82588
HIGH INTEGRATION LOCAL AREA NETWORK CONTROLLER

- **Integrates ISO Layers 1 and 2**
  - **— CSMA/CD Medium Access Control (MAC)**
  - **— On-Chip Manchester, NRZI Encoding/Decoding**
  - **— On-Chip Logic Based Collision Detect and Carrier Sense**
- **Supports Mid-Range Industry Standard LANs**
  - **— StarLAN (IEEE 802.3 1BASE5)**
  - **— IBM/PC Network-Baseband and Broadband**
- **High Level Command Interface Offloads the CPU**
- **Efficient Memory Use Via Multiple Buffer Reception**
- **2 Clocks per Data Transfer**
- **User Configurable**
  - **— Up to 2 Mb/s Bit Rates with On-chip Encoder/Decoder (High Integration Mode)**
  - **— Up to 5 Mb/s with External Encoder/ Decoder (High Speed Mode)**
- **No TTL Glue Required with iAPX 186 and 188 Microprocessors**
- **Network Management and Diagnostics**
  - **— Short or Open Circuit Localization**
  - **— Station Diagnostics (External Loopback)**
  - **— Self Test Diagnostics Internal Loopback User Readable Registers**

The 82588 is a highly integrated CSMA/CD controller designed for cost sensitive, mid-range Local Area Network (LAN) applications, such as personal computer networks.

At data rates of up to 2 Mb/s, the 82588 provides a highly integrated interface and performs: CSMA/CD Data Link Control, Manchester, Differential Manchester or NRZI encoding/decoding, clock recovery; Carrier Sense, and Collision Detection. This mode is called "High Integration Mode." In the 82588 "High Speed Mode", the user can transfer data at a rate of up to 5 Mb/s. In this mode the physical link functions are done external to the 82588.

The 82588 is available in a 28 pin DIP and 44 lead PLCC package and fabricated in Intel's reliable HMOS II 5 volt technology.

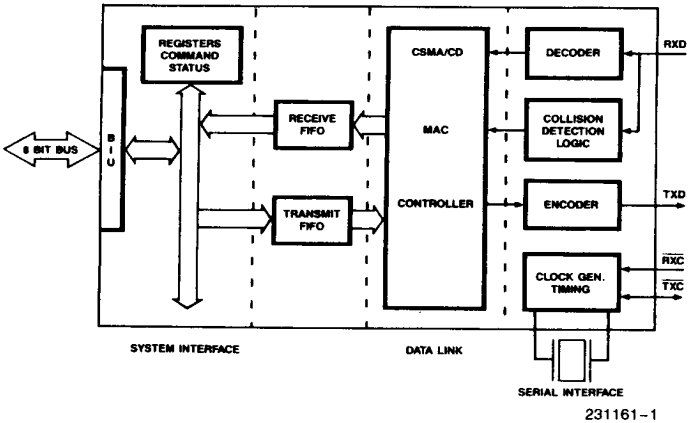


Figure 1. 82588 Block Diagram

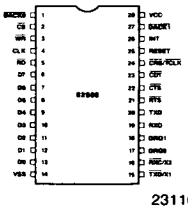


Figure 2. 82588 Pin Configuration (DIP)

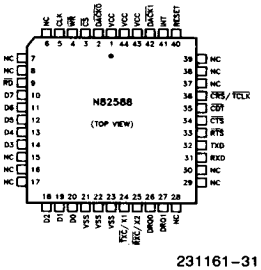


Figure 3. 82588 Pin Configuration (PLCC)

September 1990
Order Number: 231161-005

Table 1. Pin Description

Symbol	Pin No.		Type	Name and Function
	DIP	PLCC		
D7 D6 D5 D4 D3 D2 D1 D0	6 7 8 9 10 11 12 13	10 11 12 13 14 18 19 20	I/O	**DATA BUS:** The Data Bus lines are bi-directional three state lines connected to the system's Data Bus for the transfer of data, commands, status and parameters.
$\overline{RD}$	5	9	I	**READ:** Together with $\overline{CS}$, $\overline{DACK0}$ or $\overline{DACK1}$, Read controls data or status transfers out of the 82588 registers.
$\overline{WR}$	3	4	I	**WRITE:** Together with $\overline{CS}$, $\overline{DACK0}$ or $\overline{DACK1}$, Write controls data or command transfers into the 82588 registers.
$\overline{CS}$	2	3	I	**CHIP SELECT:** When this signal is LOW, the 82588 is selected by the CPU for transfer of command or status. The direction of data flow is determined by the $\overline{RD}$ or $\overline{WR}$ inputs.
CLK	4	5	I	**CLOCK:** System clock. TTL compatible signal.
RESET	25	40	I	**RESET:** A HIGH signal on this pin will cause the 82588 to terminate current activity. This signal is internally synchronized and must be held HIGH for at least four Clock cycles.
INT	26	41	O	**INTERRUPT:** Active HIGH signal indicates to the CPU that the 82588 is requesting an interrupt.
DRQ0	17	26	O	**DMA REQUEST (CHANNEL 0):** This pin is used by the 82588 to request a DMA transfer. DRQ0 remains HIGH as long as 82588 requires data transfers. Burst transfers are done by having the signal active for multiple transfers.
DRQ1	18	27	O	**DMA REQUEST (CHANNEL 1):** This pin is used by the 82588 to request a DMA transfer. DRQ1 remains HIGH as long as 82588 requires data transfers. Burst transfers are done by having the signal active or multiple transfers.
$\overline{DACK0}$	1	2	I	**DMA ACKNOWLEDGE (CHANNEL 0):** When LOW, this input signal from the DMA Controller notifies the 82588 that the requested DMA cycle is in progress. This signal acts like chip select for data and parameter transfer using DMA channel 0.
$\overline{DACK1}$	27	42	I	**DMA ACKNOWLEDGE (CHANNEL 1):** When LOW, this input signal from the DMA controller notifies the 82588 that the requested DMA cycle is in progress. This signal acts like chip select for data and parameter transfer using DMA channel 1.

Table 1. Pin Description (Continued)

Symbol	Pin No.		Type	Name and Function
	DIP	PLCC		
				High Integration Mode
X1/X2	15/16	24/25	I	**OSCILLATOR INPUTS:** These inputs may be used to connect a quartz crystal that controls the internal clock generator for the serial unit. X1 may also be driven by a MOS level clock whose frequency is 8 or 16 times the bit rate of Transmit/Receive data. X2 must be left floating if X1 has an external MOS clock.
				High Speed Mode
$\overline{TxC}$	15	24	I	**TRANSMIT CLOCK:** This signal provides timing information to the internal serial logic, depending upon the mode of data transfer. 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 $\overline{TxC}$ LOW to HIGH transition.
$\overline{RxC}$	16	25	I	**RECEIVE CLOCK:** This signal provides timing information to the internal serial logic. NRZ data should be provided for reception (RxD). The state of the RxD pin is sampled on the HIGH to LOW transition of $\overline{RxC}$. The operating mode of the 82588 is defined when configuring the chip.
$\overline{TCLK}/\overline{CRS}$	24	36	I	In High Speed Mode, this pin is Carrier Sense, input CRS, and is used to notify the 82588 that there is activity on the serial link.
			O	In High Integration Mode, this pin is Transmit Clock, $\overline{TCLK}$, and is used to output the transmit clock.
$\overline{CDT}$	23	35	I	**COLLISION DETECT:** This input notifies the 82588 that a collision has occurred. It is sensed only if the 82588 is configured for external Collision Detect (external circuitry is then required for detecting the collision).
RxD	19	31	I	**RECEIVE DATA:** This pin receives serial data.
TxD	20	32	O	**TRANSMIT DATA:** This pin transmits data to the Serial Link. This signal is HIGH when not transmitting.
$\overline{RTS}$	21	33	O	**REQUEST TO SEND:** When this signal is LOW, the 82588 notifies an external interface that it has data to transmit. It is forced HIGH after a reset and when transmission is stopped.
$\overline{CTS}$	22	34	I	**CLEAR TO SEND:** CTS enables the 82588 to start transmitting data. Raising this signal to HIGH stops the transmission.
VCC	28	1, 43, 44		**POWER:** + 5V Supply
VSS	14	21, 22, 23		Ground

Table 1. Pin Description (Continued)

Symbol	Pin No.		Type	Name and Function
	DIP	PLCC		
NC		6 7 8 15 16 17 28 29 30 37 38 39		**NO CONNECT:** These pins are reserved for future use.

FUNCTIONAL DESCRIPTION

High Integration Mode

The 82588 LAN Controller is a highly integrated CSMA/CD controller for cost sensitive LAN applications such as personal computer networks. Included on chip is a programmable CSMA/CD controller, an NRZI and Manchester encoder/decoder with clock recovery, and two collision detection mechanisms. With the addition of simple transceiver line drivers or RF Modem, the 82588 performs all the major functions of the ISO Physical and Data Link Layers.

CSMA/CD Controller

The 82588 on-chip CSMA/CD controller is programmable, which allows it to operate in a variety of LAN environments, including industry standards such as StarLAN (IEEE 802.3 1BASE5) and the 2 Mb/s IBM PC Network (both baseband and broadband transmission). Programmable parameters include:

- Framing (End of Carrier of SDLC)
- Address field length
- Station priority
- Interframe spacing
- Slot time
- CRC-32 OR CRC-16

Encoder/Decoder

The on-chip NRZI and Manchester encoder/decoder supports data rates up to 2 Mb/s. Manchester encoding is typically used in baseband applications and NRZI is used in broadband applications.

Collision Detection

One of the 82588's unique features is its on-chip logic based collision detection. To ensure a high probability of collision detection two mechanisms are provided. The Code Violation method defines a collision when a transition edge occurs outside the area of normal transitions as specified by either the Manchester or NRZI encoding methods. Bit Comparison method compares the signature of the transmitted frame to the received frame signature (re-calculated by the 82588 while listening to itself). If the signatures are identical the frame is assumed to have been transmitted without a collision.

System Interface

In addition to providing the functions necessary for interfacing to the LAN, the 82588 has a friendly system interface that eases the design effort. First, the 82588 has a high level command interface; that is the CPU sends the 82588 commands such as Transmit or Configure. This means the designer does not have to write low level software to perform these tasks, and it offloads the CPU in the application. Second, the 82588 supports an efficient memory structure called Multiple Buffer Reception in which buffers are chained together while receiving frames. This is an important feature in applications with limited memory, such as personal computers. Third, the 82588 has two independent sixteen byte FIFO's, one for reception and one for transmission. The FIFO's allow the 82588 to tolerate bus latency. Finally the 82588 provides an eight byte data path that supports up to 4 Mbytes/second using external DMA.

Network Management & Diagnostics

The 82588 provides a rich set of diagnostic and network management functions including: internal and external loopback, channel activity indicators, optional capture of all frames regardless of destination address (Promiscuous Mode), capture of collided frames, (if address matches), and time domain reflectometry for locating fault points in the network cable. The 82588 register Dump command ensures reliable software by dumping the content of the 82588 registers into the system memory.

The next section will describe the 82588 system bus interface, the 82588 network interface, and the 82588 internal architecture.

82588/Host CPU Interaction

The CPU communicates with the 82588 through the system's memory and 82588's on-chip registers. The CPU creates a data structure in the memory, programs the external DMA controller with the start address and byte count of the block, and issues the command to the 82588.

The 82588 is optimized for operating with the iAPX 186/188, but due to the small number of hardware signals between the 82588 and the CPU, the 82588 can operate easily with other processors. The data bus is 8 bits wide and there is no address bus.

Chip Select and Interrupt lines are used to communicate between the 82588 and the host as shown in the Figure 3. Interrupt is used by the 82588 to draw the CPU's attention. The Chip Select is used by the CPU to draw the 82588's attention.

There are two kinds of transfer over the bus: Command/Status and data transfers. Command/Status transfers are always performed by the CPU. Data transfers are requested by the 82588, and are typically performed by a DMA controller. The table given in Figure 4 shows the Command/Status and data transfer control signals.

The CPU writes to 82588 using $\overline{CS}$ and $\overline{WR}$ signals. The CPU reads the 82588 status register using $\overline{CS}$ and $\overline{RD}$ signals.

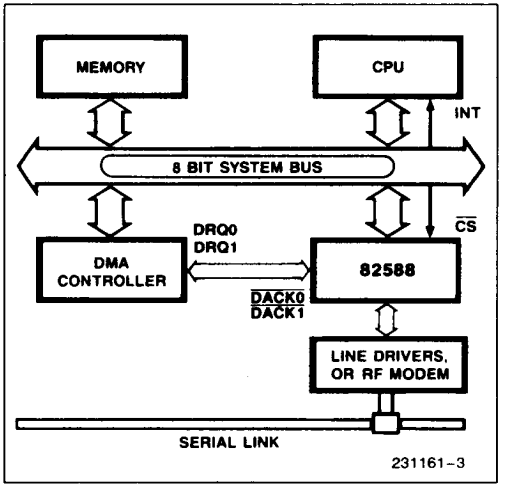


Figure 3. 82588/HOST CPU Interaction

To initiate an operation like Transmit or Configure (see Figure 5), a Write command from CPU to 82588 is issued by the CPU. A Read operation from CPU gives the status of the 82588. Although there are four status registers they're read using the same port in a round robin fashion (Figure 6).

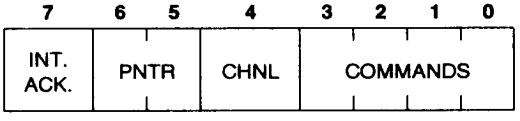
Any parameters or data associated with a command are transferred between the memory and 82588 using DMA. The 82588 has two data channels, each having Request and Acknowledge lines. Typically one channel is used to receive data and other to transmit data and perform all the other initialization and maintenance operations like Configure, Address Set-Up, Diagnose, etc. The channels are identical and can be used interchangeably.

When the 82588 requires access to the memory for parameter or data transfer it activates the DMA request lines and uses the DMA controller to achieve the data transfer. Upon the completion of an operation, the 82588 interrupts the CPU. The CPU then reads results of the operation (the status of the 82588).

Pin Name			Function
$\overline{CS}$*	$\overline{RD}$	$\overline{WR}$	
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 $\overline{CS}$, $\overline{DACK0}$ and $\overline{DACK1}$ may be active at any time.

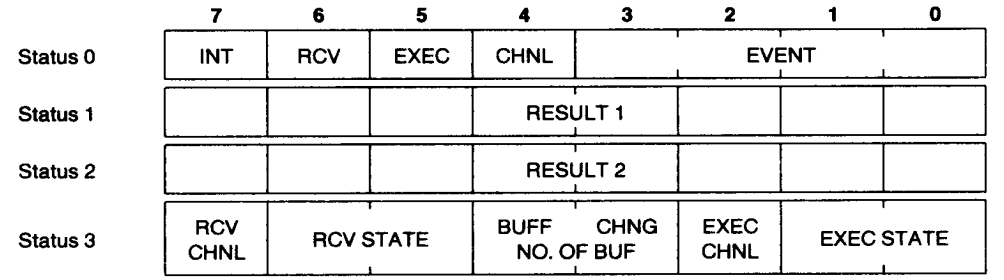
Figure 4. Databus Control Signals and Their Functions



COMMAND REGISTER

COMMANDS		VALUE	COMMANDS		VALUE
NOP	—	0	ABORT	—	13
IA-SETUP	—	1	RECEIVER-ENABLE	—	8
CONFIGURE	—	2	ASSIGN NEXT BUF	—	9
MC-SETUP	—	3	RECEIVE-DISABLE	—	10
TRANSMIT	—	4	STOP-RECEPTION	—	11
TDR	—	5	RESET	—	14
DUMP	—	6	FIX PTR	—	15 (CHNL = 1)
DIAGNOSE	—	7	RLS PTR	—	15 (CHNL = 0)
RETRANSMIT	—	12			

Figure 5. Command Format and Operation Values



EVENTS		VALUE (STATUS 0)
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

Figure 6. Status Registers and Event Values

Transmitting a Frame

To transmit a frame, the CPU prepares a Transmit Data Block in memory as shown in Figure 7. Its first two bytes specify the length of the rest of the block. The next few bytes (Up to 6 bytes long) contain the destination address of the node it 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 the Transmit command to the 82588.

Upon receiving the command, the 82588 fetches the first two bytes of the block to determine the length of the block. If the link is free, and the first data byte was fetched, the 82588 begins transmitting the preamble and concurrently fetches the bytes from the Transmit Data Block and loads them into a 16 byte FIFO to keep them ready for transmitting. The FIFO is a buffer between the serial and parallel part of the 82588. The on-chip FIFOs help the 82588 to tolerate system bus latency as well as provide efficient usage of system bandwidth.

The destination address is sent out after the preamble. This is followed by the source or the station individual address, which is stored earlier on the 82588 using the IA-SETUP command. After that, the entire information field is transmitted followed by a CRC field calculated by the 82588. If during the transmission of the frame, a collision is encountered, then the transmission is aborted and a jam pattern is sent out after completion of the preamble. The 82588 generates an Interrupt indicating the experience of a collision and the frame has to be re-transmitted. Retransmission is done by the CPU exactly as the Transmit command except the Re-Transmit command keeps track of the number of collisions encountered. When the 82588 gets the Retransmit command and the exponential back-off time is expired, the 82588 transmits the frame again. The transmitted frame can be coded to either Manchester, Differential Manchester or NRZI methods.

Collision Detection

The 82588 eliminates the need for external collision detection logic, in most applications, while easing or eliminating the need for complex transceivers. Two algorithms are used for collision detection: Bit Comparison and Code Violation. The Bit Comparison Method is useful in Broadband networks where there are separate transmit and receive channels. Bit Comparison compares the "signature" of the transmitted data and received data at the end of the collision window in any network configuration. This algorithm calculates the CRC after a programmable number of transmitted bits, holds this CRC in a register, and compares it with received data's CRC. A CRC or "signature" difference indicates a collision. The code violation is detected if the encoding of the received data has any bit that does not fit the encoding rules. The code violation method is useful in short bus topology and serial backplane applications where bit attenuation over the bus is negligible.

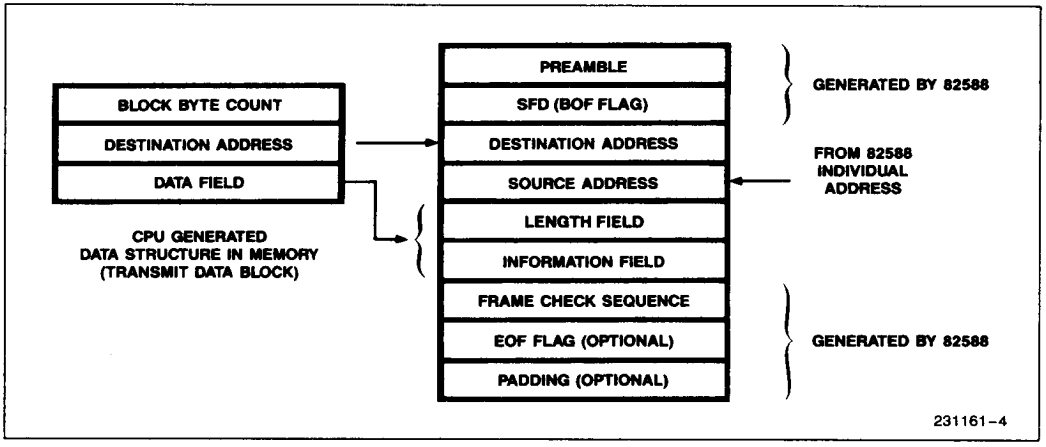


Figure 7. The 82588 Frame Structure and location of Data element in System Memory

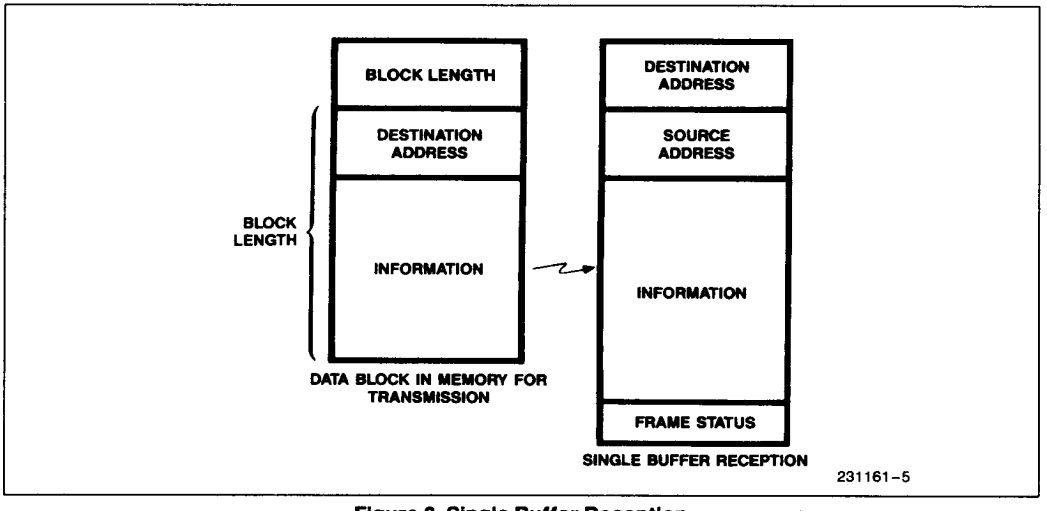


Figure 8. Single Buffer Reception

Receiving a Frame

The 82588 can receive a frame when its receiver has been enabled. The received frame is decoded by either on-chip Manchester, Differential Manchester or NRZI decoders in High Integration Mode and NRZI in High Speed Mode. The 82588 checks for an address match for an individual address, a Mulitcast address or a Broadcast address. In the Promiscuous mode the 82588 receives all frames. Only when the address match is successful does the 82588 transfer the frame to the memory using the DMA controller. Before enabling the receiver, the CPU makes a memory buffer area available to the Receive Unit and programs the starting address of the DMA controller. The received frame is transferred to the memory buffer in the format shown in Figure 8. This method of reception is called "Single Buffer" reception. The entire frame is contained in one continuous buffer. Upon completion of reception the total number of bytes written into the memory buffer is loaded into status registers 1 and 2 and the status of the reception itself is appended to the received frame. An interrupt to the CPU follows.

If the frame size is unknown, memory usage can be optimized by using "Multiple Buffer" reception.

This way the user does not have to allocate large memory space for short frames. Instead, the 82588 can dynamically allocate memory space as it receives frames. This method requires both DMA channels alternately to receive the frame. As the frame reception starts, the 82588 interrupts the CPU and automatically requests assignment of the next sequential buffer. The CPU does this and loads the second DMA channel with the next buffer information so that the 82588 can immediately switch to the other channel as soon as the current buffer is full. When the 82588 switches from the first to the second buffer it again interrupts the CPU requesting it to allocate another buffer on the other (previous) channel in advance. 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 (see Figure 9).

This dynamic (pre) allocation of memory buffers results in efficient use of available storage when handling frames of widely differing sizes. Since the buffers are pre-allocated one block in advance, the system is not time critical.

80188 Based System

Figure 10 shows a high performance, high-integration configuration of the 82588 with the 80188 in a typical iAPX188-based microcomputer. The 80188 controls the 82588, as well as providing DMA control services for data transfer, using its on-chip two channel DMA controller.

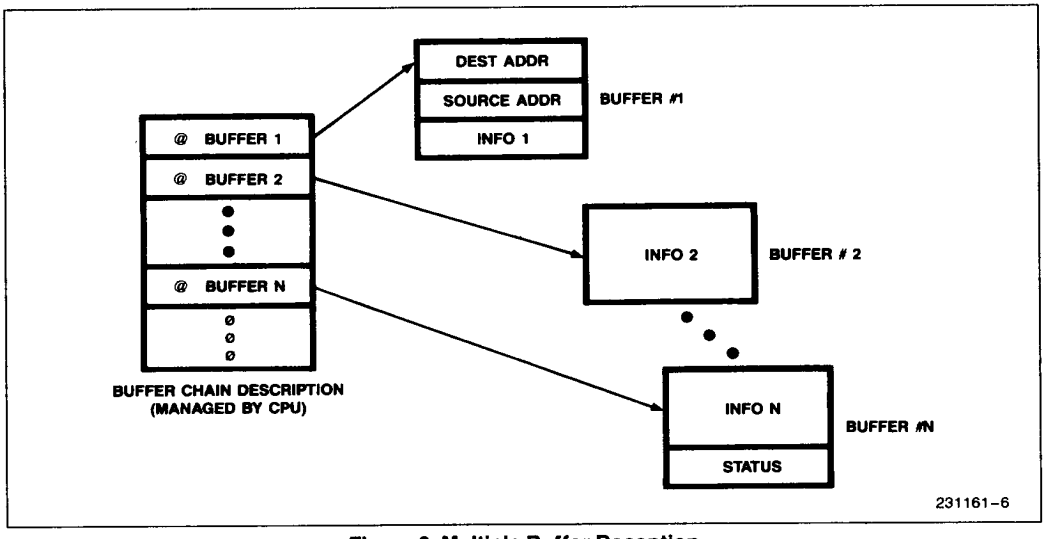


Figure 9. Multiple Buffer Reception

Link Interface

The Serial Interface Mode configuration parameter selects either a highly integrated Direct Link interface (High Integration Mode) or a highly flexible Transceiver Interface (High Speed Mode).

Application

In the High Integration Mode it is possible to connect the 82588 on a very short "Wired OR" link, on a longer twisted pair cable, or a broadband connection.

Twisted Pair Connection

The link consists of a twisted pair that interconnects the 82588. The transmit data pin is connected via a driver and the receive data pin is connected via a buffer. The twisted pair must be properly terminated to prevent reflections.

In the minimum configuration, TxD and RxD are connected to the twisted pair and $\overline{CTS}$ is grounded. The 82588 may control the driver with the $\overline{RTS}$ pin. It is also possible to use external circuitry for performing collision detection, and feeding it to the 82588 through the $\overline{CDT}$ pin.

Broadband Connection

The 82588 supports data communications over a broadband link in both its modes. Proper MODEM interface should be provided. Collision Detection by Bit Comparison, in High Interface Mode, can be applied to transmission over broadband links.

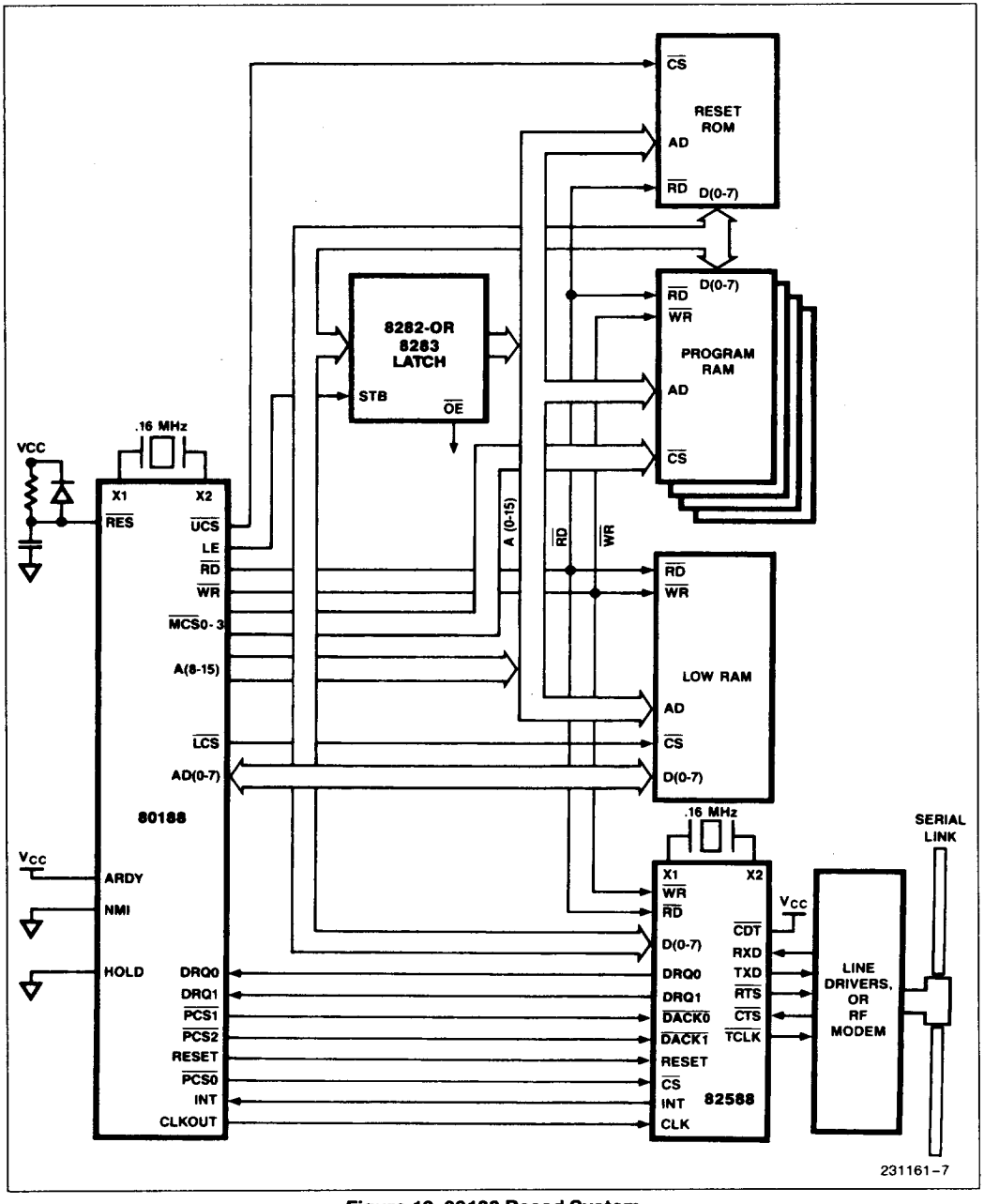


Figure 10. 80188 Based System

Absolute Maximum Ratings*

Ambient Temperature Under Bias ....0°C to +70°C
Storage Temperature .....−65°C to +150°C
Voltage on Any Pin With
Respect to Ground .....−1.0V to 7V
Power Dissipation .....1.7 Watts

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

D.C. Characteristics

(T_A = 0°C to +70°C; T_C (DIP) = 52°C to 108°C, T_C (PLCC) = 63°C to 116°C; VCC = +5V ±10%)

1

$\overline{TxC}$, $\overline{RxC}$ have MOS levels (See VMIL, VMIH). All other signals have TTL levels (See VIL, VIH, VOL, VOH).

Symbol	Parameter	Min	Max	Units	Test Conditions
VIL	Input Low Voltage (TTL)	−0.5	+0.8	V	
VIH	Input High Voltage (TTL)	2.0	VCC + 0.5	V	
VOL	Output Low Voltage (TTL)		0.45	V	IOL = 2.0 mA
VOH	Output High Voltage (TTL)	2.4		V	IOH = −400 μA
VMIL	Input Low Voltage (MOS)	−0.5	0.6	V	
VMIH	Input High Volatge (MOS)	3.9	VCC + 0.5	V	
ILI	Input Leakage Current		+10	μA	0 = VIN = VCC
ILO	Output Leakage Current		±10	μA	0.45 = VOUT = VCC
ICC	Power Supply Current		400 300	mA mA	T_A = 0°C T_A = +70°C

A.C. Characteristics

(T_A = 0°C to +70°C; T_C (DIP) = 52°C to 108°C, T_C (PLCC) = 63°C to 116°C; VCC = +5V ±10%)

System Clock Parameters

Symbol	Parameter	Min	Max	Units	Test Conditions
T1	CLK Cycle Period	125		ns	
T2	CLK Low Time	53	1000	ns	*5
T3	CLK High Time	53		ns	*6
T4	CLK Rise Time		15	ns	*1
T5	CLK Fall Time		15	ns	*2

A.C. Characteristics (Continued)

Symbol	Parameter	Min	Max	Units	Test Conditions
Reset Parameters					
T6	Reset Active to Clock Low	20		ns	*3
T8	Reset Pulse Width	4T1		ns	
T9	Control Inactve After Reset		T1	ns	
Interrupt Timing Parameters					
T10	CLK High to Interrupt Active		85	ns	*4
T11	$\overline{WR}$ Idle to Interrupt Idle		85	ns	*4
Write Parameters					
T12	$\overline{CS}$ or $\overline{DACK0}$ or $\overline{DACK1}$ Setup to $\overline{WR}$ Low	0		ns	
T13	$\overline{WR}$ Pulse Width	95		ns	
T14	$\overline{CS}$ or $\overline{DACK0}$ or $\overline{DACK1}$ Hold After $\overline{WR}$ High	0		ns	
T15	Data Setup to $\overline{WR}$ High	75		ns	
T16	Data Hold After $\overline{WR}$ High	0		ns	
Read Parameters					
T17	$\overline{CS}$ or $\overline{DACK0}$ or $\overline{DACK1}$ Setup to $\overline{RD}$ Low	0		ns	
T18	$\overline{RD}$ Pulse Width	95		ns	
T19	$\overline{CS}$ or $\overline{DACK0}$ or $\overline{DACK1}$ Address Valid After $\overline{RD}$ High	0		ns	
T20	$\overline{RD}$ Low to Data Valid		80	ns	*7
T21	Data Float After $\overline{RD}$ High		55	ns	*7
DMA Parameters					
T22	CLK Low to DRQ0 or DRQ1 Active		85	ns	*4
T23	$\overline{WR}$ or $\overline{RD}$ Low to DRQ0 or DRQ1 Inactive		60	ns	*4

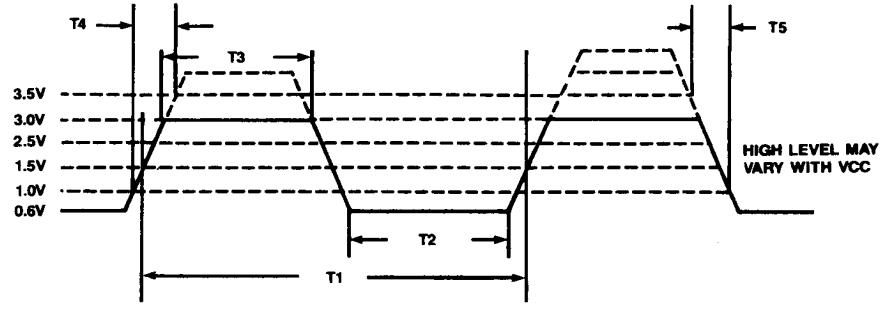
NOTES:
*1—0.8V–2.0V
*2—2.0V–0.8V
*3—to guarantee recognition at next clock
*4—CL = 50 pF
*5—measured at 1.5V
*6—measured at 1.5V
*7—CL = 20 pF–200 pF

A.C. TESTING INPUT/OUTPUT WAVEFORM

231161-8

AC Testing Inputs are Driven at 2.4V for a Logic 1 and 0.45V for a Logic Q Timing Measurements are Made at 1.5V for Both a Logic 1 and Q:
Rise and Fall Time of Input/Output Signals are Measured Between 0.8V to 2.0V Respectively.

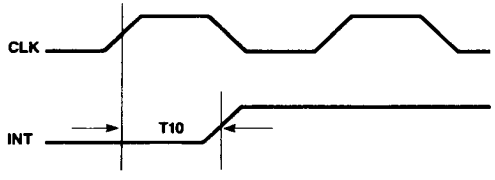
TTL Input/Output Voltage Levels for Timing Measurements



231161-9

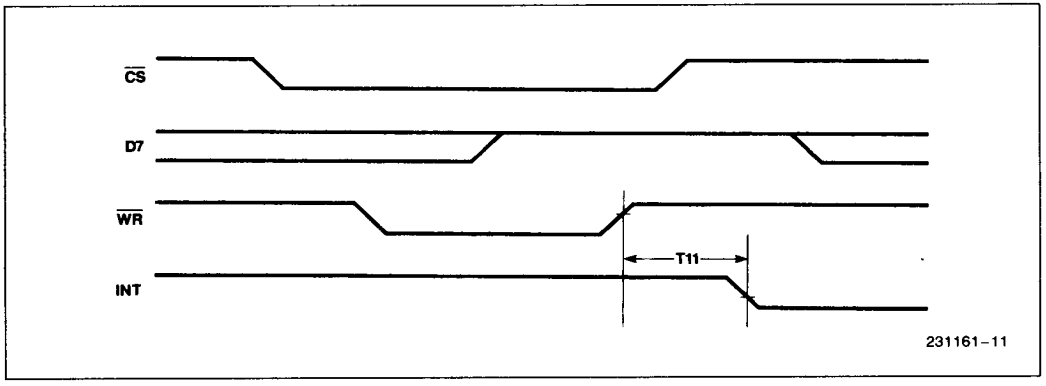
Rise and Fall Time of Input Signals are Measured Between 1.0V to 3.5V Respectively.

Clocks MOS Input Voltage Levels for Timing Measurements

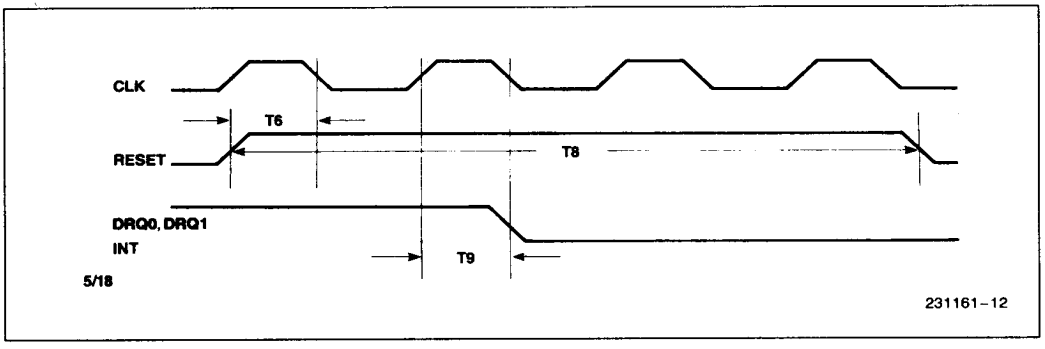


231161-10

Interrupt Timing (Going Active)



Interrupt Timing (Going Inactive)



Reset Timing

Serial Interface A.C. Timing Characteristics
High Integration Mode

$\overline{TFC}$ is the crystal or serial clock input at the X1 pin. When a serial clock is provided at the X1 pin, the maximum capacitive load allowed on the X2 pin is 15 pF.
$\overline{TFC}$ Frequency Range:

For Oscillator Frequency = 1 to 16 MHz (High)		
	×8 Sampling	×16 Sampling
$\overline{TCLK}$ Frequency	0.125 – 2 MHz	62.5 kHz – 1 MHz
T29 = $\overline{TCLK}$ Cycle Time	8 × T24	16 × T24
T30 = $\overline{TCLK}$ High Time	T24 (Typically)	T24 (Typically)
T31 = $\overline{TCLK}$ Low time	7 × T24 (Typically)	15 × T24 (Typically)

For Oscillator Frequency = 0 to 1 MHz (Low)*		
	×8 Sampling	×16 Sampling
$\overline{TCLK}$ Frequency	0 – 0.125 MHz	0 – 6.25 kHz
T29 = $\overline{TCLK}$ Cycle Time	8 × T24	16 × T24
T30 = $\overline{TCLK}$ High Time	T25 (Typically)	T25 (Typically)
T31 = $\overline{TCLK}$ Low Time	7 × T24 + T26 (Typically)	15 × T24 + T26 (Typically)

*A non-symmetrical clock should be provided so that T25 is less than 1000 ns.
T24 = Serial Clock Period
T25 = Serial Clock High Time
T26 = Serial Clock Low Time

High Speed Mode

- Applies for $\overline{TxC}$, $\overline{RxC}$
- f max = 5 MHz ±100 ppm
- For Manchester, symmetry is required: T_{63}, $T_{64} = \frac{1}{2f} \pm 5\%$

High Integration Mode

Symbol	Parameter	Min	Max	Units	Test Conditions
External (Fast) Clock Parameters					
T24	Fast Clock Cycle	62.5		ns	*1
T25	$\overline{TFC}$ High Time	18.5	1000	ns	*1, *14
T26	$\overline{TFC}$ Low Time	23.5		ns	*1
T27	$\overline{TFC}$ Rise Time		5	ns	*1
T28	$\overline{TFC}$ Fall Time		5	ns	*1
Transmit Clock Parameters					
T29	Transmit Clock Cycle	500		ns	*3, *12
T30	$\overline{TCLK}$ High Time	*8	1070	ns	*3
T31	$\overline{TCLK}$ Low Time	*9			*3
T32	$\overline{TCLK}$ Rise Time		15	ns	*3
T33	$\overline{TCLK}$ Fall Time		15	ns	*3

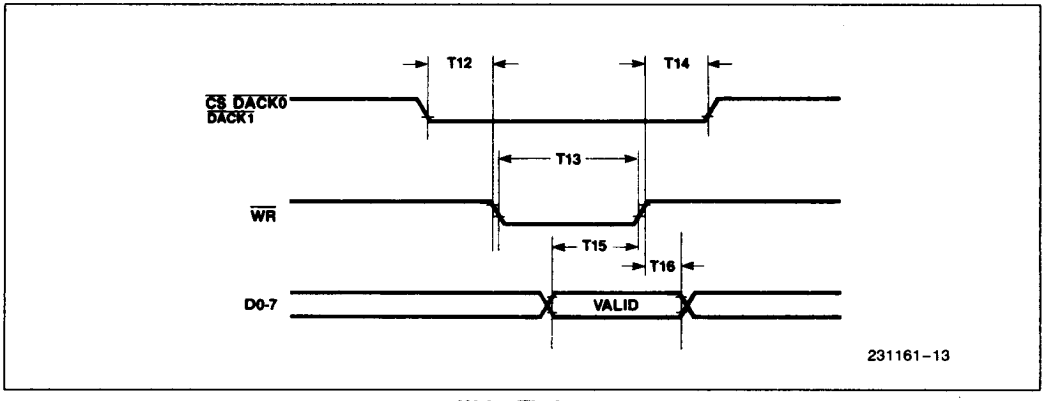
High Integration Mode (Continued)

Symbol	Parameter	Min	Max	Units	Test Conditions
Transmit Data Parameters (Manchester, Differential Manchester)					
T34	TxD Transition-Transition	4T24-10		ns	*12
T35	$\overline{TCLK}$ Low to TxD Transition Half Bit Cell		*10		*2, *12
T36	$\overline{TCLK}$ Low to TxD Transition Full Bit Cell		*11		*2, *12
T37	TxD Rise Time		15	ns	*2
T38	TxD Fall Time		15	ns	*2
Transmit Data Parameters (NRZI)					
T39	TxD Transition-Transition	8T24-10		ns	*12
T40	$\overline{TCLK}$ Low to TxD Transition		*10		*2, *12
T41	$\overline{TxD}$ Rise Time		15	ns	*2
T42	TxD Fall Time		15	ns	*2
$\overline{RTS}$, $\overline{CTS}$, Parameters					
T43	$\overline{TCLK}$ Low To $\overline{RTS}$ Low		*10		*3, *12
T44	$\overline{CTS}$ Low to $\overline{TCLK}$ Low $\overline{CTS}$ Setup Time	65		ns	
T45	$\overline{TCLK}$ low to $\overline{RTS}$ High		*10		*3, *12
T46	$\overline{TCLK}$ Low to $\overline{CTS}$ Invalid. $\overline{CTS}$ Hold Time	20		ns	*4, *13
T47	$\overline{CTS}$ High to $\overline{TCLK}$ Low. $\overline{CTS}$ Setup Time to Stop Transmission	65		ns	*4
IFS Parameters					
T48	Interframe Delay	*5			
Collision Detect Parameter					
T49	$\overline{CDT}$ Low to $\overline{TCLK}$ High. External Collision Detect Setup Time	50		ns	*13
T50	$\overline{CDT}$ High to $\overline{TCLK}$ Low	50		ns	*13
T51	$\overline{TCLK}$ High to $\overline{CDT}$ Inactive. $\overline{CDT}$ Hold Time	20		ns	*13

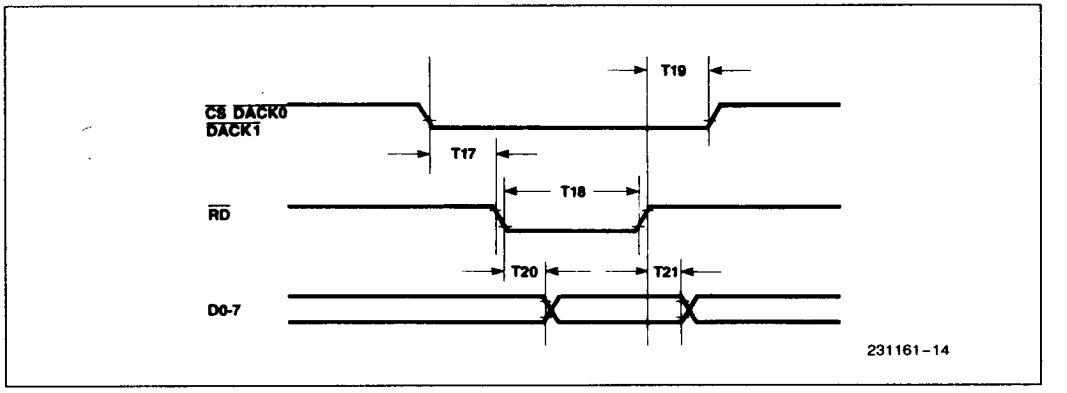
High Integration Mode (Continued)

Symbol	Parameter	Min	Max	Units	Test Conditions
Collision Detect Parameters (Continued)					
T52	$\overline{CDT}$ Low to Jamming Start		*6		
T53	Jamming Period	*7			
Received Data Parameters (Manchester)					
T54	RxD Transition-Transition	4T24		ns	*12
Received Data Parameters (Manchester)					
T55	RxD Rise Time		10	ns	*1
T56	RxD Fall Time		10	ns	*1
Received Data Parameters (NRZI)					
T57	RxD Transition-Transition	8T24		ns	*12
T58	RxD Rise Time		10	ns	*1
T59	RxD Fall Time		10	ns	*1

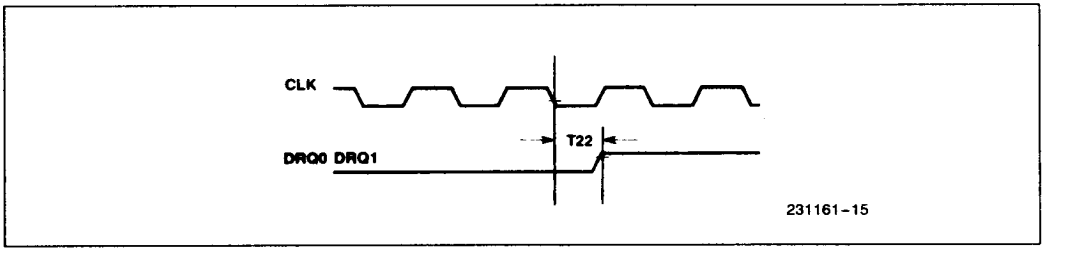
NOTES:
*1—MOS levels.
*2—1 TTL load + 50 pF.
*3—1 TTL load + 100 pF.
*4—Abnormal end to transmission: $\overline{CTS}$ expires before $\overline{RTS}$.
*5—Programmable value: T48 = NIFS × T29 (ns) NIFS—the IFS configuration value.
If NIFS is less than 12, then it is enforced to 12.
*6—Programmable value:
T52 = NCDF × T29 + (12 to 15) × T29 (if collision occurs after preamble).
*7—T53 = 32 × T29
*8—Depends on T24 frequency range:
High Range: T24 − 10
Low Range: T25 − 10
*9—T31 = T29 − T30 − T32 −T33
*10—2T24 + 40 ns
*11—6T24 + 40 ns
*12—For ×16 sampling clock parameter minimum value should be multiplied by a factor of 2.
*13—To guarantee recognition on the next clock.
*14—62.5 ns minimum in Low Range.



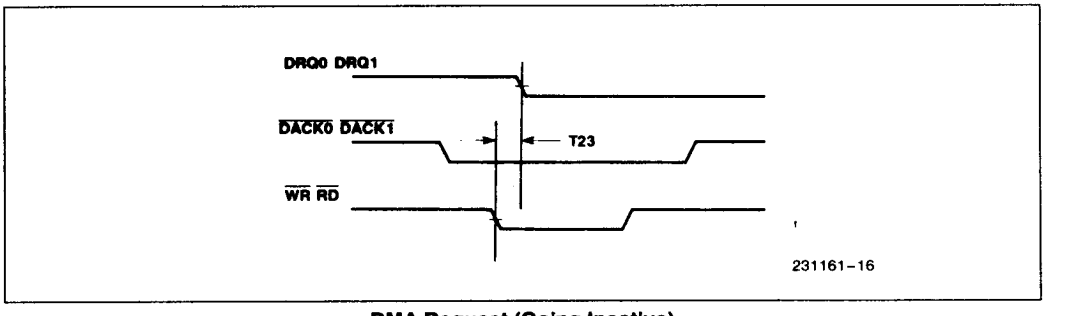
Write Timing



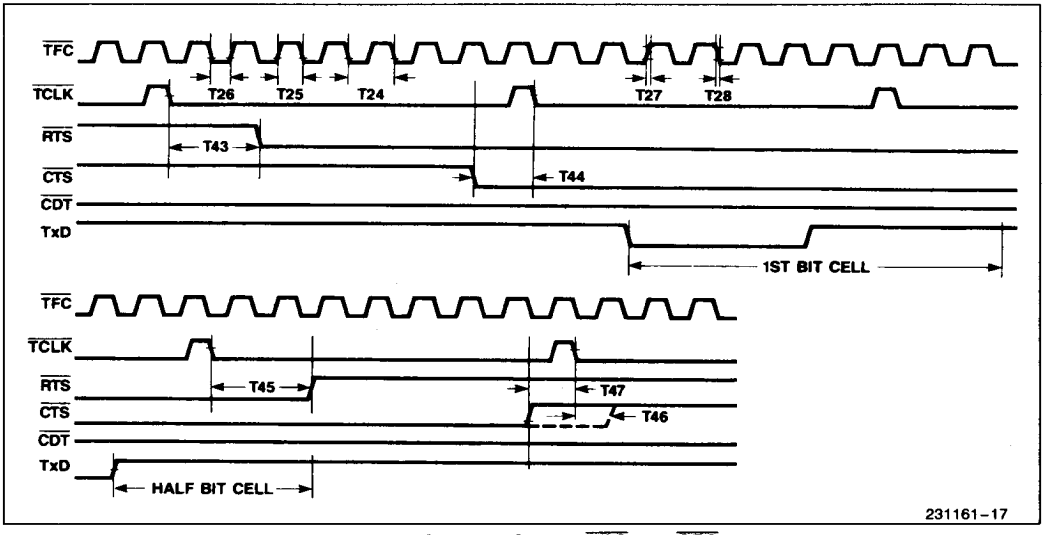
Read Timing



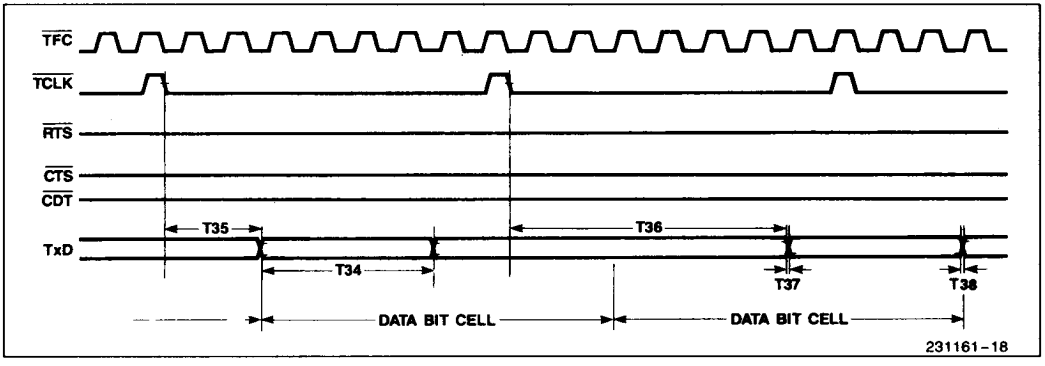
DMA Request (Going Active)



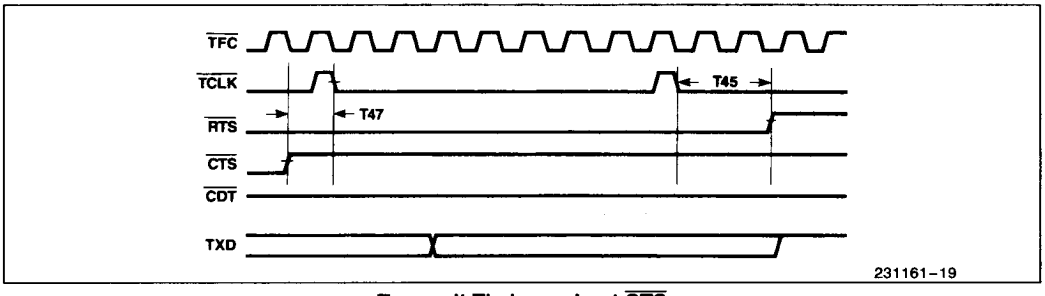
DMA Request (Going Inactive)



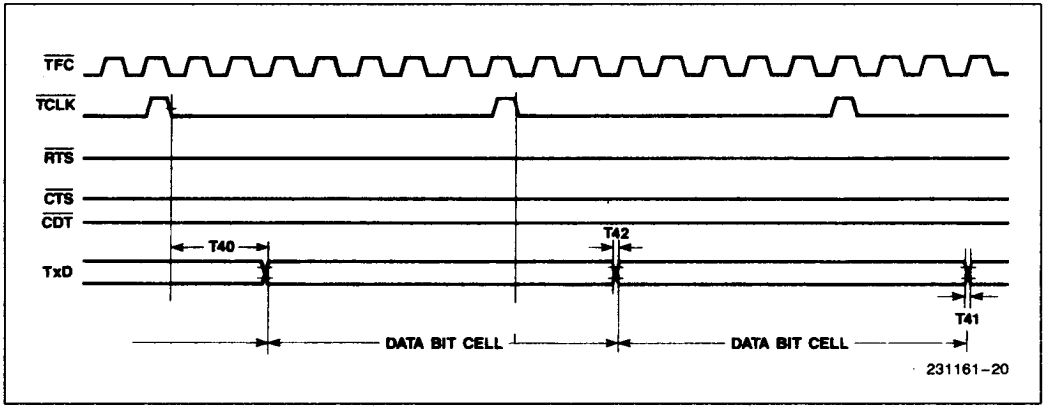
Transmit Timings: Clocks $\overline{RTS}$ and $\overline{CTS}$



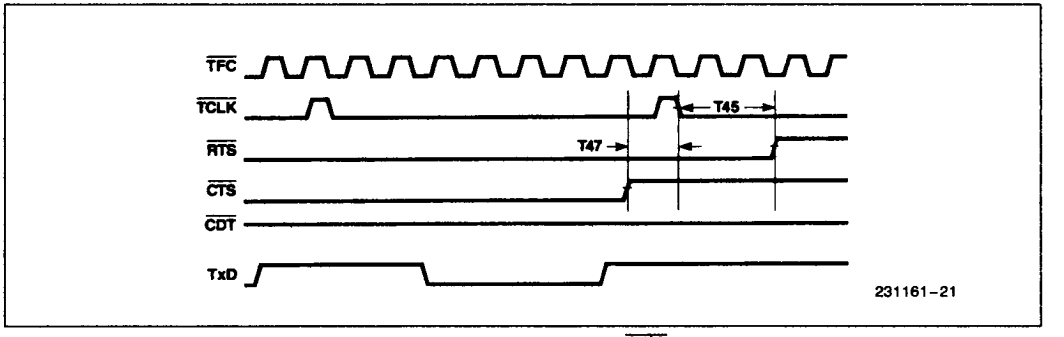
Transmit Timings—Manchester Data Encoding



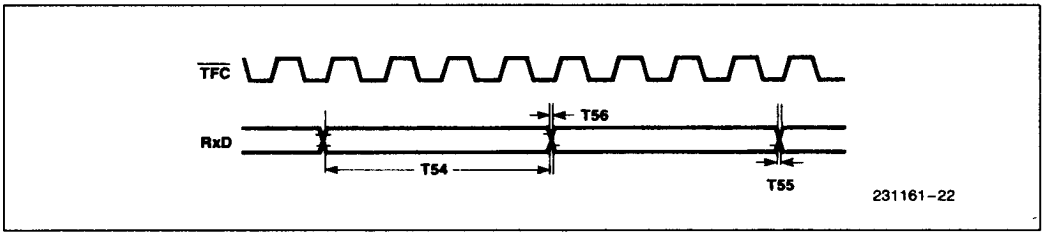
Transmit Timings—Lost $\overline{CTS}$



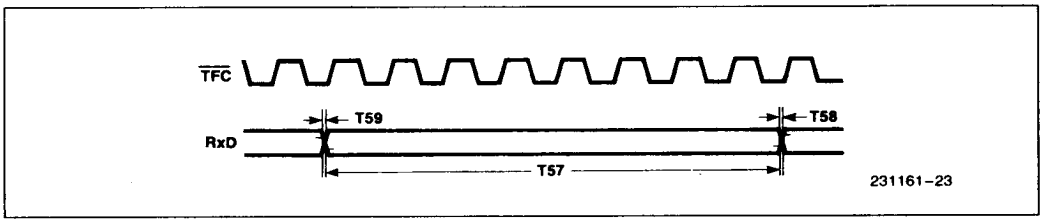
Transmit Timings—NRZI Data Encoding



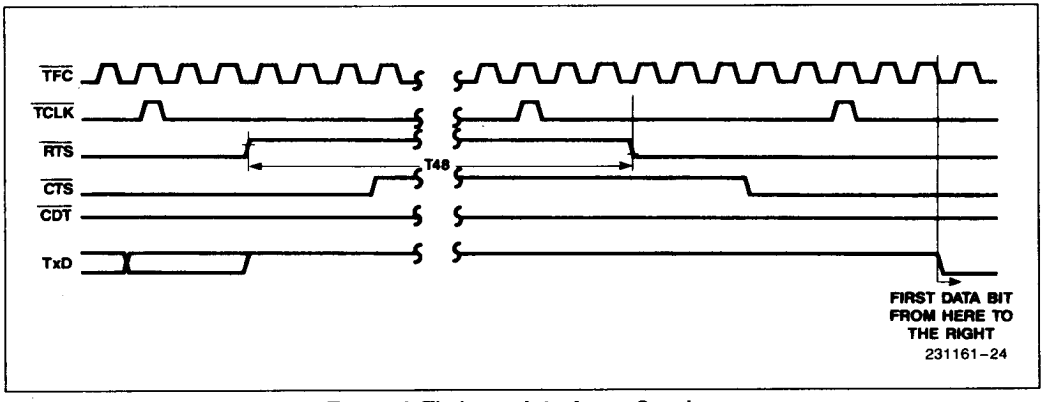
Transmit Timings—Lost $\overline{CTS}$



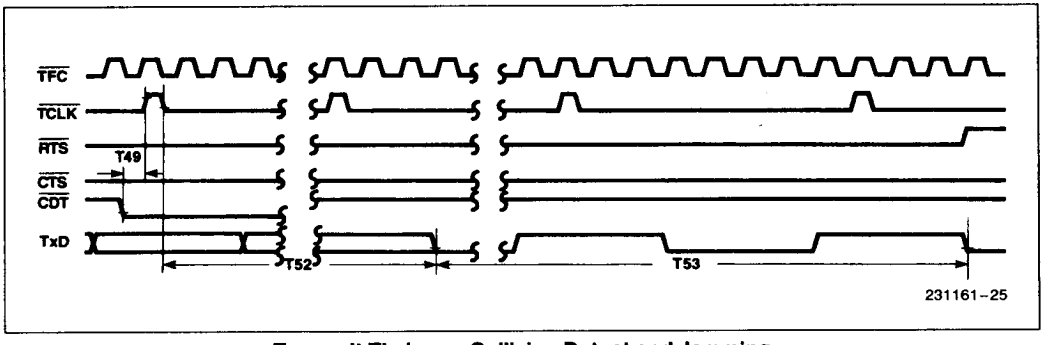
Receive Data Timings (Manchester)



Receive Data Timings (NRZI)



Transmit Timings—Interframe Spacing



Transmit Timings—Collision Detect and Jamming

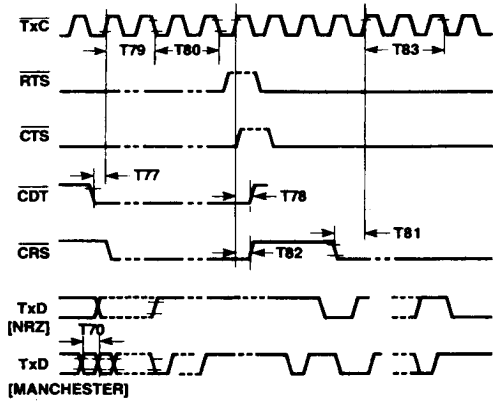
High Speed Mode

Symbol	Parameter	Min	Max	Units	Test Conditions
Transmit/Receive Clock Parameters					
T60	$\overline{RxC}$ $\overline{TxC}$ Cycle	200	*13	ns	
T61	$\overline{TxC}$ Rise Time		10	ns	*1
T62	$\overline{TxC}$ Fall Time		10	ns	*1
T63	$\overline{TxC}$ High	80	1000	ns	*1, *3
T64	$\overline{TxC}$ Low	80		ns	*1, *3
Transmit Data Parameters					
T65	TxD Rise Time		20	ns	*4
T66	TxD Fall Time		20	ns	*4
T67	$\overline{TxC}$ Low to TxD Valid		60	ns	*4, *6
T68	$\overline{TxC}$ Low to TxD Transition		60	ns	*2, *4
T69	$\overline{TxC}$ High to TxD Transition		60	ns	*2, *4
T70	TxD Transition— Transition	70			*2, *4
T71	$\overline{TxC}$ Low to TxD High (At the Transmission End)		60	ns	*4
$\overline{RTS}$, $\overline{CTS}$ Parameters					
T72	$\overline{TxC}$, Low to $\overline{RTS}$ Low Time to Activate $\overline{RTS}$		60	ns	*5
T73	$\overline{CTS}$ Low to $\overline{TxC}$ Low $\overline{CTS}$ Setup Time	65		ns	
T74	$\overline{TxC}$ Low to $\overline{RTS}$ High		60	ns	*5
T75	$\overline{TxC}$ Low to $\overline{CTS}$ Invalid	20		ns	
T75A	$\overline{CTS}$ High to $\overline{TxC}$ Low $\overline{CTS}$ Set-up Time to Stop Transmission	65		ns	*7
Interframe Spacing Parameters					
T76	Inter Frame Delay	*9			
$\overline{CRS}$, $\overline{CDT}$, Parameters					
T77	$\overline{CDT}$ Low to $\overline{TxC}$ High External Collision Detect Setup Time	45		ns	
T78	$\overline{TxC}$ High to $\overline{CDT}$ Inactive $\overline{CDT}$ Hold Time	20		ns	*14
T79	$\overline{CDT}$ Low to Jamming Start		*10		
T80	Jamming Period	*11			
T81	$\overline{CRS}$ Low to $\overline{TxC}$ High Carrier Sense Setup Time	45		ns	*14
T82	$\overline{TxC}$ High to $\overline{CRS}$ Inactive $\overline{CRS}$ Hold Time	20		ns	*14

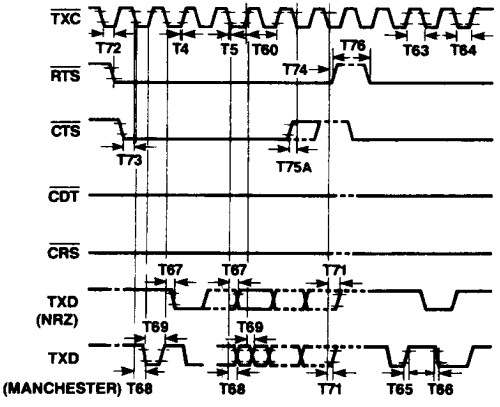
High Speed Mode (Continued)

Symbol	Parameter	Min	Max	Units	Test Conditions
$\overline{CRS}$, $\overline{CDT}$, Parameters (Continued)					
T83	$\overline{CRS}$ High to Jaming (Internal Collision Detect)		*12		
T84	$\overline{CRS}$ High to $\overline{RxC}$ High. End of Receive Packet	80		ns	
T85	$\overline{RxC}$ High to $\overline{CRS}$ High. End of Receive Packet.	20		ns	
Receive Clock Parameters					
T86	$\overline{RxC}$ Rise Time		10	ns	*1
T87	$\overline{RxC}$ Fall Time		10	ns	*1
T88	$\overline{RxC}$ High Time	80		ns	*1
T89	$\overline{RxC}$ Low Time	80		ns	*1
Received Data Parameters					
T90	RxD Setup Time	45		ns	*1
T91	RxD Hold Time	45		ns	*1
T92	RxD Rise Time		20	ns	*1
T93	RxD Fall Time		20	ns	*1

NOTES:
*1 — MOS levels.
*2 — Manchester only.
*3 — Manchester. Needs 50% duty cycle.
*4 — 1 TTL load + 50 pF.
*5 — 1 TTL load + 100 pF.
*6 — NRZ only.
*7 — Abnormal end to transmissions: $\overline{CTS}$ expires before $\overline{RTS}$.
*8 — Normal end to transmission.
*9 — Programmable value.
T76 = NIFS × T60 (ns)
NIFS - the IFS configuration value.
If NIFS is less than 12, then NIFS is enforced to 12.
*10 — Programmable value:
T79 = NCDF × T60 + (12 to 15) × T60 (ns) (if collision occurs after preamble).
*11 — T80 = 32 × T60
*12 — Programmable value:
NCSF × TTRC + (12 to 15) × TTRC
T83 = NCSF × T60 + (12 to 15) × T60
NCDF - collision detect filter configuration value.
*13 — 2000 ns if configured for Manchester encoding.
*14 — To guarantee recognition on the next clock.

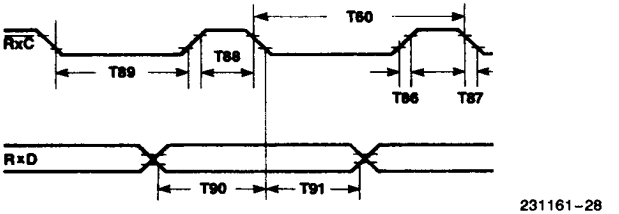


231161-26

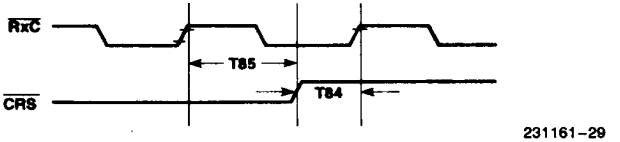


231161-27

Transmit Data Waveforms



Receive Data Waveforms (NRZ)



Receive Data Waveforms