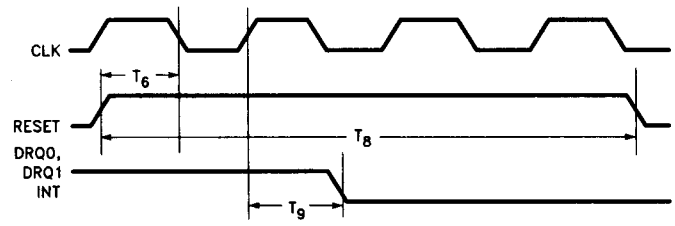
A.C. testing inputs are driven at 2.4V for a Logic "1" and 0.45V for a Logic "0". Timing measurements are made at 1.5V for both a Logic "1" and "0". Rise and fall times are measured between 0.8V and 2.0V.

**System Clock I/O Voltage Levels for Timing Measurements**

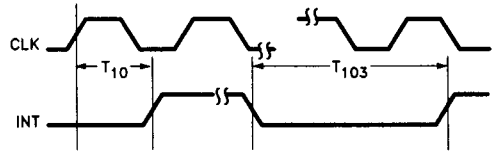



271120-13

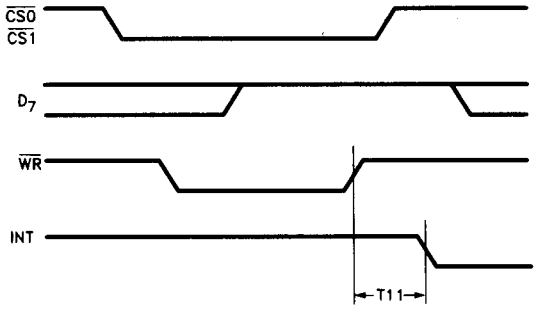
## RESET TIMING



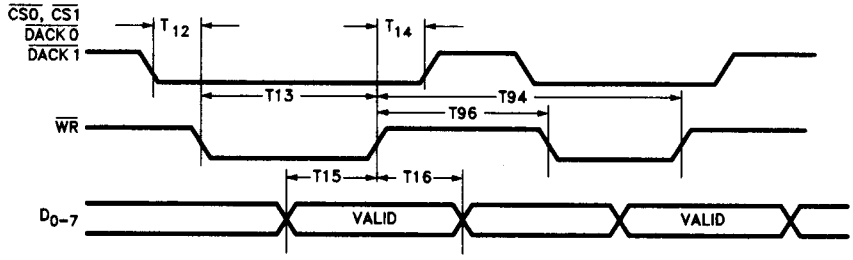

## INTERRUPT TIMING (GOING ACTIVE)




## INTERRUPT TIMING (GOING INACTIVE)

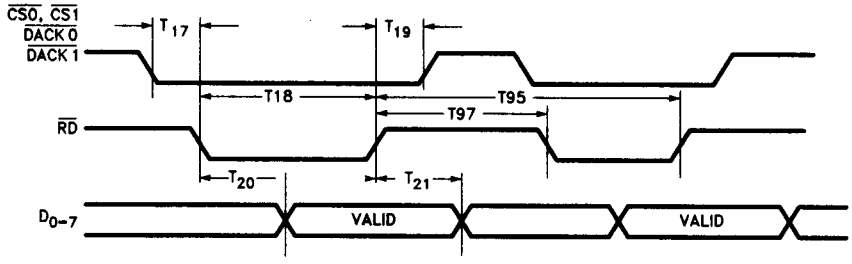


## WRITE TIMING



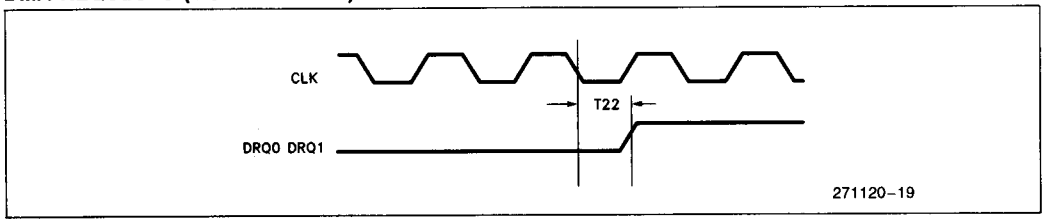

**NOTE:**
DACK0 or DACK1 may remain active continuously for consecutive DMA cycles.

## READ TIMING

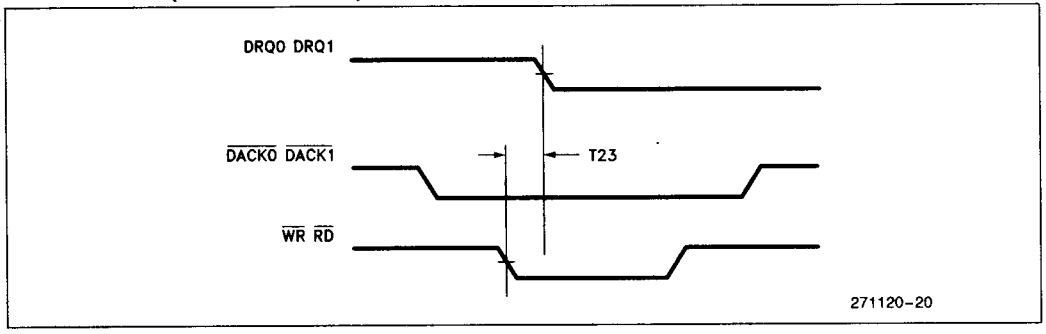



**NOTE:**
DACK0 or DACK1 may remain active continuously for consecutive DMA cycles.

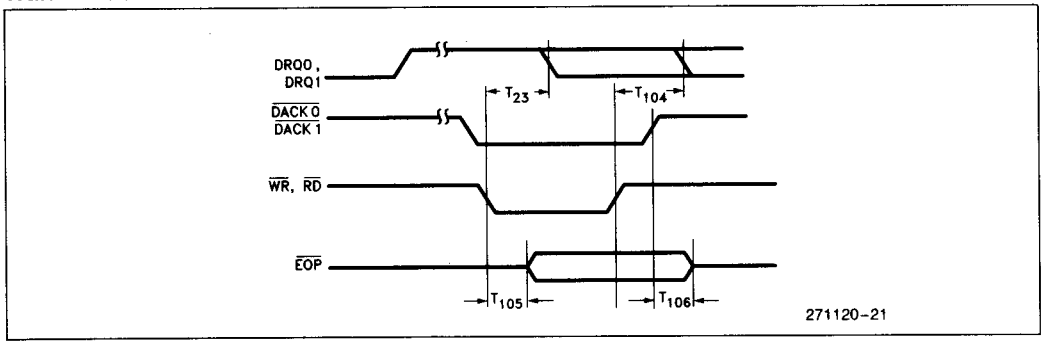
## DMA REQUESTS (GOING ACTIVE)




## DMA REQUEST (GOING INACTIVE)




## TIGHTLY COUPLED INTERFACE



## SERIAL INTERFACE A.C. TIMING CHARACTERISTICS

### HIGH INTEGRATION MODE

$\overline{TFC}$ is the Crystal or Serial Clock Input at X1.

| | X8 Sampling | X10 Sampling | X16 Sampling | X18 Sampling |
|---|---|---|---|---|
| **For $\overline{TFC}$ Frequency = 1 MHz to 32 MHz (High)** | | | | |
| $\overline{TCLK}$ Frequency | 0.125 MHz–4 MHz | 100 kHz– 3.2 MHz | 62.5 kHz–2 MHz | 55.6 kHz–1.78 MHz |
| $t_{29}$ = $\overline{TCLK}$ Cycle Time | $8 \times t_{24}$ | $10 \times t_{24}$ | $16 \times t_{24}$ | $18 \times t_{24}$ |
| $t_{30}$ = $\overline{TCLK}$ High High Time | $t_{24}$ (Typically) | $t_{24}$ (Typically) | $t_{24}$ (Typically) | $t_{24}$ (Typically) |
| $t_{31}$ = $\overline{TCLK}$ Low Time | $7 \times t_{24}$ (Typically) | $9 \times t_{24}$ (Typically) | $15 \times t_{24}$ (Typically) | $17 \times t_{24}$ (Typically) |
| **For $\overline{TFC}$ Frequency = 0 to 1 MHz (Low)** | | | | |
| | **X8 Sampling** | **X10 Sampling** | **X16 Sampling** | **X18 Sampling** |
| $\overline{TCLK}$ Frequency | 0–0.125 MHz | 0–100 kHz | 0–62.5 kHz | 0–55.6 kHz |
| $t_{29}$ = $\overline{TCLK}$ Cycle Time | $8 \times t_{24}$ | $10 \times t_{24}$ | $16 \times t_{24}$ | $18 \times t_{24}$ |
| $t_{30}$ = $\overline{TCLK}$ High Time | $t_{25}$ (Typically) | $t_{25}$ (Typically) | $t_{25}$ (Typically) | $t_{25}$ (Typically) |
| $t_{31}$ = $\overline{TCLK}$ Low Time | $7 \times t_{24} + t_{26}$ (Typically) | $9 \times t_{24} + t_{26}$ (Typically) | $15 \times t_{24} + t_{26}$ (Typically) | $17 \times t_{24} + t_{26}$ (Typically) |

**NOTES:**
X10 and X18 are available only for Manchester or Differential Manchester encoding/decoding.
$t_{24}$ = Serial Clock Cycle Time.
$t_{25}$ = Serial Clock High Time.
$t_{26}$ = Serial Clock Low Time.

### HIGH SPEED MODE

- Applies for $\overline{TxC}$, $\overline{RxC}$
- $f_{max}$ = 20 MHz ±100 ppm
- For Manchester, symmetry is required:

$t6_4, t_{64} = \frac{1}{2f} \pm 5\%$

## HIGH INTEGRATION MODE (Continued)

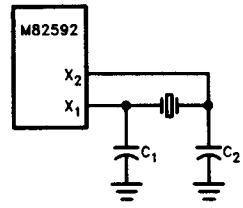
| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **EXTERNAL (FAST) CLOCK PARAMETERS** | | | | | |
| $t_{24}$ | Fast Clock ($\overline{TFC}$) Cycle Time | 31.25 | | ns | (Notes 1, 16) |
| $t_{25}$ | $\overline{TFC}$ High Time | (Note 13) | | ns | (Note 1, 17) |
| $t_{26}$ | $\overline{TFC}$ Low Time | 12 | | ns | (Notes 1, 17) |
| $t_{27}$ | $\overline{TFC}$ Rise Time | | 3 | ns | (Note 1) |
| $t_{28}$ | $\overline{TFC}$ Fall Time | | 3 | ns | (Note 1) |
| **TRANSMIT CLOCK PARAMETERS** | | | | | |
| $t_{29}$ | Transmit Clock ($\overline{TCLK}$) Cycle Time | (Note 13) | | ns | (Note 2) |
| $t_{30}$ | $\overline{TCLK}$ High Time | (Note 7) | | ns | (Note 2) |
| $t_{31}$ | $\overline{TCLK}$ Low Time | (Note 8) | | ns | (Note 2) |
| $t_{32}$ | $\overline{TCLK}$ Rise Time | | 10 | ns | (Note 2) |
| $t_{33}$ | $\overline{TCLK}$ Fall Time | | 10 | ns | (Note 2) |
| **TRANSMIT DATA PARAMETERS (MANCHESTER, DIFFERENTIAL MANCHESTER)** | | | | | |
| $t_{34}$ | TxD Transition-Transition | (Note 14) | | ns | |
| $t_{35}$ | $\overline{TCLK}$ Low to TxD Mid Bit Cell Transition | | (Note 10) | ns | (Note 2) |
| $t_{36}$ | $\overline{TCLK}$ Low to TxD Bit Cell Boundary Transition | | (Note 9) | ns | (Note 2) |
| $t_{37}$ | TxD Rise Time | | 10 | ns | (Note 2) |
| $t_{38}$ | TxD Fall Time | | 10 | ns | (Note 2) |
| **TRANSMIT DATA PARAMETERS (NRZI)** | | | | | |
| $t_{39}$ | TxD Transition-Transition | (Note 15) | ns | | |
| $t_{40}$ | $\overline{TCLK}$ Low to TxD Transition | | (Note 9) | ns | (Note 2) |
| $t_{41}$ | TxD Rise Time | | 10 | ns | (Note 2) |
| $t_{42}$ | TxD Fall Time | | 10 | ns | (Note 2) |
| **$\overline{RTS}$, $\overline{CTS}$ PARAMETERS** | | | | | |
| $t_{43}$ | $\overline{TCLK}$ Low to $\overline{RTS}$ Low | | (Note 9) | ns | (Note 2) |
| $t_{44}$ | $\overline{CTS}$ Low to $\overline{TCLK}$ Low | 35 | | ns | |
| $t_{45}$ | $\overline{TCLK}$ Low to $\overline{RTS}$ High | | (Note 9) | ns | (Note 2) |
| $t_{46}$ | $\overline{TCLK}$ Low to $\overline{CTS}$ Invalid $\overline{CTS}$ Hold Time | 10 | | ns | (Notes 3, 12) |
| $t_{47}$ | $\overline{CTS}$ High to $\overline{TCLK}$ Low; $\overline{CTS}$ Setup Time to Stop Transmission | 35 | | ns | (Note 3) |
| **IFS PARAMETERS** | | | | | |
| $t_{48}$ | Interframe Delay | (Note 4) | | ns | |

## HIGH INTEGRATION MODE (Continued)

| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **COLLISION DETECT PARAMETERS** | | | | | |
| $t_{49}$ | $\overline{CDT}$ Low to $\overline{TCLK}$ High External Collision Detect Setup Time | 35 | | ns | (Note 12) |
| $t_{50}$ | $\overline{CDT}$ High to $\overline{TCLK}$ High | 35 | | ns | (Note 12) |
| $t_{51}$ | $\overline{TCLK}$ High to $\overline{CDT}$ Inactive $\overline{CDT}$ Hold Time | 10 | | ns | (Note 12) |
| $t_{52}$ | $\overline{CDT}$ Low to Jamming Start | | (Note 5) | ns | |
| $t_{53}$ | Jamming Period | (Note 6) | | ns | |
| **RECEIVED DATA PARAMETERS (MANCHESTER, DIFFERENTIAL MANCHESTER)** | | | | | |
| $t_{54}$ | RxD Transition-Transition | 115 | | ns | (Note 14) |
| $t_{55}$ | RxD Rise Time | | 10 | ns | |
| $t_{56}$ | RxD Fall Time | | 10 | ns | |
| **RECEIVED DATA PARAMETERS (NRZI)** | | | | | |
| $t_{57}$ | RxD Transition-Transition | 240 | | ns | (Note 11) |
| $t_{58}$ | RxD Rise Time | | 10 | ns | |
| $t_{59}$ | RxD Fall Time | | 10 | ns | |
| **EXTERNAL LOOPBACK PARAMETERS** | | | | | |
| $t_{107}$ | $\overline{TCLK}$ Low to LPBK High | | 50 | ns | (Note 2) |
| $t_{108}$ | $\overline{TCLK}$ Low to LPBK Float | | 50 | ns | (Note 2) |

**NOTES:**
1. MOS levels.
2. 1 TTL load + 50 pF.
3. Abnormal end of transmission: $\overline{CTS}$ expires before $\overline{RTS}$.
4. Programmable value: $t_{28} = N_{IFS} \times t_{29}$ (ns).
   $N_{IFS}$: the IFS configuration value.
   If $N_{IFS}$ is less than 12 then it is enforced to 12.
5. Programmable values:
   $t_{52} = N_{CDF} \times t_{29} + (12 \text{ to } 15) \times t_{29}$ (if collision occurs after preamble).
   $N_{CDF}$: The collision detect filter configuration value.
6. $t_{53} = 32 \times t_{29}$
7. Depends on frequency range:
   High range: $t_{24} - 10$ ns
   Low range: $t_{25} - 10$ ns
8. $t_{31} = t_{29} - t_{30} - t_{32} - t_{33}$
9. $2 \times t_{24} + 40$ ns for 8X or 10X
   $4 \times t_{24} + 40$ ns for 16X
10. $6 \times t_{24} + 40$ ns for 8X
    $12 \times t_{24} + 40$ ns for 16X
    $7 \times t_{24} + 40$ ns for 10X
    $13 \times t_{24} + 40$ ns for 18X
11. $8 \times t_{24} - 10$ ns for 8X
    $10 \times t_{24} - 10$ ns for 10X
    $16 \times t_{24} - 10$ ns for 16X
    $18 \times t_{24} - 10$ ns for 18X
12. To guarantee recognition on the next clock.
13. 10 ns for high range
    30 ns for low range
14. $4 \times t_{24} - 10$ ns for 8X
    $5 \times t_{24} - 10$ ns for 10X
    $8 \times t_{24} - 10$ ns for 16X
    $9 \times t_{24} - 10$ ns for 18X
15. $t_{29} - 10$ ns
16. See Figure "CRYSTAL CONNECTION".
17. Maximum capacitance load on the $X_2$ pin when an external MOS clock is connected to $X_1$:
    15 pF for DC to 16 MHz
    5 pF for 16 MHz to 32 MHz

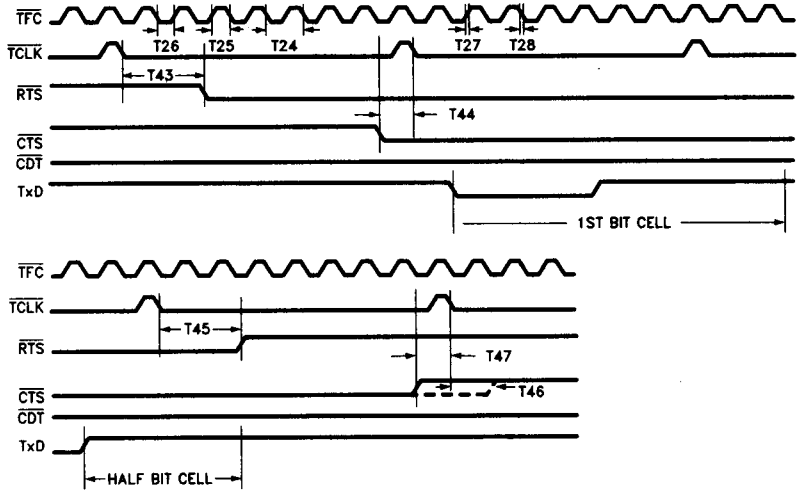
## CRYSTAL CONNECTION




271120-22

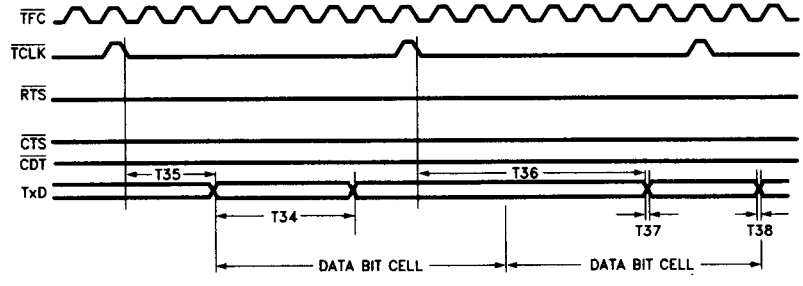
**NOTES:**
1. High-quality, parallel resonant, fundamental-mode crystals are recommended for maximum accuracy.
2. C1, C2, and stray capacitance of the board should be adjusted so the total capacitance load on the crystal is approx. 15 pF.
3. For IEEE 802.3 applications, the crystal must be accurate to ±35 PPM over a range of 0°C to 70°C.

## TRANSMIT TIMINGS—CLOCKS $\overline{RTS}$ AND $\overline{CTS}$



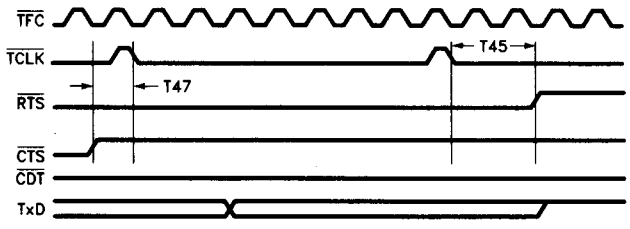

271120-23

## TRANSMIT TIMINGS—MANCHESTER DATA ENCODING

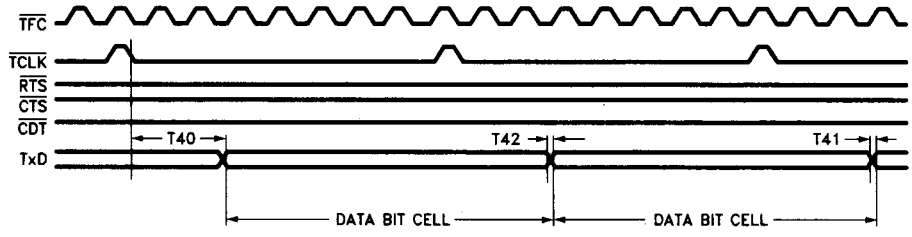



271120-24

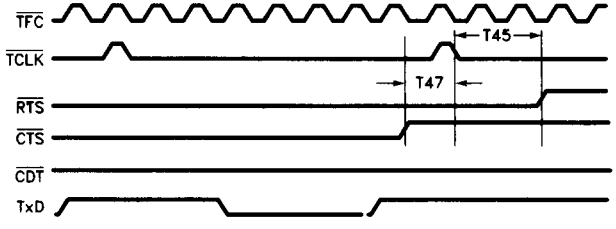
### TRANSMIT TIMINGS—LOST $\overline{CTS}$



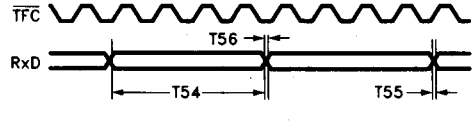

### TRANSMIT TIMINGS—NRZI DATA CODING



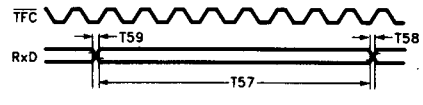

### TRANSMIT TIMINGS—LOST $\overline{CTS}$




### RECEIVE DATA TIMINGS (MANCHESTER)

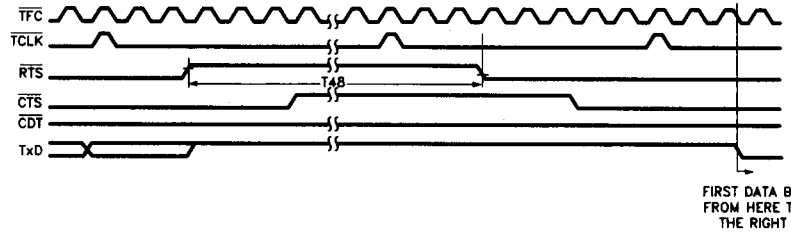


## RECEIVE DATA TIMINGS (NRZI)



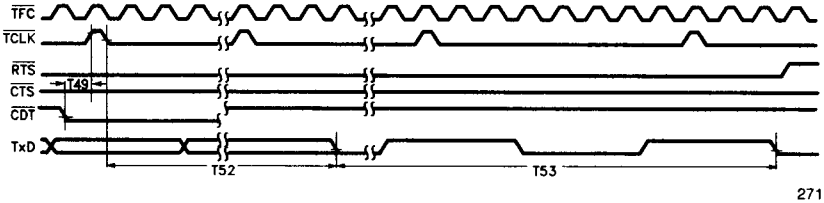

271120-29

## TRANSMIT TIMINGS—INTERFRAME SPACING



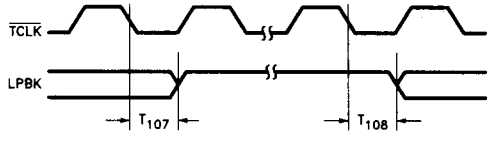

271120-30

## TRANSMIT TIMINGS—COLLISION DETECT AND JAMMING




271120-31

## LOOPBACK OUTPUT SIGNAL TIMINGS




271120-32

## HIGH SPEED MODE

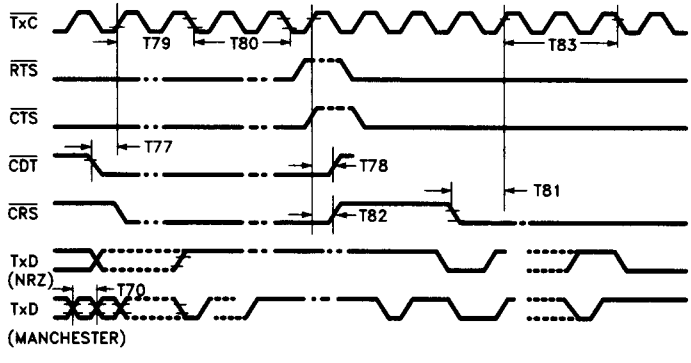
| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **TRANSMIT/RECEIVE CLOCK PARAMETERS** | | | | | |
| $t_{60}$ | $\overline{RxC}$, $\overline{TxC}$ Cycle Time | 50 | | ns | (Notes 1, 3) |
| $t_{61}$ | $\overline{TxC}$ Rise Time | | 5 | ns | (Note 1) |
| $t_{62}$ | $\overline{TxC}$ Fall Time | | 5 | ns | (Note 1) |
| $t_{63}$ | $\overline{TxC}$ High Time | 18 | | ns | (Notes 1, 3) |
| $t_{64}$ | $\overline{TxC}$ Low Time | 19 | | ns | (Notes 1, 3) |
| **TRANSMIT DATA PARAMETERS** | | | | | |
| $t_{65}$ | TxD Rise Time | | 10 | ns | (Note 4) |
| $t_{66}$ | TxD Fall Time | | 10 | ns | (Note 4) |
| $t_{67}$ | $\overline{TxC}$ Low to TxD Valid | | 30 | ns | (Notes 4, 5) |
| $t_{68}$ | $\overline{TxC}$ Low to TxD Transition | | 30 | ns | (Notes 2, 4) |
| $t_{69}$ | $\overline{TxC}$ High to TxD Transition | | 30 | ns | (Notes 2, 4) |
| $t_{70}$ | TxD Transition-Transition | 20 | | ns | (Notes 2, 4) |
| $t_{71}$ | $\overline{TxC}$ Low to TxD High (At the Transmission End) | | 30 | ns | (Note 4) |
| **$\overline{RTS}$, $\overline{CTS}$ PARAMETERS** | | | | | |
| $t_{72}$ | $\overline{TxC}$ Low to $\overline{RTS}$ Low Time to Activate $\overline{RTS}$ | | 30 | ns | (Note 4) |
| $t_{73}$ | $\overline{CTS}$ Low to $\overline{TxC}$ Low $\overline{CTS}$ Setup Time | 20 | | ns | |
| $t_{74}$ | $\overline{TxC}$ Low to $\overline{RTS}$ High | | 30 | ns | (Note 4) |
| $t_{75}$ | $\overline{TxC}$ Low to $\overline{CTS}$ Invalid. $\overline{CTS}$ Hold Time | 10 | | ns | (Note 6) |
| $t_{75a}$ | $\overline{CTS}$ High to $\overline{TxC}$ Low. $\overline{CTS}$ Setup Time to Stop Transmission | 20 | | ns | (Note 6) |
| **INTERFRAME SPACING PARAMETER** | | | | | |
| $t_{76}$ | Inter Frame Delay | (Note 8) | | ns | |

## HIGH SPEED MODE (Continued)

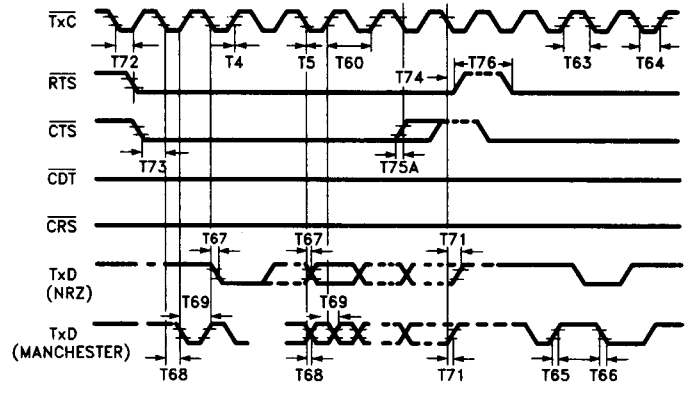
| Symbol | Parameter | Min | Max | Units | Comments |
|---|---|---|---|---|---|
| **$\overline{CRS}$, $\overline{CDT}$ PARAMETERS** | | | | | |
| $t_{77}$ | $\overline{CDT}$ Low to $\overline{TxC}$ High. External Collision Detect Setup Time | 20 | | ns | |
| $t_{78}$ | $\overline{TxC}$ High to $\overline{CDT}$ Inactive. $\overline{CDT}$ Hold Time | 10 | | ns | (Note 12) |
| $t_{79}$ | $\overline{CDT}$ Low to Jam Start | | (Note 9) | ns | |
| $t_{80}$ | Jamming Period | | (Note 10) | ns | |
| $t_{81}$ | $\overline{CRS}$ Low to $\overline{TxC}$ High. Carrier Sense Setup Time | 25 | | ns | |
| $t_{82}$ | $\overline{TxC}$ High to $\overline{CRS}$ Inactive. $\overline{CRS}$ Hold Time | 10 | | ns | (Note 12) |
| $t_{83}$ | $\overline{CRS}$ High to Jamming Start. (Internal Collision Detect) | | (Note 11) | ns | |
| $t_{84}$ | $\overline{CRS}$ High to $\overline{RxC}$ High. $\overline{CRS}$ Inactive Setup Time | 30 | | ns | |
| $t_{85}$ | $\overline{RxC}$ High to $\overline{CRS}$ High. $\overline{CRS}$ Inactive Hold Time | 10 | | ns | |
| **RECEIVE CLOCK PARAMETERS** | | | | | |
| $t_{86}$ | $\overline{RxC}$ Rise Time | | 5 | ns | (Note 1) |
| $t_{87}$ | $\overline{RxC}$ Fall Time | | 5 | ns | (Note 1) |
| $t_{88}$ | $\overline{RxC}$ High Time | 18 | | ns | (Note 1) |
| $t_{89}$ | $\overline{RxC}$ Low Time | 19 | | ns | (Note 1) |
| **RECEIVED DATA PARAMETERS** | | | | | |
| $t_{90}$ | RxD Setup Time | 15 | | ns | (Note 5) |
| $t_{91}$ | RxD Hold Time | 15 | | ns | (Note 5) |
| $t_{92}$ | RxD Rise Time | | 10 | ns | |
| $t_{93}$ | RxD Fall Time | | 10 | ns | |
| **EXTERNAL LOOPBACK PARAMETERS** | | | | | |
| $t_{109}$ | $\overline{TxC}$ Low to LPBK High | | $t_{60}$ | ns | (Note 4) |
| $t_{110}$ | $\overline{TxC}$ Low to LPBK Float | | $t_{60}$ | ns | (Note 4) |

**NOTES:**
1. MOS levels.
2. Manchester only.
3. Manchester. Needs 50% duty cycle.
4. 1 TTL load + 50 pF.
5. NRZ only.
6. Abnormal end of transmission: $\overline{CTS}$ expires before $\overline{RTS}$.
7. Normal end of transmission.
8. Programmable value:
   $t_{76} = N_{IFS} \times t_{60}$
   $N_{IFS}$: the IFS configuration value.
   If $N_{IFS}$ is less than 12 then NIFS is enforced to 12.
9. Programmable value:
   $t_{79} = N_{CDF} \times t_{60} + (12 \text{ to } 15) \times t_{60}$ (If collision occurs after preamble).
   $N_{CDF}$: The collision detect filter configuration value.
10. $t_{80} = 32 \times t_{60}$
11. Programmable value:
    $t_{83} = N_{CDF} \times t_{60} + (12 \text{ to } 15) \times t_{60}$
12. To guarantee recognition on the next clock.

## TRANSMIT DATA WAVEFORMS

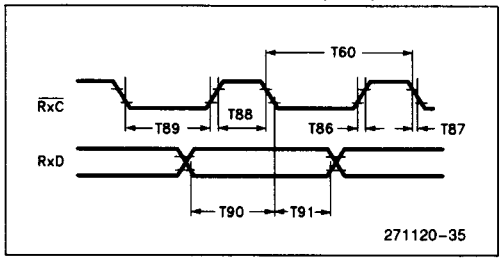



271120-33

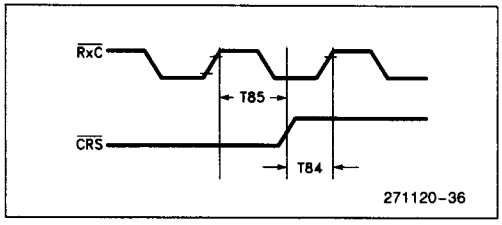



271120-34

### RECEIVE DATA WAVEFORMS (NRZ)




### RECEIVE DATA WAVEFORMS




### LOOPBACK OUTPUT SIGNAL TIMINGS